# Lotus v. Borland: Confusion within the Computer Industry Affirmed by the Supreme Court

Christopher Kanagawa

Follow this and additional works at: https://digitalcommons.law.utulsa.edu/tlr

Part of the Law Commons

### Recommended Citation

Christopher Kanagawa, *Lotus v. Borland: Confusion within the Computer Industry Affirmed by the Supreme Court*, 32 Tulsa L. J. 633 (1997).

Available at: https://digitalcommons.law.utulsa.edu/tlr/vol32/iss3/13

# *LOTUS v. BORLAND*: CONFUSION WITHIN THE COMPUTER INDUSTRY "AFFIRMED" BY THE SUPREME COURT

## I. INTRODUCTION

Computer technology has developed at an exponential rate. In a similar fashion, evolving computer companies fiercely compete for the consumer market. Most computer companies continue to offer improved variations on the same computer program ideas. Naturally, courts cannot decide cases quickly enough to give attorneys an accurate indication as to what portions of a computer program are copyrightable. Last year, the United States Supreme Court decision of *Lotus Development Corp. v. Borland International, Inc.*[1] did not help draw such a line. In fact, the Court's opinion served to confuse the computer industry. On January 16, 1996, the Supreme Court, split 4-4, issued a one sentence opinion affirming the First Circuit decision. As a result of the 4-4 vote, the circuits remain split.

There are primarily two types of computer programs: operating system programs and application programs.[2] Furthermore, each computer program consists of primarily three levels.[3] The first and lowest level consists of the machine language which is written in "bits."[4] The next level is the "intermediate" level which consists of the object code.[5] Finally, the highest level is the "source code" which consists of the written program language such as FORTAN, COBOL or BASIC.[6] As literal expressions, these levels of a computer program are copyrightable.[7] However, the difficulty arises in determining the copyrightability of the non-literal portions of computer programs.

In *Lotus v. Borland*,[8] the First Circuit made such a determination. This case note will first describe the companies and computer programs at issue and provide a time line of events prior to the suit. Next, it will provide a brief ex-

---

1. 116 S. Ct. 804 (1996), *aff'g* 49 F.3d 807 (1st Cir. 1995).
2. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 43 (D. Mass. 1990) [hereinafter *Paperback*]. In particular, operating systems include programs such as DOS, XENIX and OS/2, which essentially control the basic functions of the computer. *See id.* However, application programs merely permit a user to perform particular functions. *See id.*
3. *See id.* at 43-44.
4. *See id.* at 43.
5. *See id.* at 44. Today, most computer programmers do not write programs using the object code; rather, they write programs utilizing the source code. *See id.*
6. *See id.*
7. *See id.* at 45. *See also* Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1249 (3d Cir. 1983).
8. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807 (1st Cir. 1995) [hereinafter *Lotus*].

planation of copyright law. It will then analyze the First Circuit's decision. Finally, it will describe how computer lawyers have sought to protect their clients in light of the Supreme Court's decision.

## II. THE COMPANIES AND THEIR PRODUCTS

Lotus Development Corporation created the spreadsheet computer program "Lotus 1-2-3."[9] Like any other computer program, it would be useless without some way for the user to control it. Lotus' founder, Mitchell Kapor, initially conceived of a way in which a computer user could efficiently operate Lotus 1-2-3; the conception was Lotus 1-2-3's "menu command hierarchy."[10] The menu command hierarchy Kapor envisioned consisted of numerous "commands."[11] Each command was a word which described to the software user the particular operation the computer could perform.[12] All the commands were then to be arranged in a series of menus and submenus.[13] Ultimately, after much time and effort, Kapor and a team under his direction developed Lotus 1-2-3's menu command hierarchy.[14] Essentially, they designed it in a way they felt was the easiest for the ultimate user to understand.[15]

Lotus even allowed the user to create "macros," which is a way for the user to assign several menu commands to a single keystroke.[16] A macro provides a way for a user to perform repetitive tasks almost effortlessly.[17] In an effort to protect its creation, Lotus copyrighted the program.[18] This copyright included the series of menu commands.[19] Naturally, the easier Lotus 1-2-3 was to operate, the more likely it would be popular with consumers.

---

9. *See id.* at 809.

10. *See* Petitioner's Brief at 9, Lotus Dev. Corp. v. Borland Int'l, Inc., 116 S. Ct. 804 (1996) (No. 94-2003).

11. *See id.*

12. *See* Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F. Supp. 203, 206 (D. Mass 1992) [hereinafter *Borland II*].

13. *See* Petitioner's Brief at 5-6, *Lotus* (No. 94-2003). Specifically, Lotus claims:

> The menus appear in variable series designed by the authors of 1-2-3 to respond to users' choices from previous menus, branching out from the first or "main" menu in what often is called a "menu tree." It is the copyrightability of the overall combination of words and menus in the 1-2-3 menu command hierarchy, viewed as a whole, and not any individual menu command such as "COPY" or "PRINT," viewed in isolation, that is at issue in this case.

*Id.* (citations omitted).

14. *See id.* at 9.

15. *See id.* Specifically, "[t]he menu commands reflected Kapor's subsequent attempt to express those capabilities to users, in full words 'that would intelligently convey to the user the purpose of each command.'" *Id.* Moreover, in creating Lotus 1-2-3's menu command hierarchy, "Kapor and his team spent hundreds of hours over a period of many months considering (and discarding) dozens of organizations of the menu tree, and refining the choice of each word in the menus and the order of those words within each menu." *Id.* Ultimately, the final version consisted of a menu command hierarchy which Kapor described as one which "'was based largely on my intuition and subjective judgment . . . trying as best I could to imagine myself in the role of a typical user.'" *Id.* at 9-10 (alteration in original).

16. *See Lotus*, 49 F.3d at 809.

17. *See id.* at 809-10.

18. *See* Petitioner's Brief at 2, *Lotus* (No. 94-2003).

