

January 1991

A Proposed Legal Advisor's Roadmap for Software Developers: On the Shoulders of Giants May No Breachers of Economic Relationships nor Slavish Copiers Stand

John T. Soma

Robert D. Sprague

M. Susan Lombardi

Carolyn M. Lindh

Follow this and additional works at: <https://digitalcommons.du.edu/dlr>

Recommended Citation

John T. Soma, Robert D. Sprague, M. Susan Lombardi & Carolyn M. Lindh, A Proposed Legal Advisor's Roadmap for Software Developers: On the Shoulders of Giants May No Breachers of Economic Relationships nor Slavish Copiers Stand, 68 *Denv. U. L. Rev.* 191 (1991).

This Article is brought to you for free and open access by the Denver Law Review at Digital Commons @ DU. It has been accepted for inclusion in Denver Law Review by an authorized editor of Digital Commons @ DU. For more information, please contact jennifer.cox@du.edu, dig-commons@du.edu.

A Proposed Legal Advisor's Roadmap for Software Developers: On the Shoulders of Giants May No Breachers of Economic Relationships nor Slavish Copiers Stand

A PROPOSED LEGAL ADVISOR'S ROADMAP FOR SOFTWARE DEVELOPERS: ON THE SHOULDERS OF GIANTS MAY NO BREACHERS OF ECONOMIC RELATIONSHIPS NOR SLAVISH COPIERS STAND

BY

JOHN T. SOMA*

ROBERT D. SPRAGUE**

M. SUSAN LOMBARDI***

CAROLYN M. LINDH****

INTRODUCTION

Computer programs are literary works entitled to copyright protection under federal law. Although this may seem common knowledge today, until recently courts were still being asked to determine whether computer programs were entitled to any copyright protection at all. The debate began when the first microcomputers became available in the mid-1970s and the number of people using computers increased dramatically. By 1980, companies such as Tandy, Commodore, and Apple all offered a complete line of microcomputers. IBM, then the largest computer company in the world, entered the market in 1981 with the extremely popular microcomputer called the PC (Personal Computer).¹ Much of the software for the early microcomputers was shareware or public domain.² Many of the initial software developers were one- or two-person companies, usually run out of the programmers' basements or garages. Overall, the microcomputer software industry was small, unstable, and unorganized, and developers did not vigorously seek copyright protection.

During the 1980s, however, microcomputer software development

* John T. Soma, Professor of Law, University of Denver College of Law; B.A., 1970, Augustana College; J.D. 1973, University of Illinois College of Law; M.A. 1973, University of Illinois School of Commerce; Ph.D., 1975, University of Illinois School of Commerce.

** Robert D. Sprague, Senior Research Fellow; B.S. B.A., 1980, University of Denver; J.D., 1985, University of Denver College of Law.

*** M. Susan Lombardi, Associate Attorney, Hayes, Phillips & Mahoney, Denver, Colorado; B.A. 1983, University of Colorado; J.D. 1989, University of Denver College of Law.

**** Carolyn M. Lindh, Research Associate, University of Denver College of Law; B.A., 1988, University of Northern Colorado; J.D. Candidate, May 1991, University of Denver College of Law.

1. A "Personal Computer" or "PC" is a microcomputer—a computer which operates with a microprocessor. The terms PC, personal computer and microcomputer are, both in general usage and in this article, considered synonymous.

2. Shareware is software distributed initially without charge or for a nominal fee. If the user likes the program and uses it, he or she is encouraged to send a registration fee to the software's author. Public domain software has no proprietary rights attached. The software can be freely copied and distributed without charge.

became a multi-billion dollar industry with tens of thousands of employees. Small startups such as Microsoft Corporation and Lotus Development Corporation began to approach one billion dollars per year in sales. As software developers established major positions in the market and began to appreciate the money to be made, they wanted protection for their products. And, for the first time, they had the money to seek such protection.

Developers initially turned to the copyright laws for protection, but found them to be inadequate. Responding to this demand for protection, courts and Congress over the last fifteen years have refined the copyright laws to accommodate computer programs. *Lotus Development Corp. v. Paperback Software International*³ is the most recent in a long line of such software protection cases. To better understand the holding in *Lotus*, the practitioner should be familiar with the development of software copyright cases and laws during the last fifteen years.

This article is intended to aid the practitioner in understanding the direction software copyright case law is taking. To do so, the article will review past cases and offer insight into the *Lotus* decision. Further, it will assist the practitioner in advising a software development client as to both what to do and not to do to avoid future copyright infringement litigation. Finally, the graphic "roadmap" provided will not only guide the practitioner through past cases, but also give direction for safe passage in the future.

Section I of this article offers a brief overview of the history and development of the copyright acts. Section II provides a background of the court-derived copyright laws, while section III reviews the more significant software copyright cases of the last fifteen years. It is the contention of the authors that these cases can be separated into three generations of software copyright cases, with each successive generation constantly expanding the legal envelope of protection. Section IV then provides a thorough analysis of the *Lotus* case. Section V follows with a chart of approximately 25 cases which can be used to help the practitioner predict a future case. Finally, section VI will attempt to predict the future of copyright law based upon the decisions discussed in this article.

I. OVERVIEW OF THE HISTORY AND DEVELOPMENT OF THE COPYRIGHT ACTS

The United States Constitution authorizes Congress to pass laws "To promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."⁴ This provision gives Congress the authority to confer monopolies to the extent it deems necessary to pro-

3. 740 F. Supp. 37 (D. Mass. 1990).

4. U.S. Const. art. I, § 8, cl. 8. The copyright laws, codified in Title 17 of the United States Code, rest upon this explicit grant of legislative authority.

mote learning, culture and development.⁵ Monopolies are not granted solely for the purpose of rewarding authors. Rather, Congress has granted copyright monopolies to serve the public welfare by encouraging authors to generate new ideas and disclose them to the public, being free to do so in any uniquely expressed way they choose.⁶

In deciding the nature of the copyright laws, however, Congress must strike a balance between encouraging new expression and allowing that new expression to be used by others.⁷ Achieving the proper balance has been a painstaking task.

[Courts] must take care to guard against two extremes equally prejudicial; the one, that men of ability, who have employed their time for the service of the community, may not be deprived of their just merits, and the reward for their ingenuity and labour; the other, that the world may not be deprived of improvements, nor the progress of the arts be retarded.⁸

It was with these goals in mind that Congress passed the first Copyright Act in 1790.⁹ The 1790 Act attempted to list all the items that would be granted copyright protection. This list soon proved inadequate.¹⁰ In 1909 Congress changed its approach and extended protection to "all the writings of an author."¹¹ Even this general statement proved to be too inflexible to accommodate advances in technology.¹² In the mid-1950s, Congress began consideration of another major revi-

5. See *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1235 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987).

6. *Harper & Row, Publishers, Inc. v. Nation Enterprises*, 471 U.S. 539 (1985).

7. If Congress were to determine, for example, that copyright protection is unnecessary to "promote the progress of" computer programming—because, for example, in Congress' view the financial incentives alone of developing new computer programs (without the added benefit of copyright) are enough to encourage innovation, or because incremental innovation might be stifled by expansive copyright protection—then Congress could, without offending the Constitution, provide no copyright protection for computer programs. At the other extreme, were Congress to find that strong copyright protection is necessary to promote the progress of computer programming, Congress could provide for expansive copyright protection for all aspects of computer programs, again without having strayed beyond the bounds of the constitutionally permissible.

Lotus, 740 F. Supp. at 46.

8. *Whelan*, 797 F.2d at 1235 n. 27, quoting *Sayre v. Moore*, 102 Eng. Rep. 138, 140 n.6 (1785) (Lord Mansfield).

9. Act of May 31, 1790, ch. 15, § 1, 1 Stat. 124, 124 (repealed 1831). This Act extended copyright protection to "any map, chart, book or books already printed."

10. Congress expanded copyright protection by adding "designs, prints, etchings and engravings" in 1802, "musical composition" in 1831, "dramatic composition" in 1856, "photographs and the negatives thereof" in 1865, and "statuary" and "models or designs intended to be perfected as works of the fine arts" in 1870. Act of Apr. 29, 1802, ch. 36, § 2, 2 Stat. 171, 171, *repealed by* Act of Feb. 3, 1831, ch. 16, §§ 1, 14, 4 Stat. 436, 436, 439, *amended by* Act of Aug. 18, 1856, ch. 169, 11 Stat. 138, 139, *amended by* Act of Mar. 3, 1865, ch. 126, §§ 1, 2, 13 Stat. 540, 540, *repealed by* Act of July 8, 1870, ch. 230, § 86, 16 Stat. 198, 212 (repealed 1909).

11. Act of Mar. 4, 1909, ch. 320, § 4, 35 Stat. 1075, 1076 (previously codified at 17 U.S.C. § 4, *reprinted in* 17 U.S.C.A. App. § 4 (West Supp. 1990); recodified 1947; repealed 1976).

12. In 1912, Congress added "motion pictures" as a further example of "all the writings of an author," Act of Aug. 24, 1912, ch. 356, § 5(1)-(m), 37 Stat. 488, 488 (previously codified at 17 U.S.C. § 5(1)-(m), *reprinted in* 17 U.S.C.A. App. § 5(1)-(m), recodified 1947, repealed 1976), and in 1972, Congress added "sound recordings" to the list, Act of Oct.

sion of the copyright law. It was not until 1976, however, that Congress passed a new copyright act.¹³ The Copyright Act of 1976 extended copyright protection to: "original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device."¹⁴

The 1976 Act also codified the judicially created doctrine of the idea versus expression dichotomy. Section 102(b) states that "[i]n no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work."¹⁵ The Copyright Act of 1976 also impliedly extended copyright protection to computer programs. Congress, in the legislative history for the Act, clearly stated that computer programs were considered literary works and were protected by copyright laws.¹⁶

Further changes to the Copyright Act were forthcoming. In 1978, the National Commission on New Technological Uses of Copyrighted Works (CONTU), a committee formed by Congress to investigate copyright protection for new technologies, delivered its report to Congress.¹⁷ CONTU recommended two changes to the Copyright Act: (1) the Act should include a definition of a computer program and (2) allow copies of computer programs for archival purposes.¹⁸ In 1980, Congress amended the Copyright Act and included both of CONTU's recommendations.¹⁹ The 1976 Act and its 1980 Amendment provide the basis for the courts' analyses in the copyright cases discussed in sections III and IV of this article.²⁰

15. 1971, Pub. L. 92-140, § 1(b), 85 Stat. 391, 391 (previously codified at 17 U.S.C. § 5(n), *reprinted in* 17 U.S.C.A. App. § 5(n), repealed 1976).

13. Act of Oct. 19, 1976, Pub. L. No. 94-553, 90 Stat. 2541 (codified at 17 U.S.C. §§ 101-801).

14. 17 U.S.C. § 102(a) (1988).

15. *Id.* at § 102(b).

16. H.R. REP. NO. 1476, 94th Cong., 2d Sess. 51, *reprinted in* 1976 U.S. CODE CONG. & ADMIN. NEWS 5659, 5664-67, 5731.

17. CONTU's purpose was to study the use of the copyright laws for "automatic systems capable of storing, processing, retrieving, and transferring information", and to make recommendations to ensure that such works were protected by the copyright laws. Act of Dec. 31 1974, Pub. L. No. 93-573, § 201(b)-(c), 88 Stat. 1873, 1873-74 (1974).

18. NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, 20-21 (1978), *reprinted in* 5 COPYRIGHT, CONGRESS AND TECHNOLOGY: THE PUBLIC RECORD (N. Henry ed. 1980).

19. Act of Dec. 12, 1980, Pub. L. No. 96-517, §§ 9-10, 94 Stat. 3015, 3028 (codified at 17 U.S.C. § 101 (1988)). The Copyright Act currently defines a computer program as "[a] set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." *Id.*

20. One commentator believes that the 1976 Act and the 1980 Amendment are not adequate protection for software developers, and has suggested amending the Copyright Act again. See Abramson, *Why Lotus-Paperback Uses the Wrong Test and What the New Software Protection Legislation Should Look Like*, COMPUTER LAW., August 1990 at 6.

