

ARTICLE

COMPUTER PROGRAMS, USER INTERFACES, AND SECTION 102(b) OF THE COPYRIGHT ACT OF 1976: A CRITIQUE OF LOTUS V. PAPERBACK

PAMELA SAMUELSON[†]

Table of Contents

I.	INTRODUCTION	210
II.	THE <i>LOTUS V. PAPERBACK</i> DISPUTE	214
III.	THE "UNCOPYRIGHTABILITY" OF "NONLITERAL ELEMENTS" DEFENSE	217
	A. The Nonliteral Elements Defense	218
	B. <i>Whelan's</i> Rejection of One "Bright Line" Test and Adoption of Another	219
	C. <i>Paperback's</i> Use of the <i>Whelan</i> Test	221
	D. The Errors of the <i>Whelan</i> Test	223
IV.	WAS THE LOTUS COMMAND STRUCTURE PART OF THE LOTUS SPREADSHEET OR MACRO SYSTEM?	226
	A. <i>Baker v. Selden</i> : The Arrangement of Words As Part of a System	227
	B. The Macro Facility of Lotus 1-2-3 As a System	235
	C. The "Compatibility" Defense	239
V.	WHAT WAS "EXPRESSIVE" ABOUT THE LOTUS INTERFACE?	243
	A. Copyright Cases and Principles That <i>Paperback</i> Should Have Discussed	245

© 1992 Pamela Samuelson

† Professor of Law, University of Pittsburgh School of Law. An earlier version of this article was published in the Spring 1992 issue of *Law & Contemporary Problems*. This article, like the symposium published in that journal, is dedicated to Robert Kastenmeier in appreciation for his many contributions to U.S. copyright law and policy, especially his foresight in recognizing the need to codify some of the limiting principles of copyright law, such as those that appear in 17 U.S.C. § 102(b) (1988). The author wishes to thank Robert J. Glushko, Mitchell D. Kapor, and Professors Ralph Brown, Randall Davis, Dennis Karjala, and Jerome Reichman for their insightful comments on an earlier draft of the article and her research assistant Linda Sharif for her thoughtful, thorough, and cheerful help in the preparation of this article.

B. <i>Paperback's</i> Unwillingness To Accept the Proper Copyright Consequences of The Functionality of Computer Programs And User Interfaces.....	251
C. Two Kinds of Expressiveness Inquiries That <i>Paperback</i> Might Have Made	256
VI. CONCLUSION	262
APPENDIX	264

I. INTRODUCTION

Judges in computer software copyright cases have paid scant attention to a provision of the copyright statute¹ intended to limit the scope of copyright protection in accordance with the principles of the most venerated of American copyright cases, *Baker v. Selden*.² That provision is § 102(b) of the Copyright Act of 1976. It states: "In no event does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle or discovery."³ That § 102(b) has been so little analyzed in software copyright cases is especially surprising because it was added to the copyright statute in part to ensure that copyright protection for computer programs would not be construed too broadly.⁴

1. See, e.g., *Whelan Assoc. v. Jaslow Dental Lab., Inc.*, 609 F. Supp. 1307 (E.D. Pa. 1985) (no mention of § 102(b) in trial court ruling; section mentioned, but not incorporated into the appellate court's infringement test or infringement analysis), *aff'd* 797 F.2d 1222, 1234-42 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987); *Manufacturers Technologies, Inc. v. CAMS, Inc.*, 706 F. Supp. 984, 995-96 (D. Conn. 1989) (§ 102(b) only invoked to render certain user interface navigational conventions unprotectable by copyright law); *Broderbund Software, Inc. v. Unison World, Inc.*, 648 F. Supp. 1127, 1131 (N.D. Cal. 1986) (one "see also" reference to § 102(b), but no discussion of the provision in the court's analysis of the copyright issues).

2. 101 U.S. 99 (1879). The extent to which Congress intended to codify *Baker v. Selden* by enacting 17 U.S.C. § 102(b) (1988) is the subject of some academic debate. See J.H. Reichman, *Computer Programs As Applied Scientific Know-How: Implications of Copyright Protection for Commercialized University Research*, 42 VAND. L. REV. 639, 693, n.288 (1989). Courts interpreting § 102(b) have construed it as codifying substantial portions of *Baker v. Selden*. See, e.g., *NEC v. Intel Corp.*, 10 U.S.P.Q.2d (BNA) 1177, 1179 (N.D. Cal. 1989); *Signo Trading Int'l v. Gordon*, 535 F. Supp. 362, 365 (N.D. Cal. 1981). The words of the provision, particularly those that refer to the unprotectability of processes, procedures, systems, and methods of operation are derived from *Baker v. Selden* and the many cases following it, as Section IV will show.