19. *See id.*

After Lotus 1-2-3 swept the market, Borland International, Inc. released its spreadsheet program "Quattro."[20] Prior to its release, Borland engineers had spent almost three years developing Quattro so it would have a menu structure identical to that of Lotus 1-2-3.[21] Borland specifically wanted to make Quattro "compatible with Lotus 1-2-3 so that spreadsheet users who where already familiar with Lotus 1-2-3 would be able to switch to the Borland programs without having to learn new commands or rewrite their Lotus macros."[22] The ultimate versions of Quattro did contain Lotus 1-2-3's menu command hierarchy.[23]

## III. THE SUIT

### A. Decisions Up To and Including The Supreme Court's Decision

Prior to its suit with Borland, Lotus was involved in a suit with another computer company, Paperback Software International, who produced the spreadsheet program "V-P Planner."[24] In *Paperback*, Lotus claimed that V-P Planner infringed upon Lotus 1-2-3 in that it contained substantially similar elements.[25] On June 28, 1990, "a district court held the Lotus 1-2-3's 'menu structure, taken as a whole—including the choice of command terms [and] the structure and order of those terms,' was protected expression covered by Lotus's copyrights."[26]

The next day, on June 29, 1990, Borland filed a declaratory judgment action against Lotus in the Northern District of California seeking a declaration of non-infringement.[27] Lotus filed suit against Borland in the District of Massachusetts on July 2, 1990.[28] Ultimately, on September 10, 1990, Borland's suit was dismissed in favor of Lotus' suit.[29]

Subsequently, the district court denied both companies' motions for summary judgment.[30] However, from invitation of the district court both sides renewed their motions.[31] In its motion, Borland claimed the Lotus 1-2-3 menus

---

20. *See Lotus,* 49 F.3d at 810.
21. *See id.*
22. *Id.* In both of the spreadsheet programs, Quattro and Quattro Pro, Borland achieved this compatibility by allowing Quattro computer users to use a "Lotus Emulation Interface." *See id.* By activating the interface, the computer screen would display the Lotus menu commands. *See id.* Additionally, the interface would also contain additional menu commands not available for Lotus 1-2-3. *See id.*
23. *See id.* It should be noted that Borland had great success after marketing Quattro and Quattro Pro. Respondent's Brief at 11, *Lotus* (No. 94-2003). Specifically, Borland boasted that "Quattro Pro was first introduced in 1989 and won every major award for spreadsheet excellence given in the software industry." *Id.* Furthermore, Borland contended, "Quattro Pro invariably ranked higher than [Lotus] 1-2-3 in head-to-head reviews and user comparisons, including those conducted by Lotus." *Id.*
24. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37 (D. Mass. 1990).
25. *See id.* at 42.
26. *Lotus,* 49 F.3d at 810 (quoting *Paperback,* 740 F. Supp. at 37, 68, 70) (alteration in original).
27. *See id.*
28. *See id.*
29. *See id.*
30. *See* Lotus Dev. Corp. v. Borland Int'l, Inc., 788 F. Supp 78, 81 (D. Mass 1992) [hereinafter *Borland I*].
31. *See id.* at 82. The district court said it would allow the parties to "focus their arguments more pre-

were uncopyrightable as a matter of law and that no reasonable jury could find such similarity between Quattro and Lotus 1-2-3.[32] Lotus, on the other hand, alleged that Borland infringed on Lotus' copyright.[33] Ultimately, the district court denied Borland's motion and granted Lotus' motion in part.[34] The district court essentially concluded that Borland, as a matter of law, infringed on Lotus' copyright.[35] However, the district court denied Lotus' motion in part, reasoning that the scope of relief hinged on a jury's determination of several factual issues.[36]

Following the district court's decision, Borland appealed.[37] The First Circuit Court of Appeals reviewed only the issue of whether Lotus' menu command hierarchy constituted copyrightable subject matter.[38] The First Circuit reversed the district court by holding Lotus 1-2-3's menu command hierarchy was uncopyrightable subject matter.[39] Lotus then petitioned the Supreme Court for certiorari.[40] The Court granted certiorari and heard oral arguments.[41] With the computer industry watching, the Court issued a one sentence per curiam opinion holding that "[t]he judgment of the United States Court of Appeals for the First Circuit is affirmed by an equally divided Court."[42]

## B. *Effect of the Court's Decision*

The split vote by the Supreme Court has left the computer copyright law in disarray. This confusion can be explained by investigating the precedential effect of the Court's decision. In *Durant v. Essex Co.*,[43] the Court held that an equally divided Supreme Court merely ends the process of direct review and the judgment below remains in force.[44] Therefore, such "affirmance" settles no

---

cisely." *Id.*

32. *Lotus*, 49 F.3d at 810.

33. *See id.* In particular, Lotus asserted Borland had "copied expressive elements of the 1-2-3 interface, including 'menu commands,' 'menu structure,' 'long prompts,' and 'keystroke sequences.'" *Borland II*, 799 F.Supp. at 205.

34. *See Borland II*, 799 F. Supp. at 205.

35. *See id.* at 223. The district court concluded Borland infringed on Lotus' copyright because:

(1) the extent of copying of the "menu commands" and "menu structure" . . . is not *genuinely* disputed in this case, (2) the extent to which the copied elements of the "menu commands" and "menu structure" contain expressive aspects separable from the functions of the "menu commands" and "menu structure," and (3) the scope of those copied expressive aspects as an integral part of Lotus 1-2-3.

*Id.*

36. *See id.*

37. *See Lotus*, 49 F.3d at 812.

38. *See id.* at 809.

39. *See id.* at 819.

40. *See* Lotus Dev. Corp. v. Borland Int'l, Inc., 116 S. Ct. 39 (1995).

41. *See* Lotus Dev. Corp. v. Borland Int'l, Inc., 116 S. Ct. 561 (1995).

42. Lotus Dev. Corp. v. Borland Int'l, Inc., 116 S. Ct. 804, 805 (1996). Justice Stevens took no part in the consideration or decision of the case. *See id.*

43. 74 U.S. (7 Wall.) 107 (1868).

44. *See id.* at 112. Specifically, the Court stated:

If the judges are divided, the reversal cannot be had, for no order can be made. The judgment of the court below, therefore, stands in full force. It is, indeed, the settled practice in such case to enter a judgment of affirmance; but this is only the most convenient mode of expressing the fact that the cause is finally disposed of in conformity with the action of the court below, and that court can proceed to enforce its judgment. The legal effect would be the same if the appeal, or writ of error, were

issue of law and is not entitled to precedential value. Similar to *Durant,* it is clear that the Supreme Court's decision in *Lotus,* which was split 4-4, has no precedential effect. Consequently, the First Circuit decision remains mandatory authority within the First Circuit and only persuasive authority in the other 12 Districts.

To further add to the confusion, it is clear that by granting certiorari, the Supreme Court found the issue to be important. It seems almost inevitable that the Court will make a definitive ruling in the future. Therefore, the First Circuit's decision offers little guidance for an attorney who seeks long term solutions to computer copyrightability problems. Equally confusing, and important to this paper, is the way the First Circuit arrives at its holding that Lotus' menu command hierarchy is uncopyrightable.