II. GENERAL OVERVIEW OF COPYRIGHT LAW AS IT RELATES TO COMPUTER SOFTWARE

The fundamental basis of copyright law is that only the expression of a work is subject to protection, not its underlying idea. To establish copyright infringement, both ownership and copying must be proved.²¹ Ownership requires originality and copyrightability of the subject matter.²² Copying requires proof of access and substantial similarity of the two works.²³ Copyright infringement of computer software can first be separated into two distinct categories based upon what part of the work is copied: (1) literal elements and (2) non-literal elements. The source code and object code of computer programs constitute the literal elements.²⁴ Literal aspects of a computer program are clearly copyrightable.²⁵

In general, the user interface—the part of the computer program which the user sees and uses to interact with the program—is the non-literal aspect of the expression. Courts have long recognized that non-literal expressions of a work can be copyrightable and subject to protection.²⁶ When the alleged copying is of literal elements, substantial similarity is established by showing duplication of the original work or substantial portions of it. Establishing substantial similarity of non-literal elements is much more difficult. In determining the substantial sim-

21. *Warner Bros. v. Amer. Broadcasting Cos.*, 654 F.2d 204, 207 (2d Cir. 1981); *Manufacturers Technologies, Inc. v. Cams, Inc.*, 706 F. Supp. 984, 990 (D. Conn. 1989); 3 M. NIMMER, NIMMER ON COPYRIGHT § 13.01 at 13-14 (1990).

22. Ownership also requires the author to be a U.S. citizen and proper registration of the copyright. If the plaintiff is not the author of the work, then the plaintiff must be an assignee of the copyright. In all the cases discussed in this article, these elements are not at issue and are, therefore, not considered pertinent. For further discussion concerning these elements, see 3 M. NIMMER, NIMMER ON COPYRIGHT, § 13.01[A] (1990).

23. Proof by direct evidence of copying is generally not possible since the actual act of copying is rarely witnessed or recorded. Normally, there is no physical proof of copying other than the offending object itself. Copying therefore is generally established by showing that the defendant had access to the copyrighted work and that the offending and copyrighted articles are 'substantially similar.' *Atari, Inc. v. North Amer. Phillips Consumer Elec. Corp.*, 672 F.2d 607, 614 (7th Cir. 1982), cert. denied, 459 U.S. 880 (1982).

Concrete Mach. Co. v. Classic Lawn Ornaments, 843 F.2d 600, 606 (1st Cir. 1988). See also NIMMER ON COPYRIGHT *supra* note 21 at § 13.01.

24. Source code is the program written in a programming language such as Pascal or BASIC. Object code is the machine readable code, that is, the binary representation of the source code. A command written in source code is translated into object code in order for the computer to interpret the command. See *Lotus*, 740 F. Supp. 37, 43-45 for a further discussion of the literal aspects of computer programs.

25. See *Id.* at 45 for a list of cases supporting the holding "that literal manifestations of a computer program—including both source code and object code—if original, are copyrightable."

26. "[A]n infringement is not confined to literal and exact repetition or reproduction; it includes also the various modes in which the matter of any work may be adopted, imitated, transferred, or reproduced, with more or less colorable alterations to disguise the piracy." *Universal Pictures Co., Inc. v. Harold Lloyd Corp.*, 162 F.2d 354, 360 (9th Cir. 1947).

And, as Judge Learned Hand stated, copyright "cannot be limited literally to the text, else a plagiarist would escape by immaterial variations." *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930). See also *Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.*, 562 F.2d 1157, 1167 (9th Cir. 1977).

ilarity between two computer programs based upon non-literal elements, courts have used three basic approaches: (1) the "abstractions" test, (2) the "idea/expression dichotomy" test, and (3) the "look and feel" test.

A. *The Abstractions Test*

The abstractions test was developed by Judge Learned Hand in *Nichols v. Universal Pictures Co.*:²⁷

Upon any work, and especially upon a play, a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out. The last may perhaps be no more than the most general statement of what the play is about, and at times might consist only of its title; but there is a point in this series of abstractions where they are no longer protected, since otherwise the playwright could prevent the use of his "ideas," to which, apart from their expression, his property is never extended.²⁸

In other words, an author cannot claim protection for a basic plot such as "boy meets girl," nor can protection be claimed because the plot unfolds in a particular city. But where the plot, location, types of characters, and segments of dialogue all start to become substantially similar, the elements of the work then begin to constitute more of an expression subject to protection than of abstract ideas.²⁹

The critical step under the abstractions test, therefore, is distilling the unprotected idea from the protected expression.³⁰ Applying the abstractions test has led to a two-step analysis to determine substantial similarity between two works: applying first an "extrinsic test," and then an "intrinsic test."³¹ Under the initial extrinsic test, the court will dissect and analyze the basic components of a work (i.e., the type of work involved, the materials used, the subject matter, and the setting for the subject), to determine "whether there are sufficient articulable similarities to justify a finding that the defendant has copied from the protected work."³²

If copying is established under the extrinsic test, the court then applies the intrinsic test to determine whether copying has occurred to the extent that the two works are substantially similar. The basis of the test applied in deciding whether there is substantial similarity in expressions is "the response of the ordinary reasonable person."³³

In applying the abstractions test, the court in *Sid & Marty Krofft Tele-*

27. 45 F.2d 119 (2d Cir. 1930) (copyright infringement of a play by a motion picture).

28. *Id.* at 121 (citations omitted).

29. *Id.*

30. *Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.*, 562 F.2d 1157, 1163 (9th Cir. 1977).

31. *Id.* at 1164.

32. *Concrete Mach. Co. v. Classic Lawn Ornaments*, 843 F.2d 600, 608, *relying upon*, *Arnstein v. Porter*, 154 F.2d 464, 468 (2d Cir. 1946).

33. *Krofft*, 562 F.2d at 1164, *relying upon*, *Arnstein*, 154 F.2d at 468-69.

*vision Production, Inc. v. McDonald's Corp.*³⁴ stated that if "there is substantial similarity in ideas, then the trier of fact must decide whether there is substantial similarity in the expressions of the ideas so as to constitute infringement."³⁵ This test has led to an evolution of the basic test to establish copyright infringement when analyzing non-literal elements: "To establish copyright infringement, the plaintiff must prove ownership of the work in question, access to the work by the defendant, and substantial similarity of both the general ideas *and* the expression of those ideas between the plaintiff's and defendant's work."³⁶

B. *Idea/Expression Dichotomy*

The idea/expression dichotomy attempts to differentiate between an idea and its expression. This test was first applied in *Baker v. Selden*.³⁷ As noted in *Lotus*, the *Baker* Court held that:

the text of a book describing a special method of double-entry accounting on paper spreadsheets—the now almost universal T-accounts system—was copyrightable expression, but that the method itself, which embodied the idea of this particular kind of double-entry bookkeeping, was not. The Court thus concluded that Baker did not infringe Selden's copyright when Baker wrote his own treatise, in his own words, describing the special double-entry method of bookkeeping.³⁸

If it is determined that the idea and its expression are indistinguishable, and that there is no greater similarity between the works than is inevitable from the expression of that idea, then any copying of that particular expression will essentially be excused because there is no other way to express that idea. The idea and expression have merged.³⁹ Conversely, "the scope of copyright protection increases with the extent expression differs from the idea."⁴⁰

C. *Look and Feel*

The origination of the look and feel concept is credited to *Roth Greeting Cards v. United Card Co.*,⁴¹ in which the court considered whether greeting cards created by the defendant infringed upon the plaintiff's cards. In dissecting various elements of expression of the plaintiff's cards, the court noted:

the textual matter of each card, considered apart from its ar-

34. 562 F.2d 1157 (9th Cir. 1977).

35. *Id.*

36. *Worth v. Selchow & Righter Co.*, 827 F.2d 569, 571 (9th Cir. 1987), *cert. denied*, 485 U.S. 977 (1988) (emphasis in original), *citing* *Sid & Marty Krofft Television Prods. v. McDonald's Corp.*, 562 F.2d 1157, 1164 (9th Cir. 1977).

37. 101 U.S. 99 (1879).

38. *Lotus Dev. Corp. v. Paperback Software Int'l.*, 740 F. Supp. 37, 54 (D. Mass. 1990).

39. *Worlds of Wonder, Inc. v. Vector Intercontinental, Inc.*, 653 F. Supp. 135, 139 (N.D. Ohio 1986).

40. *Krofft*, 562 F.2d at 1168.

41. 429 F.2d 1106 (9th Cir. 1970).

rangement on the cards and its association with artistic representations, was not original to Roth and therefore not copyrightable. However, proper analysis of the problem requires that all elements of each card, including text, arrangement of text, art work, and association between art work and text, be considered as a whole.⁴²

The *Roth* court, keeping in mind that it was protecting expression and not ideas, determined that plaintiff's cards, taken as a whole, were copyrightable.⁴³ In determining whether there was a substantial similarity between the two works, the court held that "in *total concept and feel* the cards of United are the same as the copyrighted cards of Roth."⁴⁴

III. THREE GENERATIONS OF COMPUTER COPYRIGHT CASES

The 1976 Copyright Act and the 1980 Amendment extended copyright protection to computer programs. The degree of protection extended, however, was left to the courts. The cases applying copyright protection to computer programs can be separated into three generations. The first generation of software copyright cases dealt with fragmented literal copying consisting mainly of the copying of substantial portions of the source and object codes. In the second generation, the courts began to consider the extent non-literal aspects of computer programs were protected. The courts' focus was on the structure of the source and object code. The third generation of cases further extended protection of non-literal elements, focusing on the structure and organization of the user interface.

A. *The First Generation*

The first generation of software copyright cases culminated with *Apple Computer, Inc. v. Franklin Computer Corp.*⁴⁵ Prior to *Apple*, the courts generally recognized the copyrightability of the source code of computer programs. The basic definition in the 1980 amendment made it clear that Congress intended to protect the source code.⁴⁶ *Apple* provided the court with the novel issue of whether the object code of a program and a program embedded in read-only memory (ROM) chips were copyrightable. The defendant in *Apple*, Franklin Computer Corporation, manufactured and sold the ACE 100 personal computer which was designed to be "Apple compatible." In order to be Apple compati-

42. *Id.* at 1109 (footnote omitted).

43. Considering all of these elements together, the Roth cards are, in our opinion, both original and copyrightable. In reaching this conclusion we recognize that copyright protection is not available for ideas, but only for the tangible expression of ideas. *Mazer v. Stein*, 347 U.S. 201, 217, 74 S. Ct. 460, 98 L. Ed. 630 (1954). We conclude that each of Roth's cards, considered as a whole, represents a tangible expression of an idea and that such expression was, in totality, created by Roth.

Id. at 1109-10 (citation omitted).

44. *Id.* at 1110 (emphasis added). See also *Sid & Marty Krofft Television Prods. Inc., v. McDonald's Corp.*, 562 F.2d 1157 (9th Cir. 1977).

45. 714 F.2d 1240 (3rd Cir. 1983), *cert. dismissed*, 464 U.S. 1033 (1984).

