

BRIEF AMICUS CURIAE

APPLYING FUNDAMENTAL COPYRIGHT PRINCIPLES TO LOTUS DEVELOPMENT CORP. v. BORLAND INTERNATIONAL, INC.*

DENNIS S. KARJALA †

PETER S. MENELL ††

Table of Contents

I.	INTRODUCTION	177
II.	ISSUE IN THE CASE	179
III.	SUMMARY OF ARGUMENT	179
IV.	ARGUMENT	180
	A. Copyright in a Program Does Not Extend to those “Certain Results” Brought about by Program Execution	180
	B. The Critical Relationship between Copyright and Patent Law.....	182
	C. Application of these Principles to the <i>Lotus</i> Case	187

I. INTRODUCTION

Lotus v. Borland raises the fundamental question of whether, and to what extent, the copyright in a computer program protects the user interface generated by that program. Lotus owns the copyright in and markets the very popular spreadsheet program known as Lotus 1-2-3. Borland produced and marketed a competing set of spreadsheet programs known as Quattro. Borland designed and implemented in program code its own user interface, as to which no claim of infringement was made. In addition, however, the Borland programs included a Lotus “emulation mode,” which allowed users familiar with the Lotus program more easily to make the switch to Quattro. That is, when operating

© 1995 Dennis S. Karjala & Peter S. Menell.

* This article is based on a *Brief Amicus Curiae* filed by the authors with the First Circuit Court of Appeals in *Lotus Dev. Corp. v. Borland Int’l, Inc.* The decision is reported at 49 F.3d 807 (1st Cir. 1995). The *High Technology Law Journal* publishes this brief in nearly the same form as submitted to the court—this version contains a short introduction added by the authors as well as minor changes made to conform to law review style.

† Professor of Law, Arizona State University.

†† Professor of Law, University of California, Berkeley.

Quattro in the Lotus emulation mode, the Borland program presented to the user the same hierarchy of menu commands as Lotus 1-2-3 and responded identically to those commands. Lotus brought action for copyright infringement based on the similarity of the Quattro user interface when operated in the emulation mode. No copying of program code was alleged.

The federal district court concluded that the Lotus 1-2-3 user interface as a whole, as well as specific elements of the interface, were copyright protected as nonliteral aspects of the underlying Lotus 1-2-3 computer program.¹ The court based its finding of infringement on the copying of "menu commands, menu command structure, macro language, and keystroke sequences."² The district court's reasoning was based largely on the merger doctrine: Because a wide variety of formats and methods of operation are possible for spreadsheet programs, any given set of choices could be copyright protected without hindering the development of other spreadsheet programs. The court essentially ignored functionality as a limit on copyright protection.

The authors of this brief are law professors at Arizona State University and the University of California (Berkeley), respectively, who teach and write on copyright issues, and specifically on the proper application of copyright law to the protection of computer software.³ We submitted our views on what we perceive to be the crucial issue in this case to the court of appeals because we believe that the district court fundamentally misconstrued both the traditional copyright principles applicable to the facts and the role played by copyright in the overall system of intellectual property protection in the United States. As a result, the district court decision effectively overrules the Supreme Court's decision in *Baker v. Selden*,⁴ a decision that for over 100 years has served to demarcate the boundary between the patent and copyright regimes. The result of the district court's decision is to extend copyright protection far beyond its traditional bounds. We believe that Congress is the only body authorized to make such an expansion of copyright protection, and therefore urged the court of appeals to reverse the district court's decision.

1. *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 799 F. Supp. 203 (D. Mass. 1992). An earlier opinion in the case appears at 788 F. Supp. 78 (D. Mass. 1992).

2. 799 F. Supp. at 222.

3. We represented no party to the litigation nor did we act on behalf of any person other than ourselves. We offered our views to the First Circuit Court of Appeals in the spirit of pro bono publico.

4. 101 U.S. 99 (1879).

II. ISSUE IN THE CASE

Properly viewed, *Lotus v. Borland* does not involve any issue of the copyright protection of computer programs. Rather, it involves the application of traditional copyright law to the *products* of programs—those “certain results” produced by the combinations of statements or instructions that comprise computer programs. The issue in the case is whether functional aspects of these products, traditionally protected solely by patent or trade secret law under the doctrine of *Baker v. Selden* should now be brought under the copyright umbrella. Stated otherwise, the issue is whether the particular spreadsheet format and methods of operation that constitute the user interface of the Lotus 1-2-3 computer program should be protected for the 75-year term of copyright solely on the ground that a variety of other spreadsheet formats and methods of operation are possible, or whether unpatented aspects of this technology should be permitted to progress unencumbered and incrementally through the cumulative efforts of many spreadsheet developers.