## IV. COPYRIGHT LAW

To fully understand the First Circuit's decision, one must understand some of the basic concepts related to copyright law. Copyright law originates from Article 1, Section 8, Clause 8 of the Constitution. It says Congress has the power "[t]o promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."[45]

### A. *The Expression and Idea Dichotomy*

One important aspect of copyright law is the distinction between an expression of an idea and the idea itself. While the former is copyrightable, the latter is not.[46] The classic case involving this expression/idea distinction is *Baker v. Selden.*[47] In *Baker,* the plaintiff, Selden, wrote and copyrighted a book which explained a peculiar way of book-keeping.[48] Included within the book was an explanation of the accounting system, as well as forms and ta-

---

dismissed.
*Id. See also* Neil v. Biggers, 409 U.S. 188 (1972).

45. U.S. CONST. art. I, § 8, cl. 8.

46. Another crucial principle in determining whether a work is copyrightable is the distinction between utilitarian and non-utilitarian aspect of the work. This principle is demonstrated in Mazer v. Stein, 347 U.S. 201 (1954). In *Mazer,* the plaintiff held a copyright for a sculpture in the form of a human figure. *See id.* at 202. Subsequently, the plaintiff mass-produced the statuettes as lamp bases. *See id.* at 203. Without authorization, the defendants copied the statuettes, embodied them in lamps and sold them. *See id.* Later, the plaintiffs sued, claiming the defendants infringed on a valid copyright. *See id.* at 203-04.

In response, the defendants claimed that the lamps were purely utilitarian objects; thus, they were not copyrightable. *See id.* at 205-06. The defendants also argued that because the plaintiff could have received patent protection, the plaintiff should not receive copyright protection. *See id.* at 215. The Court then recited the development of copyright legislation. *See id.* at 208. Ultimately, the Court held that although the lamps were functional and clearly utilitarian, they also deserve copyright protection. *See id.* at 219.

This principle was only briefly discussed in *Lotus.* The court essentially brushed this possible argument aside, just as it did for the idea/expression distinction, reasoning that it did not apply because the menu command hierarchy represents a "method of operation." *See Lotus,* 49 F.3d at 817.

47. 101 U.S. 99 (1879).

48. *See id.* at 100.

bles.[49] Selden alleged that Baker infringed on his copyright by developing a system which was similar to Selden's; however, Baker's system used a different arrangement of columns and different headings.[50] The Court reasoned that while the description of the accounting system was entitled to the benefit of copyright as art, the actual accounting system was not, as it represented only an idea.[51] Ultimately, the Court held that Selden's copyright did not give him an exclusive right to the accounting system.[52]

## B. *The Doctrine of Merger*

In some instances, distinguishing between the expression of an idea and the idea itself becomes difficult. Consequently, the doctrine of merger has evolved.[53] The First Circuit Court of Appeals explained this doctrine in *Morrissey v. Procter & Gamble Co.*[54] In *Morrissey*, the plaintiff held a copyright to a set of rules for a contest of the "sweepstakes" type.[55] The plaintiff sued for copyright infringement after Proctor & Gamble copied one of the contest rules.[56] The appellate court held that the rules of the contest were not copyrightable.[57] Specifically, the court said:

> When the uncopyrightable subject matter is very narrow, so that "the topic necessarily requires," . . . if not only one form of expression, at best only a limited number, to permit copyrighting would mean that a party or parties, by copyrighting a mere handful of forms, could exhaust all possibilities of future use of the substance. In such circumstances it does not seem accurate to say that any particular form of expression comes from the subject matter. However, it is necessary to say that the subject matter would be appropriated by permitting the copyrighting of its expression.[58]

Consequently, when the expression of an idea is so closely linked to the idea itself and there is only a limited number of forms of expression, the expression merges with the idea, making it uncopyrightable.

---

49. *See id.*
50. *See id.*
51. *See id.* at 104-05.
52. *See id.* at 107.
53. Moreover, the doctrine of merger also applies when the utility and non-utility are inseparable.
54. 379 F.2d 675 (1st Cir. 1967).
55. *See id.* at 676.
56. *See id.*
57. *See id.* at 678.
58. *Id.* at 678-79.

## C.  Federal Copyright Law

Congress enacted the most recent copyright law in 1976.[59] Subsequently, Congress responded to recommendations of the National Commission of New Technological Uses of Copyrighted Works (CONTU) and included the definition of computer programs.[60] Computer programs are defined as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result."[61] Moreover, Congress also amended section 117 to specifically include the copyrightability of computer programs.[62] In its report, CONTU never suggests that copyright protection be limited to the literal code; rather, it addresses the dichotomy between idea and expression.[63] CONTU's report suggests that courts are better able to distinguish between expression and idea rather than adopt a *per se* rule in copyright cases.[64]

---

59. *See* 17 U.S.C. §§ 101-1101 (1994). The subject matter of copyright is covered in section 102. It reads:

> (a) Copyright protection subsists, in accordance with this title, in original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device. Works of authorship include the following categories:
>> (1) literary works;
>> (2) musical works, including any accompanying words;
>> (3) dramatic works, including any accompanying music;
>> (4) pantomimes and choreographic works;
>> (5) pictorial, graphic, and sculptural works;
>> (6) motion picture and other audiovisual works;
>> (7) sound recordings; and
>> (8) architectural works.
> (b) In 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.

17 U.S.C. § 102.

60. *See* Whelan Assoc's v. Jaslow Dental Lab., 797 F.2d 1222, 1240-41 (3d Cir. 1986).

61. 17 U.S.C § 101.

62. *See Whelan*, 797 F.2d at 1241. Section 117 now reads:

> Notwithstanding the provisions of section 106, it is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:
>> (1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or
>> (2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.
> Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

17 U.S.C. § 117.

63. *See Whelan*, 797 F.2d at 1241.

64. *See id.*

## D. *The Problems Associated with Copyrighting Computer Programs*

Although section 102 covers copyright protection of computer programs, it is clear that there exists some difficulty in classifying the gambit of computer program elements. One element includes the written code of the computer.[65] It represents the computer programmer's written work.[66] Almost like words in a book, it clearly can be protected by copyright.[67] However, the other portions of the computer program are not as easily categorized as copyrightable. For example, one portion of a computer program embodies the entire function of the program, like a word processor which enables a user to type letters or write papers. Clearly, this level of the computer programs only represents an idea, undeserving of copyright protection.[68] The trouble with copyright protection for computer programs consists of the elements between the computer code and the idea behind the computer program. Such elements include the program's "screen output," or in the case of *Lotus,* the "user interface." These elements require a much more in-depth analysis to determine whether they should be offered copyright protection. Consequently, courts have developed several tests to provide such analysis.