46. See *supra* n. 19.

ble, Franklin copied Apple's operating system verbatim.⁴⁷ Apple sued Franklin for copyright infringement. Franklin raised two defenses based upon the non-copyrightability of the subject matter: (1) object code and programs embedded in ROM were not copyrightable subject matter and (2) operating systems were not copyrightable subject matter.⁴⁸

Franklin asserted that the object code was not copyrightable because of the 1908 case of *White-Smith Music Publishing Co. v. Apollo Co.*⁴⁹ In *White-Smith*, the Supreme Court had drawn a distinction between works that could be interpreted by individuals, and those that required a machine to interpret the work. The *Apple* court found that both the 1976 Act and its legislative history clearly intended to "obliterate distinctions engendered by *White-Smith*."⁵⁰ Under the Act, copyright protection extends to works in any tangible forms which can be "perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device."⁵¹ In addition, the definition of a computer program includes "sets of statements or instructions to be used *directly or indirectly* in a computer."⁵² The Third Circuit concluded that a computer program, whether in object code or source code, was a "literary work" and was protected from unauthorized copying, whether from its object or source code version.⁵³

The *Apple* court also rejected Franklin's second argument that operating systems were not copyrightable. This argument was based upon the premise that operating systems are methods or processes.⁵⁴ The court reasoned that if the instructions in an application program are not methods or processes, then by analogy neither are the instructions in a system program. The court relied on the CONTU report which stated that works of a program which are "used ultimately in the implementation of a process should in no way affect their copyrightability."⁵⁵ *Apple* clearly established that application programs and operating systems, in either object or source code form, are copyrightable.

B. *The Second Generation*

The second generation of software copyright cases, represented by *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc.*,⁵⁶ extended protection to the structure, sequence and organization (non-literal aspects) of

47. There was evidence at trial that Franklin had blatantly copied Apple's programs. Franklin had even left in the object code the name of the original programmer and the original program's name, "Applesoft." *Apple*, 714 F.2d at 1245.

48. *Id.* at 1249.

49. 209 U.S. 1 (1908).

50. *Apple*, 714 F.2d at 1240, 1248.

51. 17 U.S.C. § 102(a) (1988).

52. *Id.* (emphasis added).

53. *Apple*, 714 F.2d at 1249.

54. The Copyright Act does not protect a method or process. See 17 U.S.C. § 102(b) (1988).

55. *Apple*, 714 F.2d at 1252 (quoting NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT 1 (1979) at 21).

56. 797 F.2d 1222 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987).

a program's source and object code. The plaintiff, Whelan Associates, developed a program for the operation of a dental laboratory for the defendant, Jaslow Dental Laboratories. The program, Dentalab, was written in Event Driven Language for the IBM Series One computer. Jaslow and Whelan entered into a licensing agreement which authorized Whelan to sell Dentalab and to pay Jaslow a ten percent royalty on all sales. Jaslow used Dentalab for two years. Jaslow later realized, however, that if the program was converted to BASIC programming language, it could be used on a wider variety of personal computers. Jaslow thereafter developed and marketed its own version of the program in BASIC. Jaslow was ultimately sued for infringing Whelan's copyright of the software written in Event Driven Language. The *Whelan* court was presented with the issue of whether the structure and organization of the computer code could be infringed. The court analyzed the issue under the idea/expression dichotomy. Jaslow argued that the computer program's structure was the idea, not the expression. The court rejected Jaslow's argument based upon the distinction made in *Baker v. Selden*⁵⁷ between an idea and an expression. The *Whelan* court noted that in *Baker*, the purpose or function of the work was the idea, and everything else was the expression.⁵⁸ The *Whelan* court then determined that the idea of Dentalab was the efficient management of a dental laboratory and the expression was everything else, including the structure of the program. The court noted that the Copyright Act of 1976 implicitly protects the sequence, order, or structure of a work:

Although the Code does not use the terms "sequence," "order" or "structure," it is clear from the definition of compilations and derivative works, and the protection afforded them, that Congress was aware of the fact that the sequencing and ordering of materials could be copyrighted. . . . [T]he sequence and order could be parts of the expression, not the idea.⁵⁹

Jaslow next contended that if the structure was "expression," then it merged with the idea. The court reasoned that since the program's structure could be written in a variety of ways, the structure did not merge with the idea.⁶⁰

The *Whelan* decision has been criticized for misapplying the idea/expression dichotomy. The court identified the idea underlying the program and then decided that everything else was expression. In so doing, the court assumed that there could only be one idea in every program. Certain functions of a computer can only be performed a particular way, usually because of hardware configurations or the programming language involved. Additionally, certain types of programs require a certain basic structure. To assume, however, that there was only one idea

57. 101 U.S. 99 (1879).

58. 797 F.2d at 1236. Cf. *Apple Computer, Inc. v. Formula Int'l, Inc.*, 562 F. Supp. 775, 783 (C.D. Cal. 1983), *aff'd*, 725 F.2d 521 (9th Cir. 1984) ("Apple seeks here not to protect ideas, (i.e., making the machine perform particular functions) but rather to protect their particular expressions . . .").

59. *Whelan*, 797 F.2d at 1239.

60. *Id.* at 1236.

underlying the program was erroneous. The court should have applied the abstractions test to ascertain whether there were other ideas not subject to protection.

Whelan extended copyright law further than other courts. The Third Circuit effectively prohibited developers from producing *substantially similar* source and object codes. Courts will not extend protection to source and object codes beyond this point. *Whelan* set the stage for the third generation in copyright protection where the issue became whether the copyright laws protect the structure, sequence, and organization of a program's user interface.

C. *The Third (and Final?) Generation*

In the third generation of software copyright cases, courts have struggled with the question of what other non-literal aspects of a computer program are protected. There are three major third generation cases.

*Synercom Technology, Inc. v. University Computing Co.*⁶¹ is generally recognized as the first case to decide if a computer program's user interface is copyrightable. Synercom developed a program to test the tolerance levels of a building. The program had an easier and more efficient method to input data on format cards. The format cards had lines and shaded areas that told "the user what data to place where and how to do it."⁶² The input forms had been properly copyrighted. The defendant, Engineering Dynamics, Inc. (EDI) developed another program that competed with Synercom's program. In fact, EDI believed that to compete in the marketplace, its new program had to be wholly compatible with Synercom's data input format.⁶³ EDI did not copy the format cards themselves, but instead wrote a program to accept data from Synercom's input format cards.⁶⁴

The *Synercom* court began its analysis by using the idea/expression dichotomy, approaching this issue differently, however, than did the *Whelan* court. In *Whelan*, the court tried to discern what the idea was behind the program, whereas in *Synercom*, the court tried to discern "whether the material proffered for copyright undertakes to express."⁶⁵ Judge Higginbotham found that the input format did express an idea. The lines, shading, and words communicated to the user the selection, arrangement and sequence of data.⁶⁶ The issue presented was: "If se-

61. 462 F. Supp. 1003 (N.D. Tex. 1978) (Although it was decided in 1978, this case is identified with the third generation because it addressed issues not generally raised until the mid-1980s, and is often distinguished in the holdings of third-generation cases.).

62. *Id.* at 1012.

63. *Id.* at 1008.

64. *Id.* at 1012.

65. *Id.* at 1011.

66. Generally, blank forms are not the subject of copyright. See *Baker v. Selden*, 101 U.S. 99 (1879); *Aldrich v. Remington Rand, Inc.*, 52 F. Supp. 732 (N.D. Tex. 1942). "Forms" which communicate information can, however, be the subject of copyright. See *Harcourt, Brace & World, Inc. v. Graphic Controls Corp.*, 329 F. Supp. 517 (S.D.N.Y. 1971).

quencing and ordering [was] expression, what separable idea [was] expressed?"⁶⁷ If the idea was the sequence and ordering of data, there was no infringement; if, however, sequencing and ordering of data was the expression, there was an infringement.⁶⁸

Judge Higginbotham resolved this issue with the now classic example of the "figure-H" pattern in manual transmission automobiles which he analogized to the input formats used in the program. Once a manufacturer chose the "figure-H," it was the only pattern that would work in that particular model of car. The "figure-H" may be expressed in several different ways. It can be described in a driver's manual, through a diagram, photograph, or driver training film. Each of these expressions may be protected through copyright, but a copyright does not prohibit another manufacturer from marketing a car using the same pattern. Judge Higginbotham concluded that the order and sequence of the data was an expressed idea. "[O]nly to the extent the expressions involve stylistic creativity *above and beyond* the bare expression of sequence and arrangement, should they be protected."⁶⁹

The *Synercom* holding has been cited in defense of copyright infringement on the basis that it provides authority that the non-literal aspects of a program—the sequence, organization, and structure—are merely expressed ideas, not copyrightable expression. In subsequent cases, courts have generally not reached the same result as Judge Higginbotham. This may be largely explained by the fact that the 1978 *Synercom* decision preceded both CONTU's report to Congress and the 1980 amendment to the 1976 Copyright Act.

*Broderbund Software, Inc. v. Unison World, Inc.*⁷⁰ was the first case to determine that the structure, sequence and organization of a computer program's audiovisual display is copyrightable. The plaintiff, Broderbund Software, marketed a program called Print Shop for the Apple computer. The defendant, Unison World, contacted Broderbund to negotiate the rights that would allow Unison to adapt Print Shop to run on the IBM/PC. Broderbund tentatively agreed to allow Unison to convert Print Shop. Under the tentative agreement, Unison was required to produce an exact copy of Print Shop. Unison painstakingly duplicated the interface of Print Shop. After Unison had copied a substantial portion of Print Shop, the parties were unable to reach a final agreement, and terminated negotiations. Unison continued to develop its own enhanced version of Print Shop for the IBM/PC called Printmaster. Unison did not, however, rewrite the portions of Printmaster that had already been copied from Print Shop. Broderbund sued Unison for copyright infringement of the audiovisual portion of the program.

The *Broderbund* court also analyzed the idea/expression dichotomy

67. *Synercom*, 452 F. Supp. at 1013.

68. *Id.*

69. *Id.* at 1014 (emphasis in original).

70. 648 F. Supp. 1127 (N.D. Cal. 1986).

underlying the program.⁷¹ The court found that the idea was the creation of greeting cards, banners, posters and signs, and everything else was expression.⁷² Unison argued that the idea and expression merged such that any menu-driven computer program used to print greeting cards, signs, banners, and posters would have a user interface substantially similar to that of Print Shop. At trial, however, the plaintiff introduced evidence of another printing program, Stickybear Printer, that had a substantially different interface from Print Shop. Based upon the Stickybear Printer program, the court concluded that the idea and expression had not merged.⁷³ Unison also raised a variation on the merger doctrine, the rules and instruction test.⁷⁴ The rules and instruction test applies when there are a limited number of ways to express the idea, and granting copyright protection to the expression would be tantamount to protecting the idea or process itself. Again, the court relied upon the Stickybear Print program to reject this argument. Based upon the evidence, there was more than one way to express the idea.

Unison next argued, relying on *Synercom*, that the interface of a computer program is not copyrightable.⁷⁵ The court, however, rejected *Synercom* and found *Whelan*, which extended copyright protection to the structure, sequence, and organization of the program's code, to be the controlling authority. The court reasoned that in non-computer copyright cases, courts protect the non-literal aspects of a work, and that computer programs should be afforded the same protection.⁷⁶ The court noted that Congress "intended sequencing and ordering to be protectable in the appropriate circumstances . . . and the computer field is not an exception to this general rule."⁷⁷ Unison also raised the defense that the audiovisual displays of the Print Shop program were not eligible for copyright protection because they do not fall within the definition of "pictorial" or "graphic" works.⁷⁸ In response to this defense the court ruled:

In the present case, it is clear that the structure, sequence, and layout of the audiovisual displays in "Print Shop" were dictated primarily by artistic and aesthetic considerations, and not by utilitarian or mechanical ones. . . . The bottom line is that the designer of any program that performed the same functions as "Print Shop" had available a wide range of expression governed predominantly by artistic and not utilitarian considerations.⁷⁹

*Digital Communications Assocs. v. Softklone Distribution Corp.*⁸⁰ is the final

71. *Id.* at 1131.

72. *Id.* at 1132.

73. *Id.*

74. *Id.* at 1134.

75. *Id.* at 1132.

76. *Id.* at 1132-33.

77. *Id.* at 1133.

78. *Id.* See 17 U.S.C. § 101 (1988).

79. 648 F. Supp. at 1134.

80. 659 F. Supp. 449 (N.D. Ga. 1987).

third generation case. Plaintiff, Digital, developed Crosstalk XVI, a telecommunications program for personal computers. Crosstalk had a unique "status screen" or "main menu" that made its telecommunications software easier to use than other programs. The defendant, Softklone, decided to clone Crosstalk XVI and aptly called it Mirror. Digital asserted that Softklone's copying of the Crosstalk XVI status screen infringed Digital's copyright of both the status screen and the program.