III. SUMMARY OF ARGUMENT

The argument that Borland has the right to create an independently written computer program that brings about the same results as the Lotus 1-2-3 computer program is two-fold: First, the statutory definition of a program distinguishes between the program instructions and what those instructions do—the “certain results” they bring about—when they are executed by a computer. The copyright in the program does not extend to the “certain results” brought about by the program. If those “certain results” are to be copyright protected, they must independently qualify as works of authorship.

Second, even if aspects of the Lotus 1-2-3 user interface qualify as works of authorship independently of the underlying computer program—for example, as graphic or audiovisual works—the traditional limits on the scope of any copyright in that interface still apply. These limits include not only the copyright doctrines of merger and *scènes à faire* but also the foundational doctrines of *Baker v. Selden* and section 102(b) of the Copyright Act. These doctrines exclude function from the scope of copyright protection. The policies underlying that tradition are and always have been firmly grounded in the fundamental allocation of protection established by the intellectual property system as a whole. Each of the intellectual property regimes—copyright, patent, trademark, and trade secret—must be understood in the context of this larger system and cannot be interpreted in isolation. Moreover, only Congress, and not the courts, has authority to expand the scope of the intellectual property laws. Because we are addressing an asserted independent copyright in the outputs generated by the Lotus 1-2-3 program and not the program

code itself, protection of functional features of the user interface are not governed by Congress' decision to protect some aspects of program code under copyright. This case centers on the interpretation of *traditional* copyright principles applied to visual images and systems for processing data, and absent instructions from Congress to the contrary, courts must adhere to the doctrines reflected in the Copyright Act and the case law interpreting the intellectual property statutes.

IV. ARGUMENT

A. Copyright in a Program Does Not Extend to those "Certain Results" Brought about by Program Execution⁵

Under the United States Copyright Act, a computer program is "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result."⁶ Unless the "certain result" effected by a program is independently protected under copyright law, anyone has the right to produce independent code bringing about even an identical result.⁷ Even one of the strongest proponents of broad copyright protection for programs agrees that third parties are "free to emulate all external aspects of the program" without infringing the copyright.⁸ It is widely recognized, and not questioned in the *Lotus* case, that different computer programs can generate identical outputs. Therefore, even identity of unprotected output in no way establishes infringement of the underlying program.⁹

5. Our brief discusses an issue not directly addressed in a *Brief Amicus Curiae of Copyright Law Professors* filed in the *Lotus* case (*reprinted* in 16 HASTINGS COMM. & ENT L.J. 657 (1994)), namely, whether a copyright in a program extends to a user interface generated by execution of program instructions. We concur with that brief's conclusions, first, that the district court failed to engage in proper analysis under *Baker v. Selden* and, second, that insofar as the court of appeals chooses to regard user interfaces as nonliteral elements of programs covered by a program copyright, the district court misapplied the test of *Computer Assocs. Int'l, Inc. v. Altai*, 982 F.2d 693 (2d Cir. 1992).

6. 17 U.S.C. § 101 (1988) (definition of "computer program").

7. NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT 21 (1979) [hereinafter CONTU Report] ("One is always free to make the machine do the same thing as it would if it had the copyrighted work placed in it, but only by one's own creative effort rather than by piracy.").

8. Duncan M. Davidson, *Common Law, Uncommon Software*, 47 U. PITT. L. REV. 1037, 1080 (1986).

9. In fact, to the extent that a specific piece of code is the only way of bringing about some aspect of this "certain result", one of only a few practical ways of doing so, or an obvious or standard way of doing so from the point of view of an experienced programmer, even that specific piece of code is not protected, and a finding of infringement cannot be based on similarities in these aspects of the code.

[C]opyright protection for programs does not threaten to block the use of ideas or program language previously developed by others when that use is

This straightforward conclusion—that what a program *does*, and therefore its “appearance” to the outside world when viewed solely through input and output devices, is not protected by the copyright in the program—comports with well over 100 years of copyright tradition and was explicitly recognized by the Second Circuit in *Computer Associates International, Inc. v. Altai*.¹⁰

[W]e note that our decision here does not control infringement actions regarding categorically distinct works, such as certain types of screen displays. These items represent products of computer programs, rather than the programs themselves, and fall under the copyright rubric of audiovisual works.¹¹

Therefore, the analysis of *Computer Associates* and other cases addressing the scope of copyright protection for nonliteral elements of program code simply does not address the central issue of *Lotus v. Borland*: the scope of protection for computer program outputs. The scope of such protection is circumscribed by all of copyright’s traditional limiting doctrines and is not affected by the 1980 amendments to the Copyright Act recognizing the protection of computer programs—sets of statements or instructions—under copyright.¹²

necessary to achieve a certain result. When other language is available, programmers are free to read copyrighted programs and use the ideas embodied in them in preparing their own works

CONTU Report, *supra* note 7, at 20 (first emphasis added). This also follows, of course, from the filtering procedure called for by *Computer Assocs. Int’l, Inc. v. Altai*, 982 F.2d 693 (2nd Cir. 1992).