## E. *Case Law Prior to the First Circuit's Decision*

The various judicial tests have developed in light of Congress' failure to provide much guidance concerning the copyrightability of non-literal elements of computer programs. The Third Circuit Court of Appeals was one of the first courts to develop such a test in *Whelan Associates v. Jaslow Dental Laboratory.*[69] *Whelan* involved the alleged copyright infringement of the computer program Dentalab which was used in a dental laboratory.[70] Ultimately, the trial court concluded that the defendant's competing computer program Dentacom infringed on the copyright of Dentalab.[71] On appeal, the defendant argued that the structure of the computer program consisted of the idea of the program; thus, it cannot be afforded copyright protection.[72] The appellate court, however, disagreed holding that a computer program's structure, a non-literal element of the computer program, entails part of its expression.[73]

Initially, the *Whelan* court briefly described the technical background behind computer programs.[74] The court concluded the coding process, a literal

---

65. *See supra* notes 2-7 and accompanying text.
66. *See supra* notes 2-7 and accompanying text.
67. *See supra* notes 2-7 and accompanying text.
68. *See Borland II,* 799 F. Supp. at 217.
69. 797 F.2d 1222, 1241 (3d Cir. 1986).
70. *See id.* at 1226.
71. *See id.* at 1228.
72. *See id.* at 1235.
73. *See id.* at 1239-40.
74. *See id.* at 1229-31.

element of the computer program, consisted of a small part of programming.[75] The court also analyzed section 102 and its legislative history.[76] The appellate court held that copyright protection extends beyond the literal computer code.[77] Ultimately, the court concluded that there was enough evidence for the trial court to find that the computer programs were substantially similar to each other.[78]

Several years later, the Second Circuit examined the same issue in *Computer Associates International, Inc. v. Altai.*[79] The *Altai* court reasoned that the *Whelan* court's opinion was based on "outdated appreciation of computer science."[80] Specifically, the *Altai* court determined that the *Whelan* decision relied "too heavily on metaphysical distinctions" and not enough on practical considerations.[81] Ultimately, the Second Circuit devised a three-step procedure to determine if the non-literal elements of competing computer programs are substantially similar.[82]

The test the *Altai* court used is known as the "abstractions" test.[83] In the first step of the "abstractions" test within the context of computer programs, the abstraction step, a court should dissect the computer program and then isolate each level of abstraction.[84] This procedure is similar to reverse engineering in that the court must retrace and map each of the computer programmer's steps in designing the computer program.[85] Consequently, the court must classify several levels of abstraction which range from the computer source code to an articulation of the program's ultimate function.[86]

In the second step, the filtration step, a court filters the different levels of abstraction to separate the protectable expressions from non-protectable ones.[87] Essentially, a court must determine why the computer programmer inserted that particular level of abstraction at that level.[88] If the court determines that the

---

75. *See id.* at 1231.

76. *See id.* at 1233-34.

77. *See id.* at 1248.

78. *See id.* The court discussed the relevant evidence showing substantial similarity between Dentalab and Dentacom. *See id.* at 1242-48. The court said:

Because all steps of a computer program are not of equal importance, the relevant inquiry cannot therefore be the purely mechanical one of whether most of the programs' steps are similar. Rather, because we are concerned with the overall similarities between the programs, we must ask whether the most significant steps of the programs are similar.

*Id.* at 1246.

79. 982 F.2d 693 (2d Cir. 1992).

80. *Id.* at 706.

81. *Id.* The court also said that its decision merely applies the traditional analysis of copyright to the new technology of computer programs. *See id.* Moreover, the court recognized the disparaging difference between the speed of computer technology advancement and the speed at with courts decide cases. *See id.*

82. *See id.* at 706-12.

83. *See id.* Judge Learned Hand created the "abstractions" test for separating idea from expression under copyright law. *See id.* at 706 (citing Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930)). *See also* Autoskill v. National Educ. Support Sys., 994 F.2d 1476, 1491 (10th Cir. 1993).

84. *See Altai,* 982 F.2d at 706.

85. *See id.* at 707.

86. *See id.*

87. *See id.* at 707-10.

88. *See id.* at 707.

level of abstraction was included because it was an "idea" or was dictated by efficiency considerations and, thus, merely incidental to the idea, then it is unprotectable expression.[89] Similarly, if the level of abstraction was inserted because of "factors external to the program itself; or taken from public domain," it is also unprotectable.[90]

Finally, in the comparison step, a court must determine whether the alleged infringer copied any of the protectable levels of abstraction.[91] Here the court uses a substantial similarity inquiry between the protectable levels of abstraction of the copyrighted program and that of the alleged infringer's program.[92] Only after applying the three step test can a court then adequately determine whether a computer program's non-literal elements have been copied by another programmer.

## V. THE FIRST CIRCUIT'S DECISION

In *Lotus*, the First Circuit began its opinion by describing the menu command hierarchy, the companies, and the facts leading up to the review.[93] The court then described the sole issue in the case: whether Lotus 1-2-3's menu command hierarchy is uncopyrightable as a system, method of operation, process, or procedure making it unprotected by federal copyright statute section 102(b).[94]

The *Lotus* court then described the test that a plaintiff must satisfy in order to prove copyright infringement.[95] It specifically discussed the two prong test developed in *Feist Publications, Inc. v. Rural Telephone Service Co.*[96] The court said that "a plaintiff must prove '(1) ownership of a valid copyright, and (2) copying of constituent elements of the work that are original.'"[97] It went on to describe that in order for a plaintiff to satisfy *Feist's* first prong, it must

89. *See id.* In this step, the Second Circuit reasoned that under traditional copyright concepts expressions which were merely incidental to the idea being expressed, merge with the underlying idea. *See id.* (citing Morrissey v. Proctor & Gamble Co., 379 F.2d 675, 678-79 (1st Cir. 1967)). Furthermore, the court stated several reasons why the merger doctrine applied to computer programs as it did to other copyrightable materials. First, it determined that CONTU recognized the doctrine's applicability to computer programs. *See id.* at 708. It also recognized that the more computer programmers strive to create the most efficient programs, the closer such designs approximate the idea or process embodied in the structure. *See id.* Finally, it considered "a program's essentially utilitarian nature and competitive forces that exist in the software marketplace." *Id.*

90. *Id.* at 707. With respect to external factors, the court reasoned that programmers do not have the absolute freedom of design choice. *See id.* at 709. Specifically, programmers choices are limited "by extrinsic considerations such as (1) mechanical specifications of the computer; . . . (2) compatibility requirements of other programs with which a program is designed to operate in conjunction; (3) computer manufacturers' design standards; (4) demands of the industry; . . . and (5) widely accepted programming practices within the computer industry." *Id.* at 709-10.