3. 17 U.S.C. § 102(b).

4. The House and Senate Reports state quite clearly:

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 copyright law.

In the twelve years since the copyright statute was amended to state explicitly that computer programs could be protected by copyright law,⁵ there have been numerous controversies about the proper application of copyright law to computer programs.⁶ Especially controversial have been several cases involving user interfaces of computer programs.⁷ A case

H.R. REP. NO. 1476, 94th Cong., 2d Sess. 57 (1976), reprinted in 1976 U.S.C.C.A.N. 5659, 5670 and in S. REP. NO. 473, 94th Cong., 2d Sess. 54 (1976). Professor Arthur Miller testified at hearings leading up to passage of the 1976 Act that without a provision of this sort in the statute, copyrights in computer programs might become the equivalent of patents for important elements of programs. This, he asserted, would stultify development of software. See *Hearings on S. 597 Before the Senate Subcomm. on Patents, Trademarks, and Copyrights of the Comm. on the Judiciary*, 90th Cong., 1st Sess. 197 (1967). See also Letter from Professor Arthur R. Miller, Professor, Harvard Law School, to Salem M. Katsh, Esq., Weil, Gotshal & Manges 6-7 (Oct. 29, 1985) (on file with author) (expressing the view that § 102(b) codified *Baker v. Selden*; also indicating that it was for patent law, not copyright, to protect the processes, systems, and methods of operations embodied in programs).

5. In 1980 Congress amended the copyright statute to make clear that copyright protection was available for computer programs. See An Act to Amend the Patent and Trademark Laws, H.R. 6943, 96th Cong., 2d Sess. (1980), which became Pub. L. No. 96-517, 94 Stat. 3007 (1980) (codified at 17 U.S.C. §§ 101, 117). This legislation implemented the recommendations of the National Commission on New Technological Uses of Copyrighted Works ("CONTU"). See FINAL REPORT OF THE NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, ch. 3 (1979) [hereinafter CONTU REPORT].

Notwithstanding the 1980 amendments, there were a number of challenges to the "copyrightability" of operating system programs and microcode because of their highly functional nature. See, e.g., Peter S. Menell, *Tailoring Legal Protection for Computer Software*, 39 STAN. L. REV. 1329 (1987); Pamela Samuelson, *CONTU Revisited: The Case Against Copyright Protection for Computer Programs in Machine-Readable Form*, 1984 DUKE L.J. 663 (1984) [hereinafter *CONTU Revisited*]. Although these challenges were unsuccessful, the functionality of computer programs or particular aspects of them has continued to be problematical in the copyright caselaw, as *Lotus v. Paperback* illustrates.

6. An overview of the many controversies about the application of copyright to computer programs can be found in Pamela Samuelson, *Reflections on The State of American Software Copyright Law and the Perils of Teaching It*, 13 COLUM.-VLA J.L. & ARTS 61 (1988) [hereinafter *Reflections on American Copyright Law*]. For a more in-depth treatment of copyright issues raised by the protection of structural abstractions of computer programs and user interfaces, see Peter S. Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 STAN. L. REV. 1045 (1989) [hereinafter *Copyright Protection for Application Programs*]; *LaST Frontier Conference Report on Copyright Protection of Computer Software*, 30 JURIMETRICS J. 15 (1989) [hereinafter *LaST Frontier Report*].

7. Because the first several software user interface copyright cases did not involve major software firms (see, e.g., cases cited *supra* note 1), the cases were not much noticed in the software industry. Some of them, however, laid the groundwork for the later more publicized user interface lawsuits brought by Lotus, Ashton-Tate, Apple, and Xerox. Until these lawsuits, the general view in the technical community seemed to be that as long as someone wrote his or her own code, it was legal to develop a competing application program with the same or very similar user interface. The apparent legality of computer hardware clones may have caused people in the industry to think, particularly in view of the interchangeability of hardware and software, that software clones would be legal as well. Software clones are thought by some in the technical community to offer many of the same benefits to consumers that hardware clones had offered: more product choices, price competition, and improved or additional features. See generally Pamela Samuelson, *Why The Look and Feel of Software User Interfaces Should Not Be Protected by Copyright Law*, 32 COMM. ACM 563 (1989) [hereinafter *Look and Feel of Software User Interfaces*]. See also

that was widely expected to yield a landmark ruling on one set of controversial user interface issues was *Lotus Dev. Corp. v. Paperback Software Int'l*.⁸ In June of 1990, Judge Keeton issued a lengthy opinion in which he ruled that Lotus' copyright in its popular 1-2-3 spreadsheet program protected the user interface of that program, and that Paperback had infringed the Lotus copyright by copying a number of aspects of that interface.⁹