10. *Computer Assocs.*, 982 F.2d at 703.

11. *Id.* For a more detailed explication of the distinction between a computer program and its user interface, including some of the early judicial and administrative errors in addressing the issue, see Pamela Samuelson, *Computer Programs, User Interfaces, and Section 102(b) of the Copyright Act of 1976: A Critique of Lotus v. Paperback*, 6 HIGH TECH. L.J. 209, 264-69 (Appendix) (1992).

12. The issues raised by this case have confounded other courts as well. In both *Broderbund Software, Inc. v. Unison World, Inc.*, 648 F. Supp. 1127 (N.D. Cal. 1986) and *Digital Comm. Assocs. v. Softklone Distrib. Corp.*, 659 F. Supp. 449 (N.D. Ga. 1987), district courts made a similar analytical mistake as the district court in *Lotus* in applying copyright law to the outputs of computer programs. As with *Lotus Dev. Corp. v. Paperback Software Int’l*, 740 F. Supp. 37 (D. Mass. 1990), these cases were not appealed. Therefore, the First Circuit will be the first appellate court to address the question of copyright protection for interfaces of this type.

Even if a court were to approach the issue of copyright protection for interfaces as a part of the copyright in the underlying program, it is clear that what a computer program does—the screen displays it generates and the methodology it presents to the outside world for using the program via such devices as keyboards and data entry formats—lies at a higher level of abstraction above literal code than even program structure and organization. The recognition of protection for so-called “structure, sequence, and organization” or “SSO” by the Third Circuit in *Whelan, Inc. v. Jaslow Dental Laboratory, Inc.*, 797 F.2d 1222, 1248 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987) caused great controversy, and *Computer Associates* explicitly reduced the breadth of *Whelan*. See, e.g., *Computer Assocs.*, 982 F.2d at 711-12 (noting that *Feist Pubs., Inc. v. Rural Tel. Serv. Co.*, 111 S. Ct. 1282 (1991), implicitly undercuts the *Whelan* rationale and citing with approval

B. The Critical Relationship between Copyright and Patent Law

While the *Lotus* case should be decided under traditional copyright doctrine, an understanding of the critical relationship between copyright and patent protection is essential to understanding why a decision significantly limiting copyright protection for some user interfaces, such as that produced by the Lotus 1-2-3 program, best comports with the overall system of intellectual property protection. In fact, it is the broad copyright protection for such interfaces conferred by the district court in *Lotus* and its earlier *Paperback* decision¹³ that constitutes a radical departure from fundamental intellectual property law principles.

Despite the similar basic approach of patent and copyright law—both draw a balance between providing an incentive for the creation of works and ensuring that the public and later creators can enjoy and build upon an expanding “public domain”—they are distinct components of a coherent scheme of intellectual property protection. Patent law seeks to promote the advancement of technology while copyright law seeks to encourage culture and the arts. Given their different foci, it is not surprising that Congress has crafted very different balances in these two statutes. In exchange for protection of claimed functional attributes of an invention, the patent law requires an inventor to demonstrate to a technically trained examiner that the asserted invention is novel, nonobvious, and useful. After meeting these exacting requirements, the holder of a United States patent is entitled to 17 years of exclusive rights to practice the invention. By contrast, the copyright law presents a much lower threshold for receiving protection—a minimal level of “creativity”—and forgoes any requirement of examination. In return, the author is entitled to protection limited to the expressive content of the work. The term of this protection, however, extends for the life of the author plus 50 years (or 75 years for entity authors).

critics of *Whelan's* sweeping scope of protection). However the debate is ultimately resolved concerning the level of detailed program structure that might be covered by a program copyright as a nonliteral element of the program, user interfaces are not nonliteral elements of programs but rather the results of programs. To the extent that *Lotus* relies on the copyright in its 1-2-3 program for protection of that program's user interface, therefore, its claim should fail. More fundamentally, basic copyright law requires that outputs of computer programs be evaluated solely on the basis of their own merits as expressive works of authorship and not any copyright in the program that produces them. For this reason, too, the district court's attempt to distinguish *Baker v. Selden* as a precedent predating the congressional mandate to protect computer programs, 799 F. Supp. at 215, is without logical support.