91. *See id.* at 710-12. The court refers to the remaining protectable expression left over after filtration and comparison as the "golden nugget." *See id.* at 710.

92. *See id.*

93. *See Lotus*, 49 F.3d at 809-12.

94. *See id.* at 812.

95. *See id.* at 813.

96. 499 U.S. 340, 361 (1991).

97. *Lotus*, 49 F.3d at 813 (quoting *Feist*, 499 U.S. at 361).

prove that the work as a whole is original and that it complied with applicable statutory formalities.[98] Moreover, having a certificate of copyright registration shifts the burden of proof to the defendant as it "'constitutes *prima facia* evidence of copyrightability.'"[99] While the First Circuit briefly described how one must satisfy *Feist's* second prong, it only had to determine whether Lotus' menu command hierarchy was copyrightable because Borland conceded that Lotus had a valid copyright.[100] The court recognized that the outcome of most copyright infringement cases hinges on whether there exists actual infringement.[101] Consequently, it noted that earlier copyright infringement cases provided little value in determining *Lotus*.[102]

After recognizing that the case was essentially one of first impression, the court examined two different cases.[103] Borland urged the court to apply *Baker v. Selden*,[104] arguing that the only difference between the facts and arguments of *Baker* and *Lotus* was the "user interface."[105] Unconvinced, the court found *Baker* inapplicable.[106] It specifically stated that "this appeal involves Lotus' monopoly over the commands it uses to operate the computer."[107]

The *Lotus* court also distinguished *Altai* from the case before it.[108] The court noted that *Altai* developed a test to deal with the fact that computer programs, copyrighted as literary works, can be copied non-literally.[109] The First Circuit recognized that when dealing with a non-literal copying case, a court must determine whether similarities exist because the works share the same underlying idea or whether the similarities are a result of the second author copying the original author's expression.[110] The court noted that the *Altai* test was developed for such an inquiry, in the context of computer programs.[111] However, the First Circuit reasoned that the *Altai* test would provide little help in determining whether the literal copying of Lotus' menu command hierarchy constituted copyright infringement.[112] Moreover, it characterized the *Altai* test as misleading in this context because "instructing courts to abstract the various levels . . . seems to encourage them to find a base [or literal] level that includes copyrightable subject matter that, if literally copied, would make the copier liable for copyright infringement."[113] This type of analysis would obscure the

---

98. *See id.*

99. *Id.* (quoting Bibero Systems, Inc. v. Colwell Systems, Inc., 893 F.2d 1104, 1106 (9th Cir. 1990)).

100. *See id.*

101. *See id.*

102. *See id.*

103. *See id.* at 813-14.

104. 101 U.S. 99 (1879). *See also supra* text accompanying notes 47-52.

105. *See Lotus*, 49 F.3d at 813-14. *See also supra* text accompanying notes 47-52.

106. *See Lotus*, 49 F.3d at 814.

107. *Id.*

108. *See id.* at 814-15.

109. *See id.* at 814 (citing *Altai*, 982 F.2d at 706). Non-literal copying "is copying that is paraphrased or loosely paraphrased rather than word for word." *Id.*

110. *See id.*

111. *See id. See also supra* notes 83-92 and accompanying text.

112. *See Lotus*, 49 F.3d at 815.

113. *Id.*

*TULSA LAW JOURNAL* [Vol. 32:633]

true question of whether the menu command hierarchy as a whole can be copyrighted.[114] Ultimately, the First Circuit refused to apply the *Altai* test to the Lotus suit.[115] In doing so, it avoided confronting several of the district court's findings.[116]

Without any useful precedent to guide it, the First Circuit went on to discuss its ultimate finding: Lotus' menu command hierarchy constituted a method of operation.[117] Applying section 102,[118] the court determined whether the menu command hierarchy was copyrightable by first asking if the menu command hierarchy was a "method of operation" as described in section 102(b).[119] If so, the court reasoned it would be irrelevant whether the menu command hierarchy fit into one of the categories described in section 102(a).[120]

Beginning its discussion, the *Lotus* court conceded that the actual computer code underlying the menu command was copyrightable.[121] However, it noted that code copying was not at issue; Borland emulated Lotus 1-2-3's menu com-

---

114. *See id.* The court stated that "[t]he initial inquiry should not be whether individual components of a menu command hierarchy are expressive, but rather whether the menu command hierarchy as a whole can be copyrighted." *Id.*

115. *See id.* While the district court did not literally use the *Altai* test, it used one that was similar. *See Borland II,* 799 F. Supp. at 211-12. In *Borland II,* the district court determined the first two parts of its test paralleled the abstraction and filtration steps of the *Altai* test. *See id.* at 211. Furthermore, the third part of the district court's test was "compatible substantively though different in methodology" with the "comparison" portion of the *Altai* test. *See id.* at 212.

116. The district court made three conclusions using its test which was similar to that of the *Altai* test. *See Borland II,* 799 F. Supp at 216-19. First, the district court analyzed the conception of Lotus 1-2-3's user interface (similar to the abstraction step of the *Altai* test). *See id.* at 216. Ultimately, it determined that the "idea" or "system" of the Lotus 1-2-3 user interface:

> involves a system of menus, each menu consisting of less that a dozen commands, arranged hierarchically, forming a tree in which the main menu is the root/trunk of the tree and submenus branch off from higher menus, each submenu being linked to a higher menu by operation of a command, so that all the specific spreadsheet operations available in Lotus 1-2-3 are accessible through the paths of the menu command hierarchy.

*Id.*

Second, the district court determined the menu command hierarchy contained identifiable elements of expression (similar to the filtration step of the *Altai* test). *See id.* at 217. It identified that "[a] very satisfactory spreadsheet menu tree can be constructed using different commands and a different command structure from those of Lotus 1-2-3." *Id.* It found particularly persuasive the fact that Borland indeed did construct such an alternative tree for use in Quattro Pro. *See id.*

Finally, the district court looked at the expressive elements from step two in order to see if such elements taken together consist of a substantial part Lotus 1-2-3's user interface (similar to the comparison step of the *Altai* test). *See id.* at 219. Here, the court determined that the expressive elements of the menu command hierarchy consisted of a substantial part of Lotus 1-2-3. *See id.*

117. *See Lotus,* 49 F.3d at 815-19.

118. *See supra* note 59.

119. *See Lotus,* 49 F.3d at 816.