The *Digital* court concluded that copyright protection of a computer program does not extend to screen displays generated by the program and that "copying of a program's screen displays, without evidence of copying of the program's source code, object code, sequence, organization or structure, does not state a claim of infringement."⁸¹ Since Digital claimed a separate copyright for the Crosstalk XVI main menu, the court next considered whether that copyright had been infringed. Softklone contended that the status screen was "not copyrightable because it is a necessary expression of the idea underlying the status screen and/or because it is simply a 'blank form.'"⁸² The court concluded that the "idea" behind the Crosstalk XVI status screen was the process or manner by which the status screen operated, and the "expression" was the method by which the idea was communicated to the user.⁸³

The court noted that certain aspects of the status screen were ideas that could not be copyrighted. The "idea" underlying the status screen was a two symbol command-driven menu with the changes reflected on a screen listing the computer program's commands. All of these elements related to how the program received commands from the user and how the program reflected the results on the screen, and were thus ideas. Certain aspects of the status screen were, however, unrelated to how the program operated and were "expression." The arrangement of the commands, the sequence for entering the command, and highlighting and capitalizing two letters of the command terms had no relation to how the program operated, and thus were protected expression.⁸⁴

Softklone, also relying upon *Synercom*, contended that the Crosstalk

81. *Id.* at 456. The court based this finding upon its conclusion that: screen displays generated by computer programs are not direct "copies" or "reproductions" of the literary or substantive content of the computer programs. This distinction results from the fact that the same screen can be created by a variety of separate and independent computer programs. It is somewhat illogical to conclude that a screen can be a "copy" of many different programs.

Id. at 455-456.

The court also based its finding upon the conclusions reached in *Whelan, Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987):

While finding that copying of a program's screen displays may serve as indirect evidence of copying of a program, the *Whelan* court did not specifically extend a computer program's copyright protection to its screen displays. The *Whelan* court cited approvingly those cases which have found a computer program's screen displays, at least in the context of "video games," to be separately copyrightable as "audiovisual works."

Digital Communications Assocs., Inc. v. Softklone Distribution Corp., 659 F. Supp 449 (N.D. Ga. 1987) (citations omitted).

82. *Digital*, 659 F. Supp. at 457.

83. *Id.* at 458.

84. *Id.* at 459.

XVI status screen is a necessary expression of the idea it expresses and therefore is expression merged with idea.⁸⁵ *Digital* concluded that the difference between *Synercom* and the instant case was that in *Synercom* the sequence of the data input into the computer was relevant to the functioning of the *Synercom* computer program. The defendant in *Synercom* duplicated the sequence within its "source code" but did not create format cards with the same headings and shaded areas. On the other hand, Softklone's arrangement of the status screen had no relationship to the functioning of the computer program.⁸⁶ The arrangement of the status screen in *Crosstalk XVI* involved "stylistic creativity and authorship above and beyond the ideas embodied in the status screen."⁸⁷ The commands and techniques used in *Mirror* could have been arranged and delineated in an infinite number of ways that would have been different from *Crosstalk*. The modes of expression chosen by the plaintiff for its status screen are not necessary to the "idea." The plaintiff's expression of the status screen, therefore, did not merge with the idea of the status screen.⁸⁸

Softklone's last argument was that the status screen was analogous to a "blank form" and that blank forms which do not convey information or contain original pictorial expression are not copyrightable.⁸⁹ The court concluded, however, that in the instant case "the status screen, even if found to be a 'form,' clearly expresses and conveys information and, therefore, is copyrightable."⁹⁰

The three cases discussed in this section, *Synercom*, *Broderbund*, and *Digital*, illustrate that the courts are not in agreement on the scope of the protection for the structure, sequence, and organization of the non-literal aspect of a computer program. *Synercom* impliedly determined that the structure, sequence, and organization of data input formats are not subject to protection. Both *Digital* and *Broderbund* declined to follow *Synercom*'s lead, and instead found that the program's interface—its method of communicating and accepting data to and from a user—is protected expression, but for different reasons. The uncertainty created by the ad hoc nature of the software cases has hampered the development and progression of the computer software field. Software developers have no adequate guidelines regarding what level of independent development is required to avoid copyright infringement.⁹¹ In *Lotus Development Corp. v. Paperback Software International*, discussed in the next section, Judge Keeton attempted to distill all the previous cases concerning copyright protection of non-literal elements, particularly related to computer software, and define the scope of the Copyright Act as it applies to the non-literal aspects of a computer program.

85. *Id.*

86. *Id.* at 460.

87. *Id.*

88. *Id.*

89. *Id.* at 461.

90. *Id.* at 462.

91. 3 M. NIMMER, NIMMER ON COPYRIGHT § 13.03[A] (1990).

IV. *LOTUS DEVELOPMENT CORPORATION V. PAPERBACK SOFTWARE INTERNATIONAL*⁹²

In *Lotus*, the court was asked to determine two critical issues: "(1) whether and to what extent plaintiff's computer spreadsheet program, Lotus 1-2-3, is copyrightable, [and] (2) whether defendants' VP-Planner [a competing spreadsheet program] was, on undisputable facts, an infringing work containing elements substantially similar to copyrightable elements of 1-2-3."⁹³ It was generally not disputed in *Lotus* "that literal manifestations of a computer program—including both source code and object code—if original, are copyrightable."⁹⁴ The central issue in *Lotus* was whether, and to what degree, non-literal elements of a computer program, in particular the Lotus 1-2-3 user interface, are copyrightable. Central to the court's decision was whether non-literal elements (including the overall organization of a program, the structure of a program's command system, and the presentation of information on the screen) were copyrightable, and if so, how the non-literal elements that are copyrightable could be identified.⁹⁵

The *Lotus* court developed its own three-step test to determine copyrightability of the non-literal elements of computer software. Initially, the underlying idea of the work must be identified. Next, individual elements of expression which comprise the work must be evaluated to determine whether each expression is limited to the functional requirements of the work or is in the public domain, or whether, conversely, it constitutes an original expression. Finally, to determine copyrightability of the work, it must be determined whether any of the elements not determined to be limited to the functional requirements of the work or in the public domain constitute a substantial part of the work.⁹⁶

The first step in the *Lotus* test is essentially a restatement of the abstractions test first expressed by Judge Learned Hand in *Nichols v. Universal Pictures Corp.*⁹⁷ The *Lotus* court noted that, at the most general

92. 740 F. Supp. 37 (D. Mass. 1990).

93. *Id.* at 42.

94. *Id.* at 45.

95. *Id.* at 46.

96. FIRST, in making the determination of 'copyrightability,' the decisionmaker must focus upon alternatives that counsel may suggest, or the court may conceive, along the scale from the most generalized conception to the most particularized, and choose some formulation—some conception or definition of the 'idea'—for the purpose of distinguishing between the idea and its expression.

....

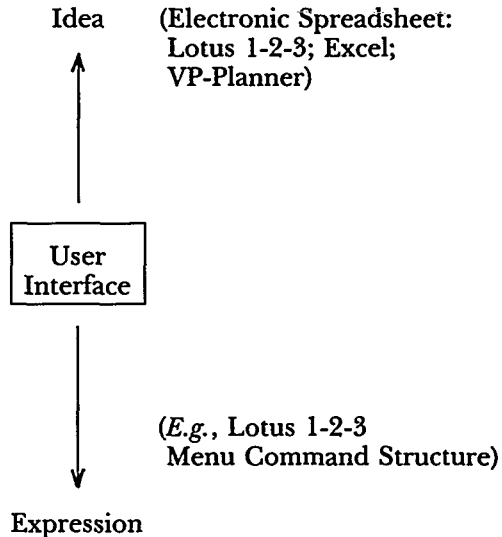
SECOND, the decisionmaker must focus upon whether an alleged expression of the idea is limited to elements essential to expression of that idea (or is one of only a few ways of expressing the idea) or instead includes identifiable elements of expression not essential to every expression of that idea.

THIRD, having identified elements of expression not essential to every expression of the idea, the decisionmaker must focus on whether those elements are a substantial part of the allegedly copyrightable 'work.'

Id. at 60-61.

97. 45 F.2d 119 (2d Cir. 1930).

level, the idea of an electronic spreadsheet is not copyrightable.⁹⁸ Thus, considered at the most general level, the Lotus 1-2-3 user interface, like a blank form, is also not copyrightable. “[I]f a particular expression of the idea of an electronic spreadsheet communicates no details beyond those essential to stating the idea itself, then that expression would not be copyrightable.”⁹⁹ A simple diagram illustrates this analysis:



The key to determining whether the Lotus 1-2-3 user interface is copyrightable is found in the *Lotus* court's use of the idea/expression dichotomy as the second step in its test. “The issue here is whether Lotus 1-2-3 does go beyond those details essential to any expression of the idea, and includes substantial elements of expression, distinctive and original, which are thus copyrightable.”¹⁰⁰

Over the years, the question as to whether an expression is distinctive and original has been determined on a sliding scale. Expressions that were once considered original have entered into the public domain and are now considered essential to the operation of computer programs. Likewise, expressions that today are considered original and non-functional may someday be considered an essential element of a standard interface. For example, the following graph illustrates the progression of the “Esc” key (the standard key to “undo” or back out of an

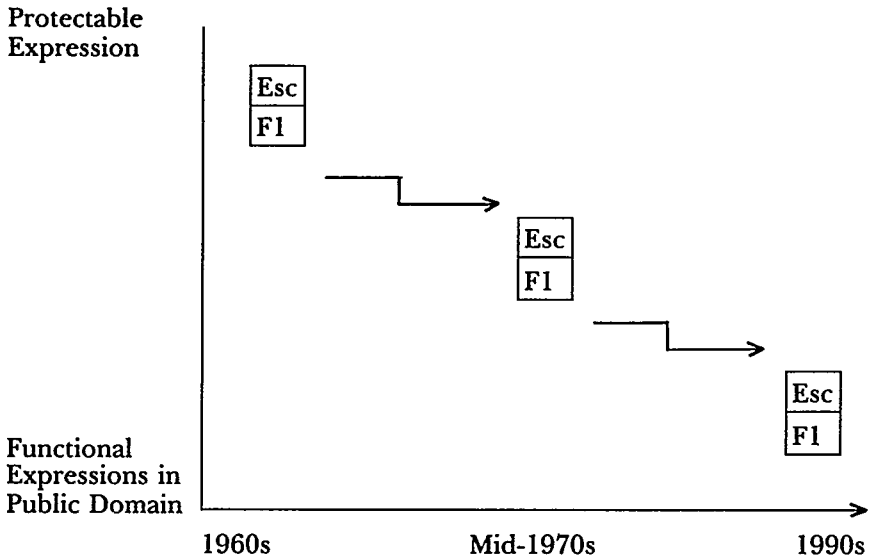
98. At the most general level of Hand's abstractions scale, *Nichols*, 45 F.2d at 121—the computer programs at issue in this case, and other computer programs that have been considered during the course of trial, are expressions of the idea of a computer program for an electronic spreadsheet. Defendants are quite correct, then, in asserting that the idea of developing an electronic spreadsheet is not copyrightable—that the core idea of such a spreadsheet is both functional and obvious, even to computer users who claim no technical competence.

Lotus, 740 F. Supp at 65.

99. *Id.*

100. *Id.*

operation) and the "F1" key (the standard key to invoke on-screen help) from original expression to public domain:



The Lotus 1-2-3 user interface was described by the plaintiffs as including "such elements as 'the menus (and their structure and organization), the long prompts, the screens on which they appear, the function key assignments, [and] the macro commands and language'"¹⁰¹ The court determined that, with the exception of the menu command structure, all of the identified elements of expression within the Lotus 1-2-3 user interface merged with the idea of an electronic spreadsheet (i.e., each is an essential element present in most if not all expressions of an electronic spreadsheet).¹⁰² The *Lotus* court specifically found that "[a]n example of distinctive details of expression is the precise 'structure, sequence, and organization' of the menu command system."¹⁰³

Since the *Lotus* court did identify an expression within the Lotus 1-2-3 user interface that was original and did not merge with the underlying

101. *Id.* at 63 (quoting Plaintiff's Post-Trial Brief at 53).