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

In view of the very different thresholds for as well as scope and terms of protection, to extend copyright by protecting functional works¹⁴ without explicit congressional authorization presents grave risks to the integrity of the entire intellectual property law system. Thus, in the seminal case of *Baker v. Selden*,¹⁵ decided more than a century ago, the United States Supreme Court explicitly recognized the fundamentally different roles played by copyright and patent law within that system:

The copyright of the book, if not pirated from other works, would be valid without regard to the novelty, or want of novelty, of its subject-matter. The novelty of the art or thing described or explained has nothing to do with the validity of the copyright. To give to the author of the book an exclusive property in the art described therein, when no examination of its novelty has ever been officially made, would be a surprise and a fraud upon the public. That is the province of letters-patent, not copyright. The claim to an invention or discovery of an art or manufacture must be subjected to the examination of the Patent Office before an exclusive right therein can

14. In general, we may say that traditional copyright law, following the dictates of *Baker v. Selden*, has eschewed the protection of function, provided the meaning of the term "function" is understood in a precise way. Copyright does, of course, protect many works that are "useful" to human beings. Maps enable us to go from one place to another; recipes tell us how to bake cakes; accounting books explain how to implement a particular system of accounting. All of these works are copyright protected. They are not, however, "useful articles" within the definition in the Copyright Act, since "[a] 'useful article' is an article having an intrinsic utilitarian function that is not merely to portray the appearance of the article or to convey information. . . ." 17 U.S.C. § 101 (1988). Although this definition was adopted in an effort to deal with the problem of industrial design—by excluding utilitarian functions other than to inform or portray an appearance from copyright protection and excluding even expressive aspects of useful articles to the extent they are not separable from the utilitarian functions, 17 U.S.C. § 101 (1988) (definition of "pictorial, graphic, and sculptural works")—it represents a statutory description of the kind of "usefulness" or "functionality" that has always been excluded from copyright protection, as opposed to the "usefulness" inherent in maps and recipe books that have long been a part of copyright.

The Lotus 1-2-3 user interface is functional in the sense used here, because it has the intrinsic functional goal of permitting users of the program to input and manipulate data in a fast, efficient, and easy-to-master manner. If, by some chance, Lotus has managed to discover the fastest, most efficient, and easiest-to-master means of using spreadsheet programs, recognition of a copyright in its user interface is equivalent to giving Lotus a 75-year patent in an optimal functional tool. If, as is more likely, Lotus has made an important advance toward the goal of an optimal spreadsheet interface, recognition of a copyright in its functional features inhibits, for the same period, further incremental development in these tools, without any showing that the advance would have been sufficient for even a 17-year patent.

15. 101 U.S. 99, 102 (1879). In this case Selden was the author of a book describing a system of accounting that Selden claimed to have invented. The book included sample forms for implementing the system by a user. Baker's books taught how to implement a similar bookkeeping system and included sample forms that were similar to the forms in Selden's book. There was no allegation that any textual explanation from Selden's book was pirated. The claim of infringement was predicated on similarity of Baker's forms to those of Selden.

be obtained; and it can only be secured by a patent from the government.¹⁶

The Supreme Court in *Baker* thus adopts an important channeling principle that it then applies to the facts of the case:

Charles Selden, by his books, explained and described a peculiar system of book-keeping, and illustrated his method by means of ruled lines and blank columns, with proper headings on a page, or on successive pages. Now, whilst no one has a right to print or publish his book, or any material part thereof, as a book intended to convey instruction in the art, any person may practise and use the art itself which he has described and illustrated therein. The use of the art is a totally different thing from a publication of the book explaining it. The copyright of a book on book-keeping cannot secure the exclusive right to make, sell, and use account-books prepared upon the plan set forth in such book. Whether the art might or might not have been patented, is a question which is not before us. It was not patented, and is open and free to the use of the public. And, of course, in using the art, the ruled lines and headings of accounts must necessarily be used as incident to it.¹⁷

The fact that Selden's accounting forms may have satisfied the minimal level of "creativity" required by the copyright law and could have been expressed in a variety of ways was irrelevant to the *Baker* decision. A copyright in the book in no way prevents others from using the methods described, or the forms needed to execute the system.¹⁸ In this way, the Supreme Court established a critical limitation on copyright protection to ensure that technological advances not satisfying the exacting requirements of patent law do not indirectly receive protection through copyright and to preserve the integrity of the entire intellectual property system.

The policy basis for the distinct approaches of patent and copyright law is the social desirability of allowing later technological creators—creators of functional works—to build on and improve, often in small ways, the earlier works of others.¹⁹ Technological improvements are often substantially similar to the products they improve and would infringe if the copyright standard were applied. Such improvements do not infringe a patent in the technology unless they adopt all of the elements, or their substantial equivalents, of a patent claim. Authors of copyright-protected works, as well, build upon public domain works and unprotected elements of protected works, but in taking from the latter, copyright law limits them to the more general or abstract features,

16. *Id.* at 102.

17. *Id.* at 104.

18. Many cases have implemented this aspect of *Baker v. Selden*. For a more extended discussion and citations, see Samuelson, *supra* note 11, at 226-27 & n.73.

19. See Dennis S. Karjala, *Copyright, Computer Software, and the New Protectionism*, 28 *JURIMETRICS* 33, 39 (1987).

together with those aspects denied copyright protection under *Baker v. Selden*, section 102(b) of the United States Copyright Act, and other limiting doctrines.