120. *See id.*

121. *See id.* They also briefly discussed the copyrightability of Lotus 1-2-3 "long prompts." *See id.* at 815-16 n.9. In the menu command of Lotus 1-2-3, the computer user sees a two line menu. The top line consists of the command, while the bottom line displays the long prompt. It is similar to help text, describing what the command will do. *See id.* at 811 n.2. On one hand the court expressly states that it takes no position on whether the "long prompt" can be copyrighted. *See id.* at 815-16 n.9 (citing Morrissey v. Proctor & Gamble Co. 379 F.2d 675, 678-79 (1st Cir. 1967)). On the 'other hand, in the same footnote the court stated that a strong argument could be made that the long prompt's explanation merges with the idea of explaining such function. *See id.* In citing *Morrissey,* the court explained that when there are only limited number of ways to express an idea, the expression merges with the idea and is uncopyrightable. *See id.* The court also expressly stated that it would take no position as to whether the Lotus 1-2-3 screen display consists of an original expression which is capable of being copyrighted. *See id.* at 816 n.10.

mand hierarchy without copying any of Lotus 1-2-3's code.[122] Moreover, the First Circuit reasoned that the phrase "method of operation" within section 102(b) "refers to the means by which a person operates something."[123] Ultimately, it held that Lotus 1-2-3's menu command hierarchy constituted an unprotectable "method of operation."[124]

In reaching its holding, the First Circuit also refuted some of the district court's conclusions.[125] Specifically, the district court found that the particular arrangement of the menu command terms constituted an expression of an idea.[126] The court of appeals disagreed, reasoning that "methods of operation" are not limited to abstractions; instead they are "the means by which a user operates something."[127] Furthermore, if specific words are essential to an operation, they are part of the "method of operation" and are unprotectable.[128]

In an attempt to bolster its finding that "methods of operation" are not limited to abstractions, the First Circuit sought support from *Baker*.[129] In *Baker*, the Supreme Court held that while the expression of an idea could be protected by copyright, the actual idea could not be protected.[130] *Baker* also described the object of the expression as explanation and the object of the idea as use.[131] In terms of Lotus 1-2-3, the First Circuit reasoned that Lotus wrote its menu command hierarchy so that people could learn it and use it.[132] Consequently, the court said that it "falls squarely within the prohibition on copyright protection established in *Baker v. Seldon* and codified by Congress in § 102(b)."[133]

In describing how the menu command hierarchy represented a "method of operation," the *Lotus* court first compared the menu commands to buttons on a video cassette recorder ("VCR").[134] It emphasized that Lotus 1-2-3 without its menu commands, just like a VCR without buttons, would be useless.[135] Consequently, the court reasoned that how the buttons are labeled and arranged does not make them an "expression" of the abstract "method of operation."[136]

---

122. *See id.* at 816.
123. *Id.* at 815.
124. *See id.*
125. *See id.* at 816.
126. *See id.*
127. *Id.*
128. *See id.*
129. *See id.*
130. *See* Baker v. Selden, 101 U.S. 99, 105 (1879).
131. *See id.*
132. *See Lotus*, 49 F.3d at 817.
133. *Id.*
134. *See id.*
135. *See id.*
136. *See id.* At this point the court also refuted any argument that the buttons of VCRs differ from the command terms used in operating a computer program. *See id.* It reasoned that even though a VCR may fit into the "sculptural works" category in 17 U.S.C. § 102(a), it must furthermore fit the "useful-article exception." Specifically, to be copyrightable a sculptural work must be able to be identified separately from and are capable of existing independently of, the utilitarian aspects of the article. *See id.* (citing 17 U.S.C. § 101). Furthermore, the court said that whatever expression exists from the arrangement of the VCR buttons is irrelevant; the buttons cannot exist separately from the VCR. *See id.* Consequently, a VCR cannot be copyrighted. *See id.* Finally, the court concluded that VCR buttons are similar to the command terms of Lotus 1-2-3, and

The First Circuit also described the program's compatibility with similar programs.[137] The court found it "absurd" that a user familiar with Lotus 1-2-3 would have to learn an entirely new method of operation when switching to a similar program.[138] It said:

> The fact that there may be many different ways to operate a computer program, or even many different ways to operate a computer program using a set of hierarchically arranged command terms, does not make the actual method of operation chosen copyrightable; it still functions as a method for operating the computer and as such is uncopyrightable.[139]

The *Lotus* court then discussed its view that an expression which is part of a "method of operation" cannot be copyrighted.[140] The court interpreted the Supreme Court decision of *Feist* as holding that not all expressions are necessarily copyrightable; but rather, a court must determine whether the expression is uncopyrightable as defined in section 102(b).[141]

The court also noted that copyright encourages others to build freely upon the ideas and information conveyed by a work and that Borland was free to build on the method of operation Lotus designed.[142] It explained that in most contexts, however, it is unnecessary to "build" on another's expression because such an expression can be conveyed in a different manner.[143] The court determined that in the context of methods of operation, "building" requires dismantling also.[144]

Finally, the First Circuit conceded that its opinion contradicted courts from other circuits.[145] It first discussed *Autoskill, Inc. v. National Educational Support Systems, Inc.*[146] In *Autoskill*, the Tenth Circuit Court of Appeals analyzed a similar copyright infringement case. While the defendant argued that the keying procedure used in a computer program constituted an uncopyrightable "procedure" or "method of operation," the Tenth Circuit disagreed.[147] The *Lotus* court recognized that the *Autoskill* court held that courts must go beyond the literal language of section 102(b) and distinguish the idea from the expres-

---

therefore, the command terms cannot be copyrighted. *See id.*

137. *See id.* at 817-18.
138. *See id.* at 818.
139. *Id.*
140. *See id.*
141. *See id.* The First Circuit explained:
   We do not think that the Court's statement that "copyright assures authors the right to their original expression" indicates that all expression is necessarily copyrightable; while original expression is necessary for copyright protection, we do not think that it is alone sufficient. Courts must still inquire whether original expression falls within one of the categories foreclosed from copyright protection by § 102(b), such as being a "method of operation."
*Id.* (quoting Feist Publications, Inc. v. Rural Tel. Serv. Co., 499 U.S. 340, 349-50 (1991)).
142. *See id.*
143. *See id.* The court again makes reference to the doctrine of "merger" of the idea and the expression when there are only a limited number of ways to express an idea. *See id.* n.13.
144. *See id.*
145. *See id.*
146. 994 F.2d 1476 (10th Cir. 1993).
147. *See id.* at 1495 n.23.

sion to resolve the issue.[148] Furthermore, the court in *Lotus* admitted that the dicta in *Brown Bag Software v. Symantic Corp.*[149] also contradicted the *Lotus* holding.[150] Specifically the *Lotus* court disagreed with the *Brown Bag Software* court's holding that "menus and keystrokes" may be copyrightable.[151]

## VI. ANALYSIS

The First Circuit's decision to classify the menu command hierarchy as a method of operation allowed the court to avoid investigating the idea/expression dichotomy. While the First Circuit's reasoning simplifies the required analysis, it does not adequately reflect Congress' intent for copyright protection. In fact, its decision contradicts Congress' intent in enacting section 102(b). Moreover, the court's reasoning seems to contradict itself. It also fails to follow pertinent case law. Finally, the policy arguments fail to justify contradicting Congress' intent in enacting section 102 or prior case law. Therefore, while the First Circuit's decision simplified the necessary analysis, it was incorrect.