102. The macro command structure was deemed to be dependent upon the menu command structure. *Id.* at 66-67.

103. *Id.* at 67 (citation omitted).

This particular expression of a menu structure is not essential to the electronic spreadsheet idea, nor does it merge with the somewhat less abstract idea of a menu structure for an electronic spreadsheet. The idea of a menu structure—including the overall structure, the order of commands in each menu line, the choice of letters, words, or "symbolic tokens" to represent each command, the presentation of these symbolic tokens on the screen (i.e., first letter only, abbreviations, full words, full words with one or more letters capitalized or underlined), the type of menu system used (i.e., one-, two-, or three-line moving-cursor menus, pull-down menus, or command-driven interfaces), and the long prompts—could be expressed in a great many if not literally unlimited number of ways.

ing idea of an electronic spreadsheet, the final step in establishing the copyrightability of the Lotus 1-2-3 user interface was accomplished by the court's determination that this particular expression is a substantial part of the user interface.¹⁰⁴ In general, therefore, the more copying of substantial elements which are expressions subject to protection, the greater the likelihood of a finding of copyright violation. This analysis can be expressed as follows:

Insubstantial Element	↑	No Violation	No Violation	No Violation
		Probable Violation	Possible Violation	No Violation
		Definite Violation	Probable Violation	No Violation
Substantial Element	↓	Extensive Copying	Partial Copying	No Copying

The test developed by the *Lotus* court is an attempt to differentiate between copyrightability and copying. Under the *Lotus* test, copyrightability is first determined before copying is considered.¹⁰⁵ The abstractions test and the idea/expression dichotomy, from which the first two steps of the *Lotus* test are derived, have traditionally been applied to determine whether the copying that exists is permissible.

The traditional abstractions test questions whether the defendant has copied so much of the non-literal elements of the plaintiff's work so as to go beyond using the basic underlying idea. The *Lotus* test requires the court to first determine whether the plaintiff's work, taken alone, is no more than an idea or an expression of that idea. The traditional idea/expression dichotomy is concerned with whether the defendant copied portions of the non-literal elements of the plaintiff's work because those particular elements of expression were the only practical expressions available. The *Lotus* test determines whether the plaintiff's selection of expression was dictated by functional requirements or independent, distinctive expression.

Finally, both the traditional approaches were used to determine

104. *Id.* at 68.

105. *Id.* at 42.

substantial similarity (i.e., did the defendant impermissibly copy a substantial portion of the plaintiff's work). As Judge Hand stated when first discussing levels of abstraction, the "question is whether the part so taken is 'substantial,' and therefore not a 'fair use' of the copyrighted work."¹⁰⁶

*Concrete Mach. Co. v. Classic Lawn Ornaments, Inc.*¹⁰⁷ offers another example of the traditional manner in which courts consider infringement of non-literal elements. In discussing the application of the idea/expression dichotomy, the *Concrete* court stated that it:

first must determine whether there has been "copying." This step involves "dissection" of the work, perhaps aided by expert testimony, to assess whether there are sufficient articulable similarities to justify a finding that the defendant has copied from the protected work. . . . Second, once "copying" is established, the court must determine whether the copying is sufficiently substantial to constitute "unlawful appropriation" ("illicit copying"). That is, *copying only trivial aspects of another's work will not result in substantial similarity*; it is only when the copying is sufficiently extensive that infringement occurs.¹⁰⁸

Under the *Lotus* test, however, substantiality is initially an element of copyrightability; the third step of the *Lotus* test is whether any elements which pass the first two steps constitute a substantial part of the work. Although the *Lotus* court essentially applied existing tests of infringement of non-literal elements, it is questionable whether the application of its own test in this particular case is accurate. The critical factor is whether the Lotus 1-2-3 menu command structure is an original and non-obvious manner of expression. While a menu command structure for an electronic spreadsheet can be expressed in a variety of ways, the court specifically noted that "some of [the] specific command terms [in the Lotus 1-2-3 menu system] are quite obvious or merge with the idea of such a particular command term."¹⁰⁹

The court further noted that "[m]ost of the submenus . . . present a list of up to about ten full-word menu choices, presented *in order of predicted frequency of use rather than alphabetically*."¹¹⁰ The court nevertheless concluded that the Lotus 1-2-3 "menu structure, taken as a whole—including the choice of command terms, the structure and order of those terms, their presentation on the screen, and the long prompts—is an aspect of 1-2-3 that . . . meets the requirements of the second element of the [*Lotus* court's] legal test for copyrightability."¹¹¹ It is interesting that the *Lotus* court first broke down the whole Lotus 1-2-3 user interface into separate components and then considered the individual components as a whole.

106. *Nichols v. Universal Pictures Co.*, 45 F.2d 119, 121 (2d Cir. 1930).

107. 843 F.2d 600 (1st Cir. 1988).

108. *Id.* at 608 (emphasis added) (citation omitted).

109. *Lotus*, 740 F. Supp. at 67.

110. *Id.* (emphasis added).

111. *Id.* at 68.

In deciding whether the menu command structure is a substantial part of the Lotus 1-2-3 user interface—the only element of the Lotus 1-2-3 user interface determined to be subject to protection—the *Lotus* court merely stated:

The user interface of 1-2-3 is its most unique element, and is the aspect that has made 1-2-3 so popular. That defendants went to such trouble to copy that element is a testament of its substantiality. Accordingly, evaluation of the third element of the legal test weighs heavily in favor of Lotus.¹¹²

Despite the fact that the *Lotus* court viewed the Lotus 1-2-3 menu command structure, taken as a whole, to be subject matter ripe for protection, the court rejected using a “look and feel” analysis which rests upon considering a work as a whole. The *Lotus* court specifically rejected the “look and feel” analysis because it is conclusory, indicating that courts have “used the concept, not in determining copyrightability, but, apparently assuming copyrightability, in applying the substantial similarity test to determine whether forbidden copying had occurred.”¹¹³

The *Lotus* determination that the Lotus 1-2-3 menu command structure is a substantial part of that user interface appears, itself, to be conclusory. The only support provided for the premise that the menu command structure made Lotus 1-2-3 “so popular” is the fact that the defendants attempted to copy it. Even this conclusion loses some of its support by the court’s own later statements.¹¹⁴

Once the *Lotus* court determined that the Lotus 1-2-3 user interface is copyrightable subject matter, it then turned to actual copying by the defendants. Based upon a reading of the court’s decision, establishment of infringement can be interpreted two ways: (1) defendants’ user interface, taken as a whole, is substantially similar to plaintiff’s user interface, taken as a whole; or (2) defendants’ menu command structure, taken as a whole, is substantially similar to plaintiff’s menu command structure, taken as a whole.

In support of the first interpretation, the *Lotus* court specifically found that, based upon its three-step test, “copyrightability of the *user interface* of 1-2-3 is established.”¹¹⁵ When determining whether there had been copying by the defendants, the court first noted general dissimilarities between the two works, such as the organization of help screens, the greater width of the VP-Planner screen, and the ability of

112. *Id.*

113. *Id.* at 63.

114. Regarding the defendants’ decision that the Lotus 1-2-3 command structure had to be copied in order to ensure the popularity of VP-Planner, the court stated that “[t]o some degree at least, defendants’ premises have proved incorrect in hindsight.” *Id.* at 69. If the Lotus 1-2-3 menu command structure’s substantiality is based upon it being the most unique element of the Lotus 1-2-3 user interface and that which made 1-2-3 so popular, as evidenced by the defendants’ desire to copy it, if that desire is misplaced, is that not evidence that the menu command structure is not quite so substantial after all. As the court also stated, Excel (a competing electronic spreadsheet) achieved commercial success without copying the Lotus 1-2-3 command structure. *Id.*

115. *Id.* at 68 (emphasis added).

VP-Planner to hide certain columns.¹¹⁶ The court then determined that the “works are, nevertheless, substantially, indeed, strikingly, similar.”¹¹⁷ In support of the second interpretation, the *Lotus* court dwelled on the similarities between the two works’ menu command structure: “The court’s comparison of the 1-2-3 menu command hierarchy and the VP-Planner menu hierarchy confirms that VP-Planner ‘has the same command tree’ as 1-2-3—that is, that defendants copied the expression embodied in the 1-2-3 menu hierarchy.”¹¹⁸ The court also noted that “defendants . . . have *admitted* that they copied these elements of protected expression.”¹¹⁹

It is unclear which “elements of protected expression” the defendants copied: the elements of expression, including the source and object codes (contained within the whole of Lotus 1-2-3 which constitute the user interface) or the elements of expression within the Lotus 1-2-3 user interface, namely the menu command structure (which the court determined to be subject to protection). The court concluded its discussion of the defendants’ copying by phrasing the issue as

Does [defendants’ product] have significant features that are substantially similar [to Lotus 1-2-3]? . . . The answer to this question must be “yes.”

Accordingly, I conclude that it is indisputable that defendants have copied substantial *copyrightable elements* of plaintiff’s copyrighted work.¹²⁰

This again begs the question: Which copyrightable elements of Lotus 1-2-3 were defendants guilty of copying—the user interface, taken as a whole, or the menu command structure? If the latter, and had defendants not copied the menu command structure, but copied the remaining substantial, non-protected, elements of the Lotus 1-2-3 user interface, would they have been guilty of infringement?

Another statement by the court implies that it was focusing on the defendants’ copying of the menu command structure:

Moreover, even if some elements of VP-Planner were very different, it would not give defendants a license to copy other substantial elements of 1-2-3 verbatim. If one publishes a 1,000-page book of which only a 10-page segment is an unauthorized reproduction of copyrighted material, and if the 10-page segment is a qualitatively substantial part of the copyrighted work, it is not a defense to a claim of infringement that the book is 99% different from the copyrighted material. Thus,

116. *Id.* at 70.

117. *Id.* In support of its conclusion of substantial similarity despite some differences, the court notes: “[A] laundry list of specific differences . . . will not preclude a finding of infringement where the works are substantially similar in other respects When analyzing two works to determine whether they are substantially similar, courts should be careful not to lose sight of the forest for the trees.” *Id.* (quoting *Atari v. North Amer. Phillips Consumer Elec. Corp.*, 672 F.2d 607, 618 (7th Cir.), *cert. denied*, 459 U.S. 880 (1982)).

118. *Id.*

119. *Id.* at 68.

120. *Id.* at 70 (emphasis added).

defendants' proof that VP-Planner has many features that are different from Lotus 1-2-3 is off point.¹²¹

By the court's holding that the defendants' copying of a "qualitatively substantial part" (the menu command structure of Lotus 1-2-3) was enough to constitute infringement, it could be implied from *Lotus* that if the menu command structure had not been copied, but every other non-literal element within Lotus 1-2-3 had been, then no infringement would have been found.¹²² Such a conclusion is reasonable if it were determined that the Lotus 1-2-3 menu command structure was such a unique and distinctive part of Lotus 1-2-3 that, on its own, it represented to the lay observer the Lotus 1-2-3 product. Very little evidence was presented in the *Lotus* opinion to support the "substantiality" of the Lotus 1-2-3 menu command structure.

As noted previously, the *Lotus* court specifically rejects applying the "look and feel" analysis developed in *Roth Greeting Cards v. United Card Co.*¹²³ In *Lotus*, the court created a more restrictive test by requiring that not only must an element subject to protection be found within the non-literal elements, but it also must be a substantial element. The *Lotus* court may have been mindful of the strong dissent in *Roth* which stated:

I cannot . . . follow the logic of the majority in holding that the uncopyrightable words and the imitated, but not copied art work, constitutes such total composition as to be subject to protection under the copyright laws. The majority concludes that in the overall arrangement of the text, the art work and the association of the art work to the text, the cards were copyrightable and the copyright infringed. This conclusion, as I view it, results in the whole becoming substantially greater than the sum total of its parts.¹²⁴

121. *Id.* (citations omitted).