Later inventors thus can apply a "ground up" approach to reliance on protected works that precede. Once they are outside all patent claims, they are safe. Later authors, however, must adopt a "top down" approach and take a serious risk of being held liable for infringement as their reliance becomes more detailed. The different foci of patent and copyright law explain their different approaches to infringement. The social utility of allowing subsequent authors to make minor variations on a copyright-protected novel is minimal. For works of fiction, art, and music, variety is the spice of both legal and real life. We prefer to have one hundred different war novels than one hundred versions of "War and Peace" that differ only in their final chapter. Consequently, the broad scope of copyright protection for novels and the long period of its duration fulfills the goal of recognizing the author's creativity without unduly hindering later authors or depriving society of desirable works.²⁰

Technology, however, improves incrementally, as later inventors add a bell or a whistle to an earlier invention to make it more desirable or useful to consumers. Many improvements on existing products, being rather straightforward or "obvious" in the sense of patent law, are given no intellectual property protection once they are released to the public. Such products often show at least as much intellectual creativity as many copyright-protected works, but their creators have a monopoly only for the period that is required for competitors to recognize the value or popularity of the improved product, figure out its "secret," if any, and gear up for production and marketing. In the case of technological products we have drawn the social policy balance at a different point than for traditional works of authorship because we believe that to grant intellectual property rights in ordinary engineering advances would hinder the development of more and better products than it would encourage.²¹ Hence the "nonobviousness" requirement of patent law, as

20. *Kepner-Tregoe, Inc. v. Carabio*, 203 U.S.P.Q. 124, 131 (BNA) (E.D. Mich. 1979) ("[T]here is no societal interest in many variants on a single theme or plot, nor is there the likelihood that by extending broad protection, entry to the market for literary works will be foreclosed.").

21. The effort and creativity that go into unpatentable technological advances can be compared to similar unprotected efforts of historians and other authors:

The urge to compensate for subsequent use of information and ideas is perhaps understandable. An inequity seems to lurk in the idea that much of the fruit of the historian's labor may be used without compensation. This, however, is not some unforeseen byproduct of a statutory scheme intended primarily to ensure a return for works of the imagination. Congress made the affirmative choice that the copyright laws should apply in this way. . . . To ensure the progress of arts and sciences and the integrity of First

well as its shorter term and disclosure requirements, recognize and promote the incremental nature of technological innovation.

Computer programs, of course, at least in object-code form, are technological in nature: They sequentially set the switches inside a computing machine in such a way that the results can be interpreted as "information processing" by human beings. To the extent that code is now copyright protected, Congress has indicated that a functional work should come under the copyright umbrella. However, the National Commission on New Technological Uses of Copyright Works (CONTU)—established by Congress "to assist the President and Congress in developing a national policy for protecting both the rights of copyright owners and insuring public access to copyrighted works when they are used in computer . . . systems, bearing in mind the public and consumer interest,"²²—concluded that literal copying of computer programs would dramatically reduce the development costs of second comers and thereby create a disincentive to invest in computer program development.²³ Even if we assume that the application of *Baker v. Selden* and section 102(b) to program code is now more limited than to other technological works—in that the literal code of every program can be said to implement a process or method of operation and yet remains copyright protected—there is no basis for concluding that Congress intended to remove any barriers to the copyright protection of function in other classes of works. The Supreme Court in *Feist*,²⁴ the Second and Ninth Circuits in *Computer Associates*²⁵ and *Sega*,²⁶ and the Northern District of California in *Apple Computer, Inc. v. Microsoft Corp.*²⁷ have recently reaffirmed the fundamental role of *Baker v. Selden* in copyright analysis.

Amendment values, ideas and information must not be freighted with claims of proprietary right.

Harper & Row Pubs., Inc. v. Nation Enters., 471 U.S. 539, 589 (1985) (Brennan, J., dissenting), *quoted with approval in* Feist Pubs., Inc. v. Rural Tel. Serv. Co., 111 S. Ct. 1282, 1289-90 (1991).

22. CONTU Report, *supra* note 7, at 3.

23. Every concrete example of infringement that the CONTU report offers involves direct and literal copying, such as photocopying of printed source code and one-to-one transcription from magnetic tape or disk to paper. CONTU Report, *supra* note 7, at 22-23. The Report goes on to say that "[m]ost infringements, at least in the immediate future, are likely to involve simply copying," but the problem it predicts for the then-future was a technology that permits programs to be stated orally or permits use of a program without copying. *Id.* The CONTU Report nowhere refers to copying of "nonliteral elements," much less copying of user interfaces or other results of programs, as potential infringement.

24. Feist Pubs., Inc. v. Rural Tel. Serv. Co., 111 S. Ct. 1282, 1290, 1297 (1991).

25. Computer Assocs. Int'l, Inc. v. Altai, 982 F.2d 693, 703-05 (2nd Cir. 1992).

26. Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1524 (9th Cir. 1992), *amended*, 1993 U.S. App. LEXIS 78 (9th Cir.).