First, the First Circuit disregarded the legislative history leading to the enactment of section 102. The legislative history behind section 102(b) in its original 1976 version undisputedly indicates Congress' intent in passing the statute.[152] Specifically, Congress stated that it did not intend section 102(b) to enlarge or contract the scope of copyright protection, but rather, to codify the basic idea/expression dichotomy.[153] Consequently, the First Circuit's all-inclusive classification of Lotus 1-2-3's menu command hierarchy as a "method of operation" is incorrect. Instead, the court should have first analyzed the menu command hierarchy in light of the idea/expression dichotomy. Only then should it have determined what portion of the computer program consists of its "method of operation."

The First Circuit's decision also contradicts itself. As stated above, the court first determined that the menu command hierarchy could not receive copyright protection because it was a "method of operation."[154] Moreover, the court explicitly stated that once it concluded it was a "method of operation" it was irrelevant whether any portion of that "method of operation" contains the

---

148. *See Lotus*, 49 F.3d at 818-19.
149. 960 F.2d 1465, 1477 (9th Cir. 1992).
150. *Lotus*, 49 F.3d at 819 n.14.
151. *See id.* (citing *Brown Bag Software*, 960 F.2d at 1477).
152. *See* H.R. REP. NO. 94-1476, at 57, *reprinted in* 1976 U.S.C.C.A.N. 5659, 5670.
153. *See id.* The exact language of Pub. L. No. 94-553 is as follows:

    Some concern has been expressed lest copyright in computer programs should extend protection to the methodology or processes adopted by the programmer, rather than merely to the "writing" expressing his ideas. Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law.

    Section 102(b) in no way enlarges or contracts the scope of copyright protection under the present law. Its purpose is to restate, in the context of the new single Federal system of copyright, that the basic dichotomy between expression and idea remains unchanged.

*Id.*

154. *See Lotus*, 49 F.3d at 815.

programmer's expression.[155] As such, the court essentially determined that portions of the menu command hierarchy did in fact merge with the uncopyrightable elements of the computer program.

The problem with classifying all portions of the menu command hierarchy as a "method of operation" is that it would lead to incorrect copyright results. Explicitly, the First Circuit acknowledges that the program code of Lotus 1-2-3 is copyrightable.[156] In differentiating between the program code and the menu command hierarchy, the First Circuit's decision viewed the menu command hierarchy as essential to the operation of the program.[157] However, it is also true that the program code becomes essential to the operation of the computer programs.[158] As such, the First Circuit's decision fails to explain the difference between the computer code (a copyrightable element of the computer program) and the menu command hierarchy (an uncopyrightable element of the computer program). This reasoning illustrates that a court should not classify Lotus 1-2-3's menu command hierarchy as a blanket "method of operation."

This point becomes even more evident when looking at other computer cases. Specifically, in *Apple Computer, Inc. v. Franklin Computer Corp.*,[159] the Third Circuit Court of Appeals analyzed the question of whether a computer operating system is copyrightable.[160] Using the same reasoning as the *Lotus* court,[161] one could only conclude that the operating system performs essentially the same function as the Lotus 1-2-3's menu command hierarchy; it serves as a method of operation for a user to interact with the computer, and thus, is uncopyrightable. In *Apple Computer*, the court held otherwise.[162] After discussing the idea/expression dichotomy, the *Apple Computer* court refused to accept the defense that such operating systems are per se uncopyrightable and recognized that this view was consistent with that of other courts and Congress.[163] Consequently, the *Lotus* decision fails to provide adequate copyright protection for computer companies.

The First Circuit's decision also fails to follow pertinent case law. It determined that the number of different ways there were to operate a program was immaterial.[164] The number of different ways that an author can express an idea directly affects the merger analysis. For example, in *Autoskill*, the Tenth

---

155. *See id.* at 816.
156. *See id.*
157. *See id.* at 817.
158. *See* Petitioner's Brief at 44-45, *Lotus* (No. 94-2003).
159. 714 F.2d 1240 (3d Cir. 1983).
160. *See id.* at 1249-54.
161. *See* discussion *supra* Part V.
162. *See Apple Computer*, 714 F.2d at 1253.
163. *See id.* In *Apple Computer*, the court said:
We . . . focus on whether the idea is capable of various modes of expression. If other programs can be written or created which perform the same function as a Apple's operating system program, then that program is an expression of the idea and hence copyrightable. In essence, this inquiry is no different than that made to determine whether the expression and idea have merged, which has been stated to occur where there are no or few other ways of expressing a particular idea.
*Id.*
164. *See supra* text accompanying note 139.

Circuit Court of Appeals emphasized the fact that the trial court did not find any evidence showing that the idea of the computer program could be expressed in only one way.[165] Consequently, the Tenth Circuit upheld the lower court's finding that the expression did not merge with the underlying idea.[166]

Similar facts exist in the present case. In particular, the *Lotus* trial court concluded that Borland could have constructed "[a] very satisfactory spreadsheet menu tree . . . using different commands and a different command structure from those of Lotus 1-2-3."[167] Therefore, following the reasoning of *Autoskill*, it is clear that the Lotus 1-2-3 menu command hierarchy does not merge with any uncopyrightable element of the computer program.