This article will argue that some important aspects of the Lotus interface copied by Paperback may have been constituent elements of a "system" which under *Baker v. Selden*, its progeny, and § 102(b) are unprotectable by copyright law.¹⁰ The court in *Paperback* seems to have been so distracted by defense arguments about the "uncopyrightability" of "nonliteral elements" of computer programs that it failed to make an appropriate inquiry about this matter. A number of statements in *Paperback* indicate that some important aspects of the Lotus interface were constituent elements of a spreadsheet or macro "system." Yet the court failed to explain why these elements, if constituent parts of a "system," were not as unprotectable under § 102(b) as the ledger sheets that the Supreme Court ruled were unprotectable components of Selden's bookkeeping system more than a century ago.¹¹ By failing to make an appropriate § 102(b) inquiry, the court in *Paperback* missed an important opportunity to provide an updated interpretation of *Baker v. Selden*, the copyright principles it represents, the embodiment of these principles in § 102(b), and how these principles should be applied in copyright disputes involving user interfaces of computer programs.

In addition, this article will argue that the *Paperback* opinion fails to offer persuasive reasons for its ruling that the aspects of the Lotus

Pamela Samuelson & Robert J. Glushko, *Comparing the Views of Lawyers and User Interface Designers on the Software Copyright "Look and Feel" Lawsuits*, 30 JURIMETRICS J. 121 (1989) (reporting on a survey of user interface designers on the "look and feel" lawsuits).

8. 740 F. Supp. 37 (D. Mass. 1990).

9. *Id.* The *Lotus v. Paperback* lawsuit was settled out of court some months after issuance of the opinion that this article will discuss at length. Under the settlement agreement, Paperback agreed not to appeal the district court's ruling, to take VP-Planner off the market, and to pay \$500,000 to Lotus. Although Paperback still regards the court's ruling as erroneous, it decided that settling the dispute was in its best business interest. See *Lotus Development to Get \$500,000 in Copyright Case*, WALL ST. J., Oct. 18, 1990, at B4; *Lotus Settles Copyright Case*, N.Y. TIMES, Oct. 18, 1990, at C4.

10. See, e.g., Brief Amicus Curiae of Copyright Law Professors, *Lotus Dev. Corp. v. Borland Int'l, Inc.* (Civ. A No. 90-1162-K) (1992) [hereinafter Copyright Professors Amicus Brief] (criticizing *Paperback* for its failure to make a proper kind of § 102(b) inquiry). Paperback did argue that § 102(b) precluded copyright protection for certain aspects of the Lotus interface, but the defense seems not to have been as well-focused as it could have been. See Pretrial Brief of Defendants Paperback Software Int'l and Stephenson Software, Ltd., at 28, *Lotus Dev. Corp. v. Paperback Software Int'l* (Civ. A No. 87-0076-K) (Feb 1, 1990) [hereinafter Paperback Pretrial Brief].

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

interface on which the court based its infringement determination were "expressive" in a copyright sense. It will also criticize the court's analysis of the functionality of the Lotus user interface and the court's failure to appreciate the proper copyright implications of this functionality. Copyright protection for functional writings, such as spreadsheet programs, is traditionally quite "thin."¹² Only exact or near-exact copying will generally be found to be an infringement. Had the court used this approach to judging infringement in *Paperback*, and persuasively explained what it found to be expressive about Lotus' interface, the *Paperback* opinion might have been the landmark opinion for which many were hoping, one on which the law of copyright as applied to computer programs could have been firmly built.

As things stand now, *Paperback* seems to have stirred up more controversy than it has settled. It has met with distinctly mixed reactions in the technical community, in part because people are uncertain about how broad or narrow the court's ruling really was.¹³ Several articles critical of the court's analysis in *Paperback* have already appeared in the law review literature.¹⁴ Moreover, *Paperback* is in direct conflict with a recent decision of the Ninth Circuit Court of Appeals affirming a trial court ruling that there was no expressiveness in the arrangement of commands for a spreadsheet program.¹⁵ There is an opportunity in a

12. See, e.g., *LaST Frontier Report*, *supra* note 6, at 18-19.

13. See sources cited in Pamela Samuelson, *How to Interpret the Lotus Decision (And How Not to)*, 33 COMM. ACM 27 (Nov. 1990).