27. 799 F. Supp. 1006, 1021 (N.D. Cal. 1992).

C Application of these Principles to the *Lotus* Case

On a technical analytical level, a court could hardly face a situation in the modern age more squarely on point with *Baker v. Selden* than *Lotus v. Borland*. Selden had created an accounting system—building on ideas of others dating back at least to the innovation of double-entry bookkeeping in 1494 by Luca Paciola²⁸—that employed “ruled lines and blank columns, with proper headings on a page, or on successive pages.”²⁹ The organization and content of the blank forms were necessary for anyone to use Selden’s system. The forms were, in other words, the predigital interface between Selden’s system and the user. Just as Selden’s accounting system was an incremental improvement upon prior systems of accounting, Lotus freely built upon prior electronic spreadsheet systems dating back at least to Visicalc, introduced in 1979.³⁰ Like Selden, Lotus started with existing technology and created a system for entering information into a computer, processing the information, displaying the information on the screen, and putting results obtained into hard copy.³¹ That system is implemented by a computer program, and because of the speed of digital processing it is more sophisticated than Selden’s accounting system, but there is no essential distinction between them as far as traditional copyright law is concerned. In fact, the presentation of a spreadsheet is often remarkably similar to and no more complex than Selden’s “ruled lines and blank columns, with proper headings on a page, or on successive pages.”³² Moreover, the menu

28. See LUCA PACIOLA, *SUMMA DE ARITHMETICA, GEOMETRIA, PROPORZIONI ET PROPORZIONALITA* (1494), discussed in T. BUDD & E. WRIGHT, *THE INTERPRETATION OF ACCOUNTS* 7 (1930).

29. *Baker v. Selden*, 101 U.S. 99, 104 (1879).

30. See Peter S. Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 STAN. L. REV. 1045, 1057 (1989) [hereinafter Menell, *Scope of Copyright Protection*] (discussing the process of technological advancement in application programming). The basic methods underlying Visicalc and Lotus 1-2-3 extend back at least to the time of Luca Paciola, as a principal application of spreadsheet analysis is bookkeeping. The key distinctive feature of the Visicalc and Lotus products is the ability simultaneously to calculate and alter tables of information. The basic means by which this ability was brought into accounting—digital processing—was of course the product of neither Visicalc nor Lotus but rather hundreds of predecessors, human and corporate, active in the development of computer science.

31. What the user creates using the Lotus system may, of course, be copyright protected. The user-created products of most spreadsheet and word processing software almost always have no function other than to inform or entertain. They are therefore not “functional” in the sense used to distinguish patent and copyright subject matter. See *supra* note 14. This product of the end user must be distinguished from the functional interface developed by Lotus to assist the user’s creation of such works. The typewriter is a system that also assists in the creation of copyright-protected works, but the machine itself is not protected by copyright, nor is its user interface—the arrangement of the keys.

32. *Id.*

structure of the Lotus system is a sequence of operations analogous to the ordered process of bookkeeping described in Selden's book.

In stark contrast to the Supreme Court's analysis in *Baker v. Selden*, the district court in *Lotus v. Borland* begins its infringement analysis with an explicit refusal to treat as unprotected idea the very systemic aspects of the Lotus 1-2-3 interface that are denied protection under *Baker*. The court sets up an abstractions scale going from an electronic spreadsheet at the high end down to the precise Lotus 1-2-3 command set as hierarchically arranged by Lotus at the low end.³³ It rejects Borland's argument that the latter is also unprotected for the following reasons:

The premise of Borland's argument is that an "idea" of Lotus 1-2-3 version 2.01 is complete compatibility with earlier versions of 1-2-3, and more precisely with macros generated for use with earlier versions. Borland argues that the precise menu commands and menu structure are necessary to such functional compatibility. . . . This argument is essentially tautological. As applied to any case involving a useful article, an argument of this kind would always define the idea to incorporate all the specifics of the particular expression of that idea in the allegedly copyrightable work. Nothing would be copyrightable under this methodology of analysis.³⁴

The analysis here is perfectly correct, as far as it goes. Nothing in the user interface of Selden's accounting system *was* copyright protected, precisely because of the necessity of using that specific interface in practicing Selden's system. Unfortunately, the district court in *Lotus* draws exactly the wrong conclusion from this analysis. Its insistence that *something* in the Lotus 1-2-3 system must be copyright protected leads to a result in direct conflict with the holding of *Baker v. Selden*.