122. There is case law which supports a finding that copying even relatively minuscule parts of a copyrighted work can constitute infringement. In *Universal City Studios, Inc. v. Kamar Indus., Inc.*, 217 U.S.P.Q. (BNA) 1162 (S.D. Tex. 1982), the court held that exact copying of two lines from a movie could constitute infringement. In this case the defendant began to promote and market certain merchandise, such as drinking mugs and pencil holders, bearing prominent inscriptions in the form of "I E.T." and "E.T. Phone Home!" In finding infringement of the plaintiff's copyrighted movie, the court stated:

The character "E.T." is a central component of [the movie] "E.T. The Extra-Terrestrial." "E.T." is a unique and distinctive character about whom the movie revolves. Plaintiffs contend, and the Court believes, that "E.T." is more than a mere vehicle for telling the story and that "E.T." actually constitutes the story being told. The name "E.T." itself is highly distinctive and is inseparable from the identity of the character. The use of the name "E.T." on Kamar's products inevitably conjures up the image and appeal of the "E.T." character. The Court finds that the average lay observer would recognize readily the "E.T." name as used on Kamar's products as having been taken from the central character of Universal's copyrighted motion picture. This conclusion is reinforced by the fact that the name "E.T." appears on Kamar's products in conjunction with actual lines of dialogue from the movie, and that it is displayed on at least one of Kamar's products in a distinctive style and format similar to that used in connection with Universal's movie.

Universal, 217 U.S.P.Q. (BNA) at 1165.

123. 429 F.2d 1106 (9th Cir. 1970). *Roth* found plaintiff's greeting cards eligible for copyright, even though they were comprised of non-copyrightable elements. *Id.* at 1109.

124. *Id.* at 1111.

There is, however, a strong line of authority which provides copyright protection, under the theory of compilation, to the non-literal elements of a work, even though that work is comprised of non-protectable components.¹²⁵ Indeed, the *Lotus* court noted that the statutory provisions regarding compilation, while not essential to its analysis under its legal test, did reinforce it.¹²⁶ A very important element of granting copyright protection to a work which is composed of non-copyrightable elements is the degree of protection granted to that work.¹²⁷ Where the computer program in question is utilitarian in nature (such as an electronic spreadsheet) as opposed to fictional (such as a computer game), it is logical to conclude that its user interface is more of a "factual" work. This approach, indirectly supported by *Lotus*, would require essentially a verbatim appropriation of the plaintiff's work before infringement could be found. Since the *Lotus* court found a number of dissimilarities between the two computer programs, verbatim copying could not be established. This may be the reason for the *Lotus* court using a test which avoided the ability to establish copyrightability without at least a substantial protectable element.

Despite the difficulties contained within the *Lotus* decision, it still reaches the correct result for the particular facts at hand, if only for public policy reasons. In *Lotus*, the defendants raised a public policy argument that the need to achieve standardization and compatibility within the computer software industry should preclude a finding of copyrightability of the *Lotus* 1-2-3 user interface.¹²⁸ While certain aspects of a computer program's user interface may or may not constitute substantial elements of the work taken as a whole, user interface design is important. Enhancing the user's productivity through better user interface design incorporates the sciences of anatomy, physiology, and psychol-

125. See *Harper House, Inc. v. Thomas Nelson, Inc.*, 889 F.2d 197, 204-05 (9th Cir. 1989) for a summary of cases which hold that a copyrightable compilation can consist mainly or entirely of uncopyrightable elements.

126. *Lotus*, 740 F. Supp. at 67 (noting 17 U.S.C. §§ 101, 103).

127. One consequence of the policy in favor of free use of ideas is that the degree of substantial similarity required to show infringement varies according to the type of work and the ideas expressed in it. Some ideas can be expressed in myriad ways, while others allow only a narrow range of expression. Fictional works generally fall into the first category. The basic idea of a fictional work might be that classic, boy meets girl. This idea can be expressed, as it has been through thousands of years of literature, with infinite variations in setting, sequence of incident, and characterization. An author wishing to write yet another work using the "boy meets girl" idea can choose from a wide range of materials in composing his or her own expression of the idea. Therefore a new work incorporating that idea need not be a verbatim copy or close paraphrase of an earlier work to infringe that work. A resemblance in details of setting, incident, or characterization that falls short of close paraphrase may be enough to establish substantial similarity and infringement. . . .

Factual works are different. Subsequent authors wishing to express the ideas contained in a factual work often can choose from only a narrow range of expression. . . . Therefore, similarity of expression may have to amount to verbatim reproduction or very close paraphrasing before a factual work will be deemed infringed.

Landsberg v. Scrabble Crossword Game Players, Inc., 736 F.2d 485, 488 (9th Cir. 1984) (citations omitted).

128. *Lotus*, 740 F. Supp. at 71.

ogy.¹²⁹ Judge Keeton acknowledged this when he noted the menu-choice commands in the Lotus 1-2-3 user interface were “presented in order of predicted frequency of use rather than alphabetically.”¹³⁰ Commentators within the computer industry have been promoting standardization and compatibility since the first microcomputers were introduced. Attempts to standardize have been driven by user demands to allow different programs to share a common user interface, forming the basis for open systems that share many common attributes, communication protocols, and data formats so that users can “easily interconnect different computers and programs and avoid learning a multitude of user interfaces.”¹³¹

Finally, the cost of learning a software package is substantial. During the early use of a new software package, a user spends considerable time hunting for advice and correcting mistakes. The anticipation of interface angst has inhibited many potential users from employing new software applications that would increase productivity. Even those who take the time to learn to use a software package may choose to master only a small number of its functions.¹³²

The *Lotus* court referred to the defendants’ standardization argument as the “OTSOG (on the shoulders of giants) Principle”: innovation in computer programming is advanced as each programmer builds upon the ideas of previous programmers.¹³³ The fallacy of defendants’ contention is that they specifically discarded any innovations in their own product in order to become a “workalike of 1-2-3.”¹³⁴ Defendants were standing on the shoulders of Lotus not to see further, but to steal market share. Their goal was not to create a new and better electronic spreadsheet but to produce a cheaper Lotus 1-2-3. In this respect, the defendants fit the classic role of “mudball.”

129. Curtis, *Engineering Computer “Look and Feel”: User Interface Technology and Human Factors Engineering*, 30 JURIMETRICS J. 51, 63 (Fall 1989).

130. *Lotus*, 740 F. Supp. at 68.

131. Curtis, *supra* at note 129.

132. *Id.*

133. Defendants’ general contention—that “Progress of Science and useful Arts” cannot occur unless authors and inventors are privileged to build upon earlier progress and earlier innovation—has long been a virtually unchallenged premise in all branches of the law of intellectual property. An early expression of the point is Newton’s declaration: “If I have seen further it is by standing on ye sholders of Giants.” Sir Isaac Newton, Letter to Robert Hooke, February 5, 1675/1676, quoted in R. Merton, *On the Shoulders of Giants: A Shandean Postscript* 31 (1965).

Lotus, 740 F. Supp. at 77. See also *Id.* n.3.

134. It is incontrovertible that, in the process [of making VP-Planner compatible with Lotus 1-2-3, defendants] . . . copied the expressive elements of 1-2-3 that the court has concluded are copyrightable:

[M]aking the changes required for macro compatibility meant that we had to revise existing elements of the [VP-Planner] spreadsheet interface, including the hierarchical menu structure; ensure that keystroke sequences would bring about the same operational result in both programs; add certain functional elements found in Lotus 1-2-3 which VP-Planner did not yet support; and discard certain features which, although beneficial, were inconsistent with the macro compatibility requirement.

Id. at 69 (emphasis added).

While the intent of the infringing party is not a factor in determining liability,¹³⁵ it may be indirectly relevant if it impacts the underlying policy reasons for copyright protection. No one should be rewarded for rejecting new and beneficial creations in order to exploit another's expenditure of time, money, and effort to create an original expression. Any other finding would be in direct contradiction to the policies underlying copyright protection.

While the holding in *Lotus* may reach the correct results for the facts of that particular case, there is concern that the legal test developed in *Lotus*, when applied to a different set of facts, will lead to an incorrect result. Of particular concern is the pending legal actions between Lotus Development Corporation and Borland International, Inc. Borland produces a competing electronic spreadsheet (Quattro Pro) which many critics consider to be superior to Lotus 1-2-3.¹³⁶ There is no question, overall, that Quattro Pro's user interface is substantially different from the Lotus 1-2-3 user interface, except in one respect. The Borland product can be executed in a "Lotus 1-2-3 emulation mode," that is, by specifying a particular command the Borland spreadsheet incorporates the Lotus 1-2-3 menu structure into its own. "Because [Lotus] 1-2-3 is the de facto standard for spreadsheets, any company hoping to achieve significant success must be compatible with it."¹³⁷

Since the *Lotus* court determined that the Lotus 1-2-3 menu command structure is a substantial, protectable part of the Lotus 1-2-3 user interface, there is concern that Quattro Pro, just like VP-Planner, may be found to infringe Lotus 1-2-3. Quattro Pro, however, unlike VP-Planner, was never written to be a "workalike of 1-2-3." This issue is compounded by the fact that the Lotus-compatible menus in Quattro Pro do not even look like the Lotus originals and may even be considered superior.¹³⁸

Lotus only partially addressed this issue. Just as Broderbund Software used a third party's computer program to establish that an underlying idea can be expressed in different ways, Lotus used Microsoft's Excel product to demonstrate a unique user interface for an electronic spreadsheet. The *Lotus* court noted that Excel contained a macro conversion program to convert Lotus 1-2-3 macros to Excel-executable macros, establishing that complete menu compatibility was not required for Lotus 1-2-3 macros to function within a competing product.¹³⁹ The court also indicated that VP-Planner would not have infringed if it had provided an on-line help function that would show users the VP-Planner

135. See generally 3 M. NIMMER, NIMMER ON COPYRIGHT § 13.08 (1990).

136. *New York Times*, July 22, 1990, at F4, col. 1. See generally Seymour, *Victory Spurs Lotus to Take-No-Prisoners Campaign*, PC WEEK, July 16, 1990, at 12.

137. *New York Times*, *supra* note 136 at F4, col. 1.

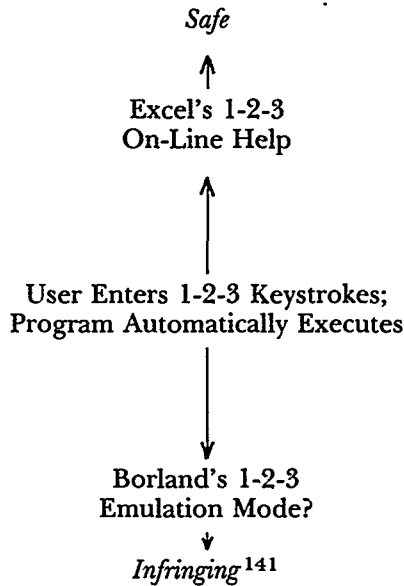
138. "Instead of wandering through a wilderness of one-line menu options with no sign of where you've been, Quattro delivers drop-down windows and daughter windows, and, when necessary, more daughter windows.

You can easily trace the command sequence you've typed—something that eludes (and frustrates) many Lotus users." Seymour, *supra* note 136 at 12.