14. See, e.g., Ronald Abramson, *Why Lotus-Paperback Uses the Wrong Test and What the New Software Protection Legislation Should Look Like*, 7 COMPUTER L. 6 (Aug. 1990); D. Lee Antton & Gary M. Hoffman, *Copyright Protection and Innovation: The Impact of Lotus Dev. v. Paperback Software*, 7 COMPUTER L. 1 (Aug. 1990); Richard H. Stern, *Legal Protection of Screen Displays and Other User Interfaces for Computers: A Problem in Balancing Incentives for Creation Against Need for Free Access to the Utilitarian*, 14 COLUM.-VLA J.L. & ARTS 283 (1990) [hereinafter *Legal Protection of Screen Displays*]; Gerard Lewis, Jr., Comment, *Lotus Dev. Corp. v. Paperback Software Int'l: Broad Copyright Protection For User Interfaces Ignores the Software Industry's Trend Toward Standardization*, 52 U. PITT. L. REV. 689 (1991); Elizabeth G. Lowry, Comment, *Copyright Protection for Computer Languages: Creative Incentive or Technological Threat?*, 39 EMORY L.J. 1293 (1990).

15. *Ashton-Tate Corp. v. Ross*, 916 F.2d 516 (9th Cir. 1990) (command structure for user interface of spreadsheet program held to be "idea" under § 102(b)). The Ninth Circuit decision in *Ross* was decided after the *Paperback* ruling. It does not cite to that opinion. *Paperback* did not cite to the trial court opinion in *Ross*, even though *Ross* was decided before the court's ruling in *Paperback*. See *Ashton-Tate Corp. v. Ross*, 728 F. Supp. 597, 602 (N.D. Cal. 1989), *aff'd*, 916 F.2d 516, 521-22 (9th Cir. 1990). *Ashton-Tate v. Ross* is discussed in Section V.

In a previous article, the author observed that software copyright cases have tended to be either lengthy and elaborately flawed or so cryptic as to provide little guidance on the proper analysis supporting the court's conclusions. See *Reflections on American Copyright Law*, *supra* note 6, at 71-72. That article gave *Whelan*, 797 F.2d 1222, as an example of the lengthy but flawed variety, and *Plains Cotton*, 807 F.2d 1256, as an example of the too cryptic variety. The facts and legal issues presented by *Plains Cotton* were quite similar to those in *Whelan*, but the Fifth Circuit rejected *Whelan's* legal conclusion and reasoning.

new Lotus lawsuit against another of its spreadsheet competitors¹⁶ for Judge Keeton, or perhaps the First Circuit Court of Appeals, to clarify the application of copyright law to user interfaces so that software developers will have the legal guidance they so urgently need.¹⁷

II. THE *LOTUS V. PAPERBACK* DISPUTE

The facts presented by the *Lotus v. Paperback* case are simple and straightforward. Lotus Development Corp. owns a copyright in the very popular spreadsheet program, Lotus 1-2-3. While preparing its own spreadsheet program, VP-Planner, Paperback Software decided to copy several aspects of the Lotus 1-2-3 user interface, including its command hierarchy. Paperback wanted to give consumers already familiar with the Lotus commands and macro facility a lower priced alternative product that could not only achieve the same functionality as Lotus 1-2-3, but also offered some other desirable features that the Lotus product did not then have.¹⁸

Lotus then sued Paperback for copyright infringement. The complaint detailed a number of specific aspects of the Lotus interface that Paperback had allegedly wrongfully copied.¹⁹ These included the instruction, command, and menu language of the 1-2-3 interface, the "structure, sequence, and organization" ("SSO") of its screen displays and

Paperback now joins *Whelan* in the lengthy and elaborately flawed category, and *Ross* joins *Plains Cotton* in the too cryptic category.

Of the two kinds of cases, the lengthy and elaborately flawed variety is the more worrisome because judges in subsequent cases may tend to equate length and elaborateness with sound analysis, which is sometimes not the case. Judges in subsequent cases may find it easier to follow a prior ruling based on a lengthy analysis of the issues than to dissect the lengthy analysis, locate its flaws, and construct the sounder analysis that should have been done. The Ninth Circuit's recent decision in *Brown Bag Software v. Symantec Corp.*, 960 F.2d 1465 (9th Cir. 1992) is an example of a well-reasoned decision applying traditional principles of copyright law to software user interfaces.