The second step of the district court's infringement standard compounds this error by simply assessing whether there were many ways of representing spreadsheet interfaces and designing menu trees for inputting and processing information:

SECOND, the decisionmaker must focus upon whether an alleged expression of the idea, system, process, procedure, or method is limited to elements essential to expression of that idea, system, process, procedure, or method (or is one of only a few ways of expressing the idea, system, process, procedure, or method) or instead includes identifiable elements of expression not essential to every expression of that idea, system, process, procedure, or method.³⁵

The court finds that by changing the names of the commands (and therefore the "natural" keystrokes used to invoke them, such as their initial letters) in the main menu and submenus, an "extremely large"

33. *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 799 F. Supp. 203, 216 (D. Mass. 1992).

34. *Id.* at 217.

35. *Id.*

number of possibilities different from the specific choices made by Lotus were possible.³⁶ The court adopts a similar approach with respect to the arrangement or hierarchy of the commands.³⁷

Had it been dealing with a fanciful work, the district court's analysis would have been appropriate, but *Baker v. Selden* and section 102(b) of the Copyright Act mean more than the merger doctrine for functional features. The whole point of *Baker's* channeling principle between patent and copyright law is that systems and methodologies are not copyright protected. In this case, users of the Lotus 1-2-3 have learned the Lotus system and many want to use *it*, not some other system. Other users have developed "macros" that must combine Lotus commands precisely as Lotus defines them in order to run on the Lotus system. In order to use the Lotus system, there *must* be identity of commands and command structures. Of course, it is possible for Borland and other spreadsheet developers to create other systems. Borland, in fact, did that with its native mode interface. But this possibility is irrelevant to the application of *Baker v. Selden* to the facts of *Lotus*: Under *Baker*, the Lotus system is simply unprotected by copyright.³⁸

Applying the district court's analysis to the facts of *Baker v. Selden* would therefore require a different outcome in that case. There are countless ways of bookkeeping and methods for inputting and processing accounting information. Hence, *Selden* would be entitled to protection for his system as well as his forms for recording the information in the manner demanded by his system. Had *Selden* implemented his accounting system by way of a computer program, as Lotus did, rather than simply describe it in a book, he would under this analysis be entitled to protect the "ruled lines and headings of accounts" and structured methodology of his system. Thus, the district court in this case has effectively overruled *Baker v. Selden*, a central pillar in the United States intellectual property protection system.

This is not to say that there can never be copyright-protected "expression" in a computer program user interface. A straightforward example would be the fanciful characters and the environments in which they chase or avoid enemies in video games.³⁹ It is even possible that

36. *Id.* at 217-18.

37. *Id.* at 218-19.

38. That the district court did not apply the proper test in this case is clear. The court repeatedly refers to functionality in elements it found to be protected: "[T]he menu commands are an important part of the functionality of the macros." *Id.* at 208. "Borland admits to copying the functionality of the keystroke sequences and macro language." *Id.* at 209. "The menu command hierarchy is an integral part of the functionality of the macros and of the keystroke sequences." *Id.* at 219.

39. *E.g.*, *Stern Elecs., Inc. v. Kaufman*, 669 F.2d 852, 854 (2nd Cir. 1982) (spaceships, ground missile bases and fuel depots, and background terrain); *Atari, Inc. v. North*

some aspects of the Lotus 1-2-3 user interface, at least its screen displays, are included solely for aesthetic purposes and are not part of any of the processes, operations, or methods by which the user actually employs the program. Textual information under the "Help" operation will in many cases be copyright protected like any other literary work. But that should be the test: Is an aspect of the interface chosen solely to portray an appearance or to convey information or does it have another utilitarian function—is it something that the user must do in order to make *this* program operate in the intended manner?⁴⁰ If it is, section 102(b) and *Baker v. Selden* require that it be excluded from copyright protection.

Lotus may feel that it has made a fundamental advance in making personal computers useful to nonspecialists. If, in fact, Lotus has made a nonobvious advance in the art over the products of predecessors on whom Lotus relied, it should have sought patent protection for those advances.⁴¹ Lotus has a copyright in its program, which is undoubtedly a complex piece of work, that affords Lotus important protection against piracy of program code. Lotus does not allege, however, that Borland copied its program code or protectable nonliteral elements (if any) of that program code. Although it is not strictly relevant to the copyright analysis, Borland did not even gain the advantage over Lotus of avoiding the design costs of a graphical user interface for a spreadsheet program because Borland designed *both* what it felt would be a better interface *and* wrote code that would implement the Lotus interface to accommodate users who, like the beleaguered users of the QWERTY typewriter keyboard, had learned the Lotus system and were reluctant to change. It is precisely that kind of productive use of unpatented functional works that the exclusion from copyright protection of *Baker v. Selden* is designed to promote.