139. *Lotus*, 740 F. Supp. at 69.

equivalent for 1-2-3 commands, as Excel also does.¹⁴⁰

What the *Lotus* court did not address is how far one can help a user transfer 1-2-3 commands to a competing spreadsheet. Excel does so by having the user type in the sequence of keystrokes that would execute the command were Lotus 1-2-3 being used. Excel then interprets the 1-2-3 commands and presents a screen to the user with instructions on how to execute the same command in Excel. The *Lotus* decision gave no indication of where a software developer may cross the line of infringement in assisting users to adjust from Lotus 1-2-3 to a new product. *Lotus* tells us Microsoft's approach is permitted; Borland's approach may not be. What we do not know is whether an approach in the middle, such as having the user type in the Lotus 1-2-3 command keystrokes and then having the program automatically execute them (without showing the user a Lotus 1-2-3 menu), is permissible. This dilemma can be graphically displayed as follows:



140. *Id.*

141. Microsoft's recently released Excel Version 3.0 actually goes one step further by providing users with the ability to turn on a "Help for Lotus 1-2-3 Users" option which gives users the ability to "watch Microsoft Excel demonstrate . . . equivalent procedure[s] for the Lotus 1-2-3 command [specified]." MICROSOFT GUIDE: USING HELP FOR LOTUS 1-2-3 USERS at 4. When this option is activated (by pressing the "/" (slash) key), Excel lists the Lotus 1-2-3 main menu commands in order of their appearance in Lotus 1-2-3 (though in a column rather than a single row), with submenu commands listed for the particular main menu command highlighted.

The submenu commands are listed in a single row toward the bottom of the screen and change as different main menu commands are highlighted. Excel Version 3.0 users can then select Lotus 1-2-3 commands by pressing the first letter of each command (as in Lotus 1-2-3). Once the commands have been selected, Excel activates its own commands to execute the selected procedure (demonstrating to the user how the particular procedure is executed under Excel), and then executes the selected procedure. In this way, users of Excel Version 3.0 are able to have the entire Lotus 1-2-3 menu structure displayed (though in a slightly different visual representation), can select Lotus 1-2-3

While the *Lotus* decision promotes public policy by preventing a software developer from marketing a product which was designed solely to siphon off market share and not to compete because it offered better tools and options, this decision has the potential of having the completely opposite effect when applied to different circumstances.

V. ANALYSIS OF COMPUTER SOFTWARE COPYRIGHT CASES

The table of cases provided in this section is based on an extensive selection of cases in which the scope of software copyright protection was a major issue. The authors also analyzed several cases in which the issue of software copyrights was a side issue. These less important cases are noted in a footnote to the table of cases. The analysis of this extensive number of cases has resulted in several observations concerning how an attorney can best advise a software developer to avoid copyright infringement when creating new software products.

Each column was chosen for specific reasons. Naturally, the date, consisting of month and year, was a logical starting point as well as the case name and level of court. The "winners circle" was used to identify the true winner, which at times could be either plaintiff or defendant, given the procedural posturing of the case. The "stage of the proceeding" is the next column, and aids in the analysis due to the different burdens of proof. The "type of software" is presented next to show whether the disputes occur over operating system or application software. The critical issues are always questions of fact, and thus the trier of fact has enormous discretion in these cases.

A column indicating the economic relationship between the parties, and whether the relationship was breached, is provided to aid in analyzing any equity concerns. Following this column, "access" is of course needed for all copyright infringement cases, followed by "reasons for copying." Reasons for copying aids in understanding the alleged infringer's rationale for any copying. The amount of work the alleged infringer did and the degree of copying are included due to the subtle influence equitable considerations undoubtedly have on the trier of fact. The "legal conclusions" column assists in showing the development of various legal theories. The "precedent column" is last, and helps show how the cases have become interrelated over time.

menu options in an identical manner as though in *Lotus 1-2-3*, and can then have those commands executed in *Excel*.

DATE	CASE NAME	LEVEL OF COURT	STAGE OF PROCEEDING	WINNER'S CIRCLE	TYPE OF SOFTWARE COPIED	ECONOMIC RELATIONSHIP/ BREACH OF DUTY	ACCESS TO SOFTWARE
8/78	Synercom v. Univ. Computing 462 F. Supp. 1005	Dist. N.D.T	Trial	No infringement by D	Application Input Formats	D Developed S/W pgm that utilized P's data input card formats- Prior Economic Relationship/No Breach	n/a
5/80	Data Cash v. JS&A Group 628 F.2d 1038	App. 7th Cir.	Summary Judgment	No infringement by D	Application Chess Game	No Economic Relationship/ No Breach	Same ROM Chip acq. from same manuf.
1/82	Stern Elec. v. Kaufman 669 F.2d 852	App. 2nd Cir.	Preliminary Injunction	P's valid C/R was infringed	App Game	No Economic Relationship/ No Breach	Access to game User interface
5/82	Williams v. Artic 685 F.2d 870	App. 3d Cir.	Injunctive Order	P's valid C/R was infringed	App-game Op- Audiovisual	No Economic Relationship/ No Breach	Yes
9/82	GCA v. Chance 217 USPQ 718	Dist. N.D. Calif.	Preliminary Injunction	P's valid C/R was infringed	App/ Diagnostic Operating Sys	Former Employees/Breach of Confidence	Yes
2/83	Hubco Data v. Mgmt Assis 219 USPQ 450	Dist. D Idaho	TRO & Prelim Injunction	D's valid C/R was infringed	Operating Sys Utility	No Economic Relationship/ Breach of Licensing Agreement	Yes
8/83	Apple v. Franklin 714 F.2d 1240	App. 3rd Circuit	Preliminary Injunction	P's valid C/R was infringed	Operating sys	No Economic Relationship/ No Breach	Yes
2/84	Apple v. Formula 725 F.2d 521	App. 9th Cir.	Preliminary Injunction	P's valid C/R was infringed	Operating Sys	No Economic Relationship/ No Breach	Yes
3/85	SAS v. S&H 605 F. Supp. 816	Dist. M.D. Tenn	Trial	P's valid C/R was infringed	Application Stat. Analysis	D was Licensee of P/ Breached	Yes
9/85	Williams v. Arndt 626 F. Supp. 571	Dist. D. Mass	Trial	P's valid C/R was infringed	Application Market Trading	P & D under Contract/ Breached	Yes
12/85	Q-CO Ind. v. Hoffman 625 F. Supp. 608	Dist. S.D.N.Y.	Preliminary Injunction	No infringement by D	Application Teleprompter	D was Employee of P/ No Breach	Yes
1/86	Kramer Mfg. v. Andrews 783 F.2d 421	App. 4th Cir.	Trial	P's valid C/R was infringed	Application Game	D was distributor of P/ Breach of distribution arrangement	Yes
8/86	Whelan v. Jaslow 797 F.2d 1222	App. 3rd Cir.	Injunctive Order	P's valid C/R was infringed	Application Dental lab S/W	D hired P to write pgm to manage dental labs. C/R eventually vests in P. D breached relationship and translated pgm to another computer language	Yes

REASON FOR COPYING	AMOUNT OF WORK INFRINGER DID	DEGREE OF COPYING	LEGAL CONCLUSIONS	PRECEDENT
Competitor	indeterminable	Input formats	If order and sequence of data is expression, then it's copyrightable	
Competitor	None	Total	S/W C/R exists in ROM but C/R owner must have C/R notice on chip. Pre-Berne Convention	
Competitor	Minimal	Total copy of game user interface	Repetitive sequence of images is C/R - Players interaction with game doesn't make pgm not fixed	Williams v. Artic; Atari v. Amusement World; Midway v. Drikschneider
Competitor	Minimal	Total	Fixation of S/W in ROM Chip occurred; C/R protection Permitted	Data Cash v. JS&A Group; Stern v. Kaufman; Tandy v. Pers. Micro; Midway V. Drikschneider; Atan v. Phillips; Midway v. Artic
Competitor	Minimal	Total	P distribution was limited, thus no C/R notice required	Tandy v. Pers Micro; Data Cash v. JS&A Group; Kroffit v. McDonald's
Copied D's S/W to Upgrade	Min to upgrade P's Computer	Total	Was the copying of an idea or an expression? Must have subst. similarity of idea and expression	Kroffit v. McDonald's; Williams v. Artic; Apple v. Franklin; Tandy v. Pers Micro
To make D's operating system compatible with P's operating system software	Minimal	Total	Pgms on ROM Chips are C/R. No Distinction between source, object and firm ware	Williams v. Artic; Midway v. Strohon; Data Cash v. JS&A Group; GCA v. Chance; Tandy v. Pers Micro; Apple v. Formula
To make D's operating system compatible with P's operating system software	Minimal	Total	No distinction between source, object and firm ware	Williams v. Artic; Kroffit v. McDonald's; Apple v. Franklin; Atan v. N.Amer. Phillips
D wanted to migrate P's S/W to a new platform	Some	Copied S/W from one platform to another	Pgm developed by D was derivative work	Midway v. Artic; Apple v. Franklin; Williams v. Artic
D allowed translation of P's book to S/W	Minimal	Translated written book	Translation of written book to S/W is infringement	Whelan v. Jaslow (Dist. Ct. Case) Russell v. Price
D created new S/W for new platform	Substantial	Minimal/Ideas only	D only copied P's unprotected ideas	Synercom v. Univ. Computing; SAS v. S&H
Make game substantially similar to P's	Minimal	Total	Audiovisual S/W is C/R; C/R of audiovisual S/W includes underlying pgm; D willful copying infringed P's C/R	Midway v. Bandai; Midway v. Artic; Stern v. Kaufman; Atan v. Phillips; Williams v. Artic; Midway v. Drikschneider; Atari v. Amusement World; Apple v. Franklin; Midway v. Strohon
To translate pgm to a different language for a new market	Translated S/W to a new language	Copied: Structure, Sequence, and Organization	C/R protection extends beyond pgm's literal code to structure, sequence & organization of pgm	Midway v. Strohon; Williams v. Artic; Arnstein v. Porter; Stern V. Kaufman; SAS v. S&H; Apple v. Franklin

DATE	CASE NAME	LEVEL OF COURT	STAGE OF PROCEEDING	WINNER'S CIRCLE	TYPE OF SOFTWARE COPIED	ECONOMIC RELATIONSHIP/ BREACH OF DUTY	ACCESS TO SOFTWARE
9/86	NEC v. INTEL 645 F. Supp. 590	Dist. N.D. Calif.	Trial	D's C/R valid, no infringement by P	Microcode	D licensor of P/No Breach of duty	Yes
10/86	Broderbund v. Unison 648 F. Supp. 1127	Dist. N.D. Calif.	Trial	P's valid C/R was infringed	App. Partnership S/W	D had preliminary agreement with P to translate P's pgm; Negotiations failed, D breached duty to terminate the translation process	Yes
1/87	Plains Cotton v. Goodpasture 807 F.2d 1256	App. - 5th Cir. Prelim Inj	Preliminary Injunction	P's C/R valid, no infringement by D	Application Cotton Futures S/W	Ds were former employees of P/No Breach	Yes
3/87	Frybarger v. IBM 812 F.2d 525	App. 9th Cir.	Summary Judgment	P's C/R valid, no infringement by D	Application Telecomm S/W	Economic Relationship/ No Breach	Yes
3/87	Digital v. Softek 659 F. Supp. 449	Dist. N.D. Georgia	Preliminary Injunction & Permanent	Separate Screen C/R Valid/D infringed	Application Game	No Economic Relationship/ No Breach	To user interface
6/88	Vault Corp. v. Quaid Software 847 F.2d 255	App. 5th Cir.	Preliminary Injunction	No Infringement by D	Application Utility Pgm	No Economic Relationship/ No Breach	Yes
7/88	Pearl v. Competition Elec. 8 U.S.P.Q.2d 1520	Dist. S.D. FL	Declaratory Judgment	D's C/R valid, P infringed	Application Timing Device	No Economic Relationship/ No Breach	Yes
1/89	Manuf. Tech. v. CAMS 706 F. Supp. 984	Dist. D. Conn	Trial	P's valid C/R was infringed	Application CAD/CAM	Economic Relationship/ Breached	Yes
9/89	Telemarketing v. Symantec 12 U.S.P.Q.2d 1991	Dist. N.D. Calif.	Summary Judgment	No infringement by D	Application Outline pgm	Economic Relationship/ No Breach	Yes
9/89	S.O.S. v. Payday 886 F.2d 1081	App. 9th Cir.	Summary Judgment	P's valid C/R was infringed	Application Acctg./Payroll	Economic Relationship/ Breached	Yes
10/89	Johnson Controls v. Phoenix 886 F.2d 1173	App. 9th Cir.	Preliminary Injunction	P's valid C/R was infringed	Application Process Control	Economic Relationship/ Breach	Yes
6/90	Lotus v. Paperback Software 740 F. Supp. 37	Dist. Mass.	Summary Judgment	P's valid C/R was infringed	Application Spreadsheet	No Economic Relationship/ No Breach	Yes