16. Some months after the *Paperback* decision, Lotus Development Corp. brought suit against Borland International, Inc. for infringement of the Lotus 1-2-3 copyright because Borland's Quattro Pro spreadsheet program has, in addition to its own "native" user interface for the spreadsheet program, an "emulation" interface that includes the Lotus command hierarchy, although presenting the commands in a different way than the Lotus program does. Judge Keeton, who is presiding over this case as well, has denied cross-motions for summary judgment in the *Borland* case. See *Lotus Dev. Corp. v. Borland Int'l, Inc.* (Civ. A No. 90-11662-K) (Mar. 20, 1992) (memorandum and order) [hereinafter *Borland Order*].

17. See, e.g., NATIONAL RESEARCH COUNCIL, COMPUTER SCIENCE AND TELECOMMUNICATIONS BOARD, INTELLECTUAL PROPERTY ISSUES IN SOFTWARE 4-6, 51-57 (1991) [hereinafter *NRC REPORT*] (concerning the need for more certainty in the legal rules applicable to software).

18. VP-Planner had a data base capability that Lotus 1-2-3 did not then have.

19. The differences between the Lotus and Paperback user interfaces are set forth in *Paperback*, 740 F. Supp. at 70.

sequences of screen displays, the macro commands and syntax of Lotus 1-2-3, and its overall "look and feel."²⁰

Paperback asserted a number of defenses to these claims. Its chief defense was that Paperback had copied only "uncopyrightable" elements of the Lotus program.²¹ Under the umbrella of this main defense, Paperback raised several more specific defenses. One was that only "literal" elements of computer programs (that is, source or object code) are protectable by program copyrights. Because user interfaces are "nonliteral" elements of programs, Paperback asserted that the copying of a user interface could not infringe a program copyright.²² A second was that user interface screen displays are only protectable by copyright law if separately registered with the Copyright Office. Because Lotus had only registered the Lotus 1-2-3 program with the Copyright Office, Paperback argued that the court lacked jurisdiction over the copyright dispute pending before it.²³ A third defense was that the Lotus interface

20. See Complaint of Lotus Development Corp., *Lotus Dev. Corp. v. Paperback Software, Int'l* (D. Mass. Jan. 1987) (Civ. A No. 87-76-K). The "look and feel" claim was among the more controversial claims in the Lotus complaint. "Look and feel" is sometimes used in the technical community to describe the valuable functional behavior of a program that is exhibited when the user interacts with the program via the user interface. See, e.g., NRC REPORT, *supra* note 17, at 53, and *Computer Assoc., Inc. v. Altai*, 775 F. Supp. 544, 559-60 (E.D.N.Y. 1991) (discussing program behavior). As a matter of copyright law, there are a number of reasons to question whether the "look and feel" of programs could be protected by copyright law. See, e.g., *Look and Feel of Software User Interfaces*, *supra* note 7, at 563. Strong opposition exists within the technical community to copyright protection for the "look and feel" of programs. See Samuelson & Glushko, *supra* note 7, at 126. At trial, Lotus seems not to have emphasized the "look and feel" claim. In *Paperback*, the court states that it did not find "look and feel" very helpful in resolving the copyright issues in the case. *Paperback*, 740 F. Supp. at 62.

Whether the "look and feel" of computer programs is protectable by copyright law may be determined in Apple Computer's lawsuit against Microsoft and Hewlett-Packard. See *Apple Computer, Inc. v. Microsoft Corp.* (Civ. No. C-88-20149-VRW), Hearing Transcript at 16 (April 14, 1992) [hereinafter *Apple Hearing Transcript*]. Most of Apple's other claims (as to the protectability of individual element of the Macintosh interface) were excised from the case when the judge ruled against Apple on summary judgment motions by Microsoft and Hewlett-Packard on the ground that these individual elements were either unprotectable by copyright law or licensed by Apple. *Id.* at 83-87.

21. Two other Paperback defenses were laches and estoppel. Paperback asserted that Lotus had known, six months before VP-Planner appeared on the market, that it would look and work like 1-2-3, and yet had waited fourteen and a half months after the Paperback product was released to raise any objections. Lotus also only made its objections known by initiating the lawsuit against Paperback. The court was persuaded that the delay in bringing suit was reasonable, citing cases that had found excusable delays in order to give the plaintiff time to evaluate the matter and prepare to bring suit. *Id.* at 82. It may be, however, that Lotus was waiting to see what the Supreme Court would do with the appeal of the *Whelan* decision, for the Lotus lawsuit against Paperback was filed the day the Supreme Court denied certiorari in that case. For more discussion of the impact of the *Whelan* ruling on the *Paperback* case, see section III.

22. *Paperback*, 740 F. Supp. at 51-55, 73.