Once one determines that a court cannot merely classify the menu command hierarchy as a "method of operation," it is evident that the district court's decision provided a superior solution in accordance with precedent. The district court utilized a test to analyze the copyrightability of a computer program's non-literal elements substantially similar to that of the *Altai* test.[168] Although somewhat different, both tests allowed the court to clearly delineate the copyrightable portions of a computer program.[169] Furthermore, the *Altai* court explained:

> [In devising this test] we are cognizant that computer technology is a dynamic field which can quickly outpace judicial decisionmaking. Thus, in cases where the technology in question does not allow for a literal application of the [*Altai* test], our opinion should not be read to foreclose the district courts of our circuit from utilizing a *modified version*.[170]

Also, the First Circuit's policy argument does not justify disregarding Congress' intent for enacting section 102 or relevant case law. Specifically, the First Circuit found the interest of the user important.[171] It determined that a Lotus 1-2-3 user should not have to learn different menu commands for different spreadsheet programs.[172] However, this reasoning focuses only on one interest of the user. In fact, a computer user may have interests other than the ease of transition between different spreadsheet programs. For example, a user may be more interested in the development of better and innovative menu command hierarchy arrangements. The First Circuit's decision, in effect, stifles a computer company's incentive to invest time and research into developing a new, innovative menu command hierarchy. Computer companies will be inclined to wait and improve on another company's ideas. As such, the First

---

165. *See Autoskill*, 994 F.2d at 1494.

166. *See id.*

167. *Borland II*, 799 F. Supp. at 217. Moreover, the trial court held that "[c]hanges in submenus increase the number of possible menu hierarchies *geometrically*. Since there are dozens of independent submenus, the number of possible menu hierarchies is extremely large." *Id.* at 217-18.

168. *See supra* note 115.

169. *See supra* notes 115, 116 and accompanying text.

170. *Altai*, 982 F.2d at 706 (emphasis added).

171. *See supra* notes 137-39 and accompanying text.

172. *See Lotus*, 49 F.3d at 818. Specifically, the court found that forcing the user to learn different commands for different programs "absurd." *Id.*

Circuit created a rule that serves to ease the computer user's transition between similar programs. Unfortunately, the ultimate effect will be to hinder the innovation of new and better menu command hierarchies.

Furthermore, the First Circuit's decision also erroneously relies on *Feist* for support for its policy argument.[173] The court quotes *Feist*:

> The primary objective of copyright is not to reward the labor of authors, but to promote the Progress of Science and useful Arts. To this end, copyright assures authors the right to their original expression, but encourages others to build freely upon the ideas and information conveyed by a work.[174]

The First Circuit decision concludes that its decision does not violate the *Feist* holding.[175] Specifically, it determines that the *Feist* holding does not indicate that all expression is necessarily copyrightable.[176] However, the First Circuit fails to recognize the first portion of the *Feist* holding: that copyright serves to promote the progress of science and useful arts. Consequently, it is clear that the First Circuit's decision fails to accomplish the one thing that would promote the progress of science and the useful arts; in fact, the decision hinders its progress. Without copyright protection, computer companies will continue to take menu command hierarchies from their competitors and not invest the time and money in developing innovative ideas.

## VII. COMPUTER COMPANIES SEEK A SOLUTION

There is a clear split in authority between the different jurisdictions. While the literal computer code still receives federal copyright protection, the menu commands may not. As it stands now, the only indicator of whether menu command hierarchies are protected by copyright is the jurisdiction in which the suit is brought. If a case is brought within the Tenth Circuit, the court will clearly provide copyright protection pursuant to *Autoskill v. National Educational Support Systems*.[177] However, if the suit is brought in a court under the First Circuit, it will not receive similar copyright protection. Rather, it will merely be viewed as a "method of operation" pursuant to *Lotus v. Borland*.

In either case, the Supreme Court decision in *Lotus v. Borland*[178] has caused the computer companies to reevaluate whether to continue to seek copyright protection for its computer programs. Assuming the Supreme Court determines the issue important enough to examine again, one can only speculate on which side of the fence Justice Steven's, or his successor's, vote will land.

---

173. *See supra* note 141 and accompanying text.
174. *Lotus*, 49 F.3d at 818 (quoting *Feist*, 499 U.S. at 349-50).
175. *See id.*
176. *See id.*
177. *See supra* notes 147-49 and accompanying text.
178. 116 S. Ct. 804 (1996).

Amidst all the uncertainty, attorneys have focused on other ways to protect their client's software. Commentators suggest that attorneys are returning to some of the older ways of protecting software by choosing patent and trade secret protection.[179] While these methods of protection have their benefits over copyright, they also have drawbacks. Specifically, patent protection, unlike copyright and trade secret protection, guards against independent creation by other computer programs.[180] Also, patent law protects against reverse engineering, while copyright does not.[181] However, patent protection lasts only seventeen years while copyright protection lasts seventy-five years.[182] Finally, another drawback of patent protection is that it takes two years to obtain such protection and in that time the computer program is often technologically obsolete.[183] Consequently, the uncertainty the Supreme Court left in issuing a split 4-4 vote in *Lotus v. Borland* will likely remain for the time being.

## VIII. CONCLUSION

In conclusion, the First Circuit's decision in *Lotus* classifies Lotus 1-2-3's menu command hierarchy as a "method of operation." In doing so, the court contradicted existing case law. Moreover, the First Circuit relied on federal copyright statutes; however, in doing so it ignored Congress' intent in enacting the statute. That intent was to codify older, court created, copyright protection. The First Circuit's policy argument fails to advance the primary objective of copyright law. Specifically, the First Circuit's policy reason is limited to the concern for the computer user's short term interest of switching between competing computer programs. It does not consider the computer user's long term interest of receiving new and innovative computer programs. It also does not consider the interests of the computer companies who hold a copyright.

The subsequent Supreme Court's split decision effectively precludes computer companies from being able to rely on copyright law to protect the nonliteral, menu command hierarchy portion of their computer programs. By failing to issue an opinion, the lower courts continue to have conflicting law. In the First Circuit, a computer program's menu command hierarchy is considered a "method of operation," and thus, it is not afforded copyright protection. However, in the Tenth Circuit, the same menu command hierarchy must be analyzed using the *Altai* test. Only then can a court in the Tenth Circuit truly know whether the menu command hierarchy can be afforded copyright protection.

Despite the uncertainties with respect to protecting computer programs, attorneys have continued to work to protect their clients. However, they have moved away from copyright law. Instead, they have focused on more traditional

---

179. *See* Dominic Bencivenga, *Beyond Copyright Law: How to Protect Software*, NAT'L L.J., Apr. 22, 1996, at B1.
180. *See id.* at B2.
181. *See id.*
182. *See id.*
183. *See id.*

intellectual property ideas. Specifically, they are attempting to utilize patent and trade secret law. These areas of law offer different protection than that given by copyright. In some instances, they offer more protection; in some instances, they offer less protection. In either case, it is clear that until the Supreme Court revisits the same issue in the future, attorneys will have to work with the confusion among the circuits.

*Christopher Kanagawa*