We also do not mean to suggest that aspects of user interfaces or other outputs of computer programs might not merit greater protection than currently exists under intellectual property law. Rather, we believe, as did CONTU,⁴² that the legislature is the appropriate body to expand the scope of intellectual property protection. In the 1980 amendments to the Copyright Act and the Semiconductor Chip Protection Act of 1984, Congress has, after careful study and broad involvement of affected interests and the public at large, extended intellectual property protection

American Philips Consumer Elecs. Corp., 672 F.2d 607, 617 (7th Cir.), *cert. denied*, 459 U.S. 880 (1982) (Pac Man characters).

40. See *supra* note 14.

41. It is worth noting that the Lotus 1-2-3 interface was itself developed at a time when the only judicial decision on the question pointed in the direction of nonprotection under copyright. Synercom Technology, Inc. v. Univ. Computing Co., 462 F. Supp. 1003 (N.D. Tex. 1978).

42. See CONTU Report, *supra* note 7, at 46.

to new classes of computer-related works. Over the past four years, the Office of Technology Assessment has undertaken, at the behest of Congress, studies of the efficacy of existing legal protection for computer software.⁴³ We fully expect that Congress will seek to address perceived gaps in this system should existing protection, properly applied by the courts, prove inadequate. It is not, however, the province of courts to take over this lawmaking function by extending protection beyond the limits established by traditional doctrines undergirding the intellectual property system. The district court in *Lotus v. Borland* clearly exceeded these bounds, expanding protection afforded *outputs* of computer programs—user interfaces—and effectively overruling *Baker v. Selden* without the authorization of Congress. Even if additional protection were warranted for computer-user interfaces, it is not at all clear that traditional copyright protection—with its long term of protection and exclusivity of rights—would be appropriate.⁴⁴ As other courts grappling with software cases have recognized, the judiciary is ill-suited to weighing the complex issues involved in expanding or reforming intellectual property law.⁴⁵

Computer software, by its very nature as written work intended to serve utilitarian purposes, defies easy categorization within our intellectual property system. Cases involving computer software have therefore presented difficult challenges for courts, which have not always correctly applied traditional copyright doctrines in conjunction with Congress's adoption of CONTU's recommendations to a complex and evolving technology. As a prime example, *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc.*,⁴⁶ stood for six years as an influential decision on the scope of protection for nonliteral aspects of program code despite cogent academic criticism. The Second Circuit's sweeping repudiation of the *Whelan* approach in the *Computer Associates* case in 1992, building upon this academic literature and reasserting essential limiting doctrines of copyright law, has supplanted the *Whelan* approach in substantially all subsequent decisions.⁴⁷ Similarly with regard to copyright protection for

43. U.S. CONGRESS, OFFICE OF TECHNOLOGY ASSESSMENT, FINDING A BALANCE: COMPUTER SOFTWARE, INTELLECTUAL PROPERTY AND THE CHALLENGE OF TECHNOLOGICAL CHANGE (May 1992); U.S. CONGRESS, OFFICE OF TECHNOLOGY ASSESSMENT, COMPUTER SOFTWARE AND INTELLECTUAL PROPERTY: BACKGROUND PAPER (March 1990).

44. See Peter S. Menell, *Tailoring Legal Protection for Computer Software*, 39 STAN. L. REV. 1329, 1370-71 (1987); Menell, *Scope of Copyright Protection*, *supra* note 30 at 1050 n.20 (1989).

45. See, e.g., *Computer Assocs. Int'l Inc. v. Altai*, 982 F.2d 693, 712 (2nd Cir. 1992).

46. 797 F.2d 1222 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987).

47. *Computer Assocs. Int'l Inc. v. Altai*, 982 F.2d 693 (2nd Cir. 1992); see *Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, 9 F.3d 823, 840 n.17 (10th Cir. 1993); *Sega Enters. Ltd. v. Accolade, Inc.*, 977 F.2d 1510, 1524-25 (9th Cir. 1992), *amended*, 1993 U.S. App. LEXIS 78 (9th Cir.); *Atari Games Corp. v. Nintendo of Am., Inc.*, 975 F.2d 832, 839 (Fed. Cir. 1992);

interfaces, most early district courts addressing the issue—resting upon an erroneous assumption that the copyright in the computer program extends to the outputs of the program and/or building upon the faulty foundation of *Whelan*—have gone astray of fundamental copyright doctrines.⁴⁸ As the first appellate court to review this issue, the First Circuit in this case should seize the opportunity to bring copyright protection for outputs of computer programs back into line with the fundamental principles of copyright law.

Apple Computer, Inc. v. Microsoft Corp., 799 F. Supp. 1006, 1024-25 (N.D. Cal. 1992); CMAX/Cleveland, Inc. v. UCR, Inc., 804 F. Supp. 337, 352 (N.D. Ga. 1992).

48. Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37 (D. Mass. 1990); Digital Commun. Assocs. v. Softklone Distrib. Corp., 659 F. Supp. 449 (N.D. Ga. 1987); Broderbund Software, Inc. v. Unison World, Inc., 648 F. Supp. 1127 (N.D. Cal. 1986).