23. *Id.* at 79. Although the court emphatically rejected this defense as "frivolous" and based on a "word game" by the defense lawyers, it is worth observing that there was at

was a functional human-machine interface that was uncopyrightable under the "useful article" doctrine of copyright law.²⁴ A fourth defense was that Paperback had not infringed the Lotus copyright because most of what it had copied from the Lotus interface was part of the macro language facility of Lotus 1-2-3. Languages, Paperback asserted, were not copyrightable.²⁵ Paperback's fifth defense was that it had been necessary for Paperback to copy certain aspects of the Lotus interface in order to offer a "compatible" product to consumers. This seems to have been an argument that "idea" and "expression" had "merged" in the certain aspects of the Lotus interface.²⁶ A sixth defense was that user interests in standardization of spreadsheet interfaces overrode the private interests of Lotus in certain aspects of the 1-2-3 interface, which seems to have been a variant of the "merger" argument.²⁷ Paperback seems also to have argued that on public policy grounds, copyright law should not be construed to extend protection to the elements of user interfaces for which Lotus was seeking protection because it would impede incremental innovation in the software industry.²⁸

Judge Keeton wrote an extensive opinion rejecting each of Paperback's defenses. A close examination of this opinion suggests that certain defense strategies backfired and may have prevented the judge from engaging in some inquiries that would have proved more fruitful to the defense's objectives. By making more extreme arguments than were necessary to advance their cause, the defense lawyers failed to focus the court's attention on the proper kind of copyright inquiry for such a case. As a consequence, a number of fundamental principles of copyright law that were quite relevant to the case went unanalyzed. The next section will demonstrate how one of Paperback's principal defenses got the copyright inquiry off on the wrong track.

least one precedent, *Digital Communications Assoc., Inc. v. Softklone Distrib. Corp.*, 659 F. Supp. 449 (N.D. Ga. 1987), that had ruled, after a thoughtful analysis on the relationship between computer programs and user interface screen displays, that such screen displays were separate works from the programs and required separate copyright registrations. At the time Lotus filed the lawsuit against Paperback, the Copyright Office policy had not been clear on the subject of whether user interfaces needed to be separately registered or were covered by the underlying program copyright. For further discussion of this issue, see *infra* Appendix.

24. This defense is discussed *infra* in Section V-B.

25. This defense is discussed *infra* in Section IV-B.

26. This defense is discussed *infra* in Section IV-C.

27. This defense is discussed *infra* at note 126.

28. Both Paperback and Lotus presented opinion evidence concerning the effect its ruling would have on the software industry. The court also took note of an article reporting the results of a survey of user interface designers which reflected strong opposition to the copyright look and feel lawsuits. See Samuelson & Glushko, *supra* note 7, at 127-28. Although the court permitted the proffered evidence to be put into the record, it ultimately concluded that this kind of evidence was irrelevant to the copyright issues in the case. See *Paperback*, 740 F. Supp. at 73-77.

III. THE "UNCOPYRIGHTABILITY" OF "NONLITERAL ELEMENTS" DEFENSE

One of Paperback's principal defenses was that only "literal" elements of computer programs (that is, source and object code) are protectable by a copyright in a computer program. Paperback asserted that because a user interface was a "nonliteral" element of a computer program, it was an "uncopyrightable" element of the program.²⁹ Whether it is appropriate to conceive of a user interface as a "literal" or "nonliteral" element of a computer program is a more complex intellectual problem than the *Paperback* opinion might suggest.³⁰ But for purposes of this section, it will suffice to accept the characterization of the Lotus interface as a "nonliteral" element of the program, and to concentrate, for the moment, on the effects that the "uncopyrightability of nonliteral elements" defense had on the nature of the copyright discourse in the case.

29. The origins of the "uncopyrightable subject matter" defense raised in *Paperback* can be traced to *Baker v. Selden*, 101 U.S. 99 (1879) in which the Supreme Court indicated that the ledger sheets demonstrating Selden's accounting system were not copyrightable subject matter. *Id.* at 107. A more modern characterization for this kind of defense would be that the aspect of the work for which the plaintiff was seeking protection was beyond the scope of the protection available under copyright law to that kind of work. *See, e.g., Brown Bag*, 960 F.2d 1465. Nevertheless, some cases continue to discuss whether certain aspects of copyrighted works are "copyrightable." *See, e.g., Freedman v. Grolier Enters.*, 179 U.S.P.Q. (BNA) 476 (S.D.N.Y. 1974) (discussing the "copyrightability" of a certain notation system for playing cards).