* Cases reviewed but not found useful to analysis:
 Brignoli v. Balch, 645 F. Supp. 1201
 Dynamic v. Planning and Control, 646 F. Supp. 1127
 Ashton-Tate v. Russ, 728 F. Supp. 597
 Apple V. Microsoft, 717 F. Supp. 1428
 Keistall-Whitney v. Mahar, 1990 WL 69013
 ISC-Bunker v. Altech, 1990 WL 103579
 Allen Myland v. IBM, 1990 Corp. L. Dec. P 26,631

REASON FOR COPYING	AMOUNT OF WORK INFRINGER DID	DEGREE OF COPYING	LEGAL CONCLUSIONS	PRECEDENT
No Copying, P developed own Microcode	Substantial	Limited to ideas	Microcode is C/R. Copying is excused when expression merges into idea	Apple v. Formula; Apple v. Franklin
Migrate S/W for different computer uses	Substantial	Some	User interface was protectable expression; P's C/R violated by D's substantially similar interface	Krofft v. McDonald's; Synercom v. Univ. Computing; Whelan v. Jaslow; Arstein v. Porter
Ds developed new competitive	Substantial	D copied ideas of P's pgm	D's structure, sequence and organization determined by market; Idea/Expression merger	Apple v. Franklin; Apple Barrel v. Beard; Synercom v. Univ. Computing; Whelan v. Jaslow
No copying	D developed independent work	None	Ideas were substantially similar but chosen avenues of expression were different	Krofft v. McDonald's; Atari v. Phillips; Atari v. Amusement World
Developed competing product	Devel. clone with same functionality	Identical menus	Substantially similar menu screen infringed C/R on Menu Screen and not C/R on pgm code	Whelan v. Jaslow; Stern v. Kaufman; Midway v. Strohon; SAS v. S&H; Apple v. Franklin; Krofft v. McDonald's; Fox v. MCA; Williams v. Artic
No Copying, D developed pgm to defeat P's anticopying pgm	Substantial	None/only disassemble	P's contractual restriction on disassembly and decompilation of pgm was unenforceable	Atari v. JS & A Group; Midway v. Strohon; Whelan v. Jaslow; Midway v. Artic
Develop competing product	Some	User interface copies	User Interface C/R	Atari v. Phillips; Williams v. Artic; Whelan v. Jaslow; Krofft v. McDonald's; SAS v. S&H; Plains Cotton v. Goodpasture; Synercom v. Univ. Computing; Frybarger v. IBM; Apple Barrel v. Beard, Digital v. Softklone
Develop competing product	Some	User Interface copies	Use interface valid C/R and was infringed	Broderbund v. Unison; Whelan v. Jaslow; Digital v. Softklone; Arstein v. Porter
Develop competing product	D developed both pgms min. work on 2nd	Substantial	Only ideas or licensed expressions copied	Whelan v. Jaslow; Data East v. Epyx
Develop competing product	Minimal	Substantial	D exceeded license thus its copying infringed P's valid C/R	Krofft v. McDonald's; Whelan v. Jaslow
Develop competing product	Minimal	Total	Nonliteral aspects of a computer program are protectable expression	Apple v. Formula; Frybarger v. IBM; Krofft v. McDonald's
Develop competing product	Minimal work to develop user interface	Total	Level of abstractions test refined to a 3 part test	Synercom v. Univ. Computing; Data Cash v. JS&A, Stern v. Kaufman; Apple v. Franklin; Whelan v. Jaslow; NEC v. INTEL; Broderbund v. Unison; Digital v. Softklone; Vault v. Quaid

As seen in *Lotus*, the scope of software copyright protection is always a question of fact. Interestingly, all of the decisions have been decided by judges, rather than juries. Given that judges are making factual decisions concerning highly subjective factors (i.e., Judge Hand's level of abstractions analysis, the *Broderbund* "look and feel" test, and now the *Lotus* three-part test), the relative equity and wholesomeness of the parties will continue to play a critical, but only an implicit, part in the decision process. In addition, sixteen of the twenty-five cases are at the pretrial level where the burden of proof for preliminary injunctions and temporary restraining orders is higher than at trial. In the cases at pretrial, the alleged infringer lost in ten of those sixteen cases.

Although the cases overwhelmingly involved disputes over application software, a few concerned operating system software. Of the operating system software cases, *Apple*¹⁴² and *NEC v. Intel*¹⁴³ are most commonly cited. In *Apple*, the level of copyright protection clearly included total copying of source code as well as code embedded in firmware. *NEC*, the other major operating system software case, found that in operating system software, technical constraints may only permit a few or only one method of writing the program. In these specific areas, the merger of idea and expression will be the dominant issue. In operating system software copyright disputes, the alleged infringer will most likely prevail if it has performed a substantial amount of independent work, can show where idea and expression merge, and can prove it with a clear paper trail.

Application software is the primary battle ground. The scope of copyright protection shows a gradual increase of protection from literal copying to protection of non-literal aspects including the structure, sequence, and organization as in *Whelan*¹⁴⁴ and the user interface as in *Lotus*. As the scope of software copyright is pushed to the edge of its "envelope," the authors have noticed a subtle influence which the issues of breach, amount of the alleged infringer's independent work, and amount of actual copying done by the alleged infringer have on the trier of fact in making the highly subjective factual decision on substantial similarity.

In the fifteen cases in which there was an economic relationship, the authors determined that nine alleged infringers had breached that economic relationship. The classification of economic relationship is somewhat subjective. The authors did not include alleged infringers who had simply purchased a license to mass market software. Rather, the classification of economic relationship is limited to those situations where the copyright holder and the alleged infringer had some type of ongoing economic relationship. Of these alleged infringers who also had breached an economic relationship, all nine were found to have infringed valid copyrights. This unanimous court determination of

142. 714 F.2d 1240 (3d Cir. 1983), *cert. denied*, 464 U.S. 1033 (1984).

143. 645 F. Supp. 590 (N.D. Cal. 1989).

144. 797 F.2d 1222 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987).

infringement is not unexpected to the authors, given the highly subjective and factually intense decision the trier of fact must make, as well as the undoubtedly subtle influence the trier of fact must feel from knowing there was a breach of economic relationship by the alleged infringer.

In all copyright disputes, access and similarity are the two critical elements to prove infringement. Naturally, in each case where infringement was found, access had occurred. The access issue in copying of source code, object code, firmware, and the internal structure, sequence, and organization is critical, and usually has occurred in conjunction with a breach of an economic relationship. In the non-literal element disputes, including the user interface, access can almost be assumed, in that any competent software programmer has undoubtedly seen, and probably even used, the software which is being cloned.

The next three columns each build on the thesis of the authors that the relative equity or wholesomeness of the alleged infringer plays a critical, but subtle, role in the decisional process of the trier of fact. The three columns show the reasons for copying, the amount of work performed by the alleged infringer (including the innovative items added), and the degree of copying. For the alleged infringers in *Lotus*, their goal was to make as identical a copy of the Lotus spreadsheet as possible, and even eliminate several innovations. With the benefit of hindsight, of course, the position of the authors is that an alleged infringer which slavishly copies software will probably be found guilty of infringement, given the inherent subjective process the trier of fact goes through in arriving at the decision of substantial similarity. The amount of work done by the alleged infringer supplements the decision concerning the relative equity and wholesomeness of the alleged infringer.

Of the seven cases where the authors could determine the amount of work done by the alleged infringer was substantial, only one of the alleged infringers, Unison, was found guilty of infringement. All of the rest of the infringers were found not to have infringed. In *Broderbund*, the economic relationship had been terminated, after which the defendant continued to use identical lines of the plaintiff's code in its program. For the degree of copying classification, the authors determined that of the twelve alleged infringers who had substantially copied, ten were found guilty of infringement. The remaining two cases were *Data Cash v. JS & A Group*,¹⁴⁵ which concerned lack of copyright notice and public domain issues, and *Telemarketing v. Symantec*,¹⁴⁶ which involved licensing of copyrightable expression.

The advice to software developers is that good guys almost always win, and mudballs almost always lose. After reviewing these selected cases, counsel advising software developers can greatly influence the

145. 628 F.2d 1038 (7th Cir. 1980).

146. 12 U.S.P.Q.2d (BNA) 1991 (N.D. Cal. 1989).

outcome of any infringement suit by several relatively simple suggestions. In the operating system area, counsel should alert software developers of the idea/expression dichotomy, and have them appropriately document, with contemporaneous memos, why the software developer wrote the operating system software in the manner in which it was written, why certain design decisions were made, and explanations as to why some decision choices were limited to a few or even one option. If the developer is working on a software application which has any similarities to an existing application, counsel should focus on whether any economic relationship exists and ensure that no breach occurs. In addition, counsel should ensure that the software developer is not merely slavishly cloning software as the reason for the development, but rather is seeking to improve the existing product with substantial independent work. With these general guidelines, counsel can steer software developers away from the vast majority of infringement actions.

VI. FUTURE DIRECTIONS

The copyrightability of the literal components of computer software is no longer an issue. The copyrightability of the non-literal aspects, however, remains uncertain. Rather than provide guidance as to what may or may not constitute infringement of non-literal elements of computer programs, the *Lotus* decision has increased the confusion and anxiety within the computer industry. The greatest fear among industry executives is that a legal free-for-all could develop, having a chilling effect on product development. Progress could be retarded due to fear of legal reprisals. Strategic litigation could become an accepted business practice in the computer industry.¹⁴⁷

Many observers in the software field have cried foul after the *Lotus* decision and subsequent threats and actual suits by Lotus against other software developers. The authors believe that these cases have a logical pattern which is evolving into a set of rules and procedures for software developers to follow in developing new software products. Indeed, upon review of the selected cases, *Lotus* was not the major watershed case it first appeared to be, given the earlier *Digital Communications*¹⁴⁸ and *Pearl v. Competition Elec.*¹⁴⁹ cases. The software industry had expected *Lotus* to provide a definite bright line test regarding copyright infringement for non-literal aspects of computer software. The *Lotus* court, however, determined that a bright line test is not possible, given the complex factual issues which must be analyzed in each case. Although the software development industry is still growing, the indus-

147. The founder of Lotus Development Corporation, Mitchell D. Kapor, stated that the "uneasiness of the industry shot up after Lotus sued Borland. . . . Lotus winning the Paperback suit has had an enormous destabilizing effect on the industry. This whole thing is starting to unravel and nobody knows what is going to happen." *New York Times*, *supra* note 136 at F4, col. 1.

148. 659 F. Supp. 449 (N.D. Ga. 1987).

149. 8 U.S.P.Q.2d (BNA) 1520 (S.D. Fla. 1988).

try is beginning to mature and will continue to mature in the 1990s. Given this maturing process, intellectual property law, including copyrights, will play an ever increasing role in the software industry.

Public policy underlying copyright protection dictates that innovation and new expressions be encouraged by appropriate levels of protection. Software developers are certainly permitted to stand on the shoulders of giants in their quest to create new and innovative software. When a software developer discards innovation for the sole purpose of slavishly copying the work of another, public policy requires that this slavish copying be prohibited. The *Lotus* court was correct in its reference to Sir Isaac Newton's observation that innovators must stand on the shoulders of previous giants. The authors would respectfully supplement Newton's observation by adding that on the shoulders of giants, may no mudballs stand.