In modern copyright parlance, no "copyrightability" issue is present in a copyright case as long as the work meets the requirements of § 102(a) as an "original work[] of authorship fixed in [a] tangible medium of expression." *See* 17 U.S.C. § 102(a) (1988). Paperback was not arguing that there was no original expression in the Lotus program, nor that the program was not fixed enough to be copyrighted, nor that the program was otherwise disqualified from being considered a protectable "work of authorship" under the statute. Only these are today considered to be "copyrightability" challenges. Even if the defense lawyers in *Paperback* mischaracterized the nature of their defense, it would have been possible for the court to have straightened matters out, but instead the court adopted the defense's characterization of the defense, and from that flowed the errors discussed in the rest of the article.

One reason that *Baker v. Selden* is in need of an updated interpretation is to clarify this distinction between "copyrightability" and "beyond the scope of protection" defenses. The Supreme Court not only speaks of Selden's ledger sheets as uncopyrightable subject matter, but also refers to the "uncopyrightability" of news reports and advertisements. *See* 101 U.S. at 105-06. These two kinds of works were later ruled to be "copyrightable subject matter." *See, e.g., Bleistein v. Donaldson Lithography Co.*, 188 U.S. 239, 251 (1903) (advertisement was copyrightable) and *International News Serv. v. Associated Press*, 248 U.S. 215, 234 (1918) (reflecting view that Associated Press newspapers could have been copyrighted).

30. *See infra* Appendix. Notwithstanding the court's apparent adoption of Paperback's assertion that the Lotus interface was a "nonliteral" element of the program, the opinion seems to rely on near-verbatim copying to support the conclusion of infringement. *See Paperback*, 740 F. Supp. at 70. Section V-C *infra* will discuss how *Paperback* might have been analyzed had the court been consistent in its treatment of the Lotus interface as a nonliteral element of the program; it will also analyze *Paperback* as a "literal copying" case.

One very important effect of this defense was to narrow dramatically the analytic focus of the case. The copyright discourse shifted from one about the proper scope of the Lotus copyright in the context of an infringement analysis to one about the "copyrightability" of "nonliteral" elements of computer programs. The "nonliteral elements" defense affected virtually everything about the case, including the manner in which the judge "phased" the trial of the case and the test the court adopted for judging the copyright issues in the case.³¹

The "nonliteral elements" defense was very risky because if the court could be persuaded that *any* nonliteral element of *any* computer program could properly be protected by copyright law, the defense would founder. Because several previous cases had ruled that some nonliteral elements of computer programs could be protected by copyright law, including the widely cited Third Circuit Court of Appeals decision, *Whelan Assoc. v. Jaslow Dental Lab*,³² a defense based on the uncopyrightability of nonliteral elements of programs was, as a strategic matter, almost suicidal.

A. The Nonliteral Elements Defense

Paperback made much the same argument in the *Lotus* case that Jaslow had made in the *Whelan* case. Support for the proposition that copyright protection is only available for source and object code of computer programs is said to come from three sources: the statutory definition of computer programs,³³ the absence of legislative history indicating an intent to bring nonliteral elements of programs within the

31. The first phase of the trial was supposed to determine "whether and to what extent plaintiff's computer spreadsheet program, Lotus 1-2-3, was copyrightable." 740 F. Supp. at 42. Phase two was to be a jury trial on issues of fact relating to whether Paperback had copied protected expression from the Lotus program, including from the source or object code of the Lotus program. *Id.* After ruling that the Lotus interface was a copyrightable component of the program, *id.* at 68, the court decided that there was no genuine issue of fact requiring a phase two proceeding. The court observed that Paperback had conceded it had copied many elements of the Lotus interface. Because these elements were substantial components of the Lotus program, the court found infringement. The test the court devised for determining infringement in *Paperback* is discussed *infra* in Section III-C.

32. 797 F.2d 1222 (1986). *Whelan* has been cited with approval in several cases for the proposition that nonliteral elements of programs can be protected by copyright. *See, e.g., Manufacturers Technologies*, 706 F. Supp. at 992; *Pearl Sys. v. Competition Elec.*, 8 U.S.P.Q.2d (BNA) 1520, 1524 (S.D. Fla. 1988); *Broderbund*, 648 F. Supp. at 1133.

33. The copyright statute defines "computer program" as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101. The user interface of a program can accurately be described as among the "results" that can be generated when a program is executed in a computer. The user interface itself is not among the set of statements or instructions of a program that can be processed in the computer to bring about results. *See, e.g., Dennis S. Karjala, Copyright, Computer Software, and the New Protectionism*, 28 JURIMETRICS J. 33 (1987); *Legal Protection of Screen Displays*, *supra* note 14, at 350. The statute does not say that the "results" of the program are protectable by copyright law.

scope of copyright protection,³⁴ and judicial decisions that had denied copyright protection to some nonliteral elements of programs.³⁵ In addition, the defendants in both *Whelan* and *Paperback* argued there was a need for a "bright line" standard, such as that between literal and nonliteral elements of programs, so that software developers could know what they could lawfully do.³⁶ A narrow scope of copyright protection for programs was also claimed to be desirable so as not to interfere with the kind of incremental development that characterizes software innovation.³⁷

B. *Whelan's* Rejection of One "Bright Line" Test and Adoption of Another

The Third Circuit rejected the "nonliteral elements" defense in *Whelan*. Because nonliteral elements of "literary works" have long been protected by copyright law,³⁸ the court thought it appropriate for such

34. There is no consensus among former CONTU Commissioners and staff members about whether CONTU thought that nonliteral elements of programs, such as their "structure, sequence, and organization," would be protectable by copyright law. The Executive Director of the CONTU staff and the chairman of the CONTU subcommittee on the copyrightability of computer programs think not. See Kenneth A. Liebman, Salem M. Katsch & David D. Leitch, *Back to Basics: A Critique of the Emerging Judicial Analysis of the Outer Limits of Computer Program "Expression,"* 2 COMPUTER L. 1 (Dec. 1985) (discussing CONTU Commissioner Miller's and Director Levine's views); Arthur J. Levine, *Comment on Bonito Boats Follow-Up: The Supreme Court's Likely Rejection of Nonliteral Software Copyright Protection,* 6 COMPUTER L. 29 (July 1989). Two other CONTU Commissioners disagree. See E. Gabriel Perle, Christopher Meyer, and Victor Siber, *Bonito Boats Redux,* 7 COMPUTER L. 1 (Feb. 1990) (discussing the views of Commissioner Perle and the now-deceased CONTU Commissioner Nimmer on CONTU's intent with regard to protection of nonliteral elements of programs). Both sides have found something in the CONTU Report to support their views on this matter. *Id.* It is, however, worth noting that all of the examples of wrongful copying that CONTU discusses are "literal" copying examples. See CONTU REPORT, *supra* note 5, at 22.

Because of this nonconsensus, the CONTU Report is not an especially helpful source of information about the Commission's understanding of what it was recommending. There is little else in the legislative history of the 1980 software amendments to indicate Congressional intent on this issue. See *CONTU Revisited*, *supra* note 5, at 666 & n.9. The *Paperback* opinion indicates that CONTU had not explicitly addressed the central issue presented by the case then before the court. 740 F. Supp. at 50. It nonetheless quotes from Nimmer's views, and merely notes the differing views of Miller and Levine. *Id.* at 51.

35. See, e.g., *Synercom Technology, Inc. v. University Computing Co.*, 462 F. Supp. 1003, 1011 (N.D. Tex. 1978); *Plains Cotton*, 807 F.2d 1256.

36. See *Whelan*, 797 F.2d at 1237-38; *Paperback*, 740 F. Supp. at 73.

37. *Whelan*, 797 F.2d at 1237-38; *Paperback* 740 F. Supp. at 77-79. *Paperback* treats this as a "policy" argument rather than as a principle of copyright law applicable to functional writings. See *infra* Section V.

38. *Whelan*, 797 F.2d at 1234; *Paperback*, 740 F. Supp. at 51. Although *Paperback* acknowledges that the statutory definition of a "literary work" is far broader than "works of literature" (which the term would, in ordinary discourse, connote), 740 F. Supp. at 51, it nonetheless cites a string of cases involving works of literature and other artistic and fanciful works in support of the proposition that nonliteral elements of copyrighted works can be protected by copyright law. *Id.* at 51-52. As Section V-A will show, functional

elements to be protected in programs as well. It rejected the "bright line" standard argument as inconsistent with the Congressional intent to have courts use the traditional idea/expression distinction,³⁹ and the incremental innovation argument as one which, if accepted, would provide too little incentive to invest in software development.⁴⁰ Consequently, it concluded that nonliteral elements of a program, such as its "structure, sequence, and organization" could be protected by copyright law.⁴¹

Although rejecting the "bright line" standard proposed by the defense, the Third Circuit in *Whelan* proffered a highly protectionistic one in its stead: "the purpose or function [of a program is] the work's idea, and everything that is not necessary to that purpose or function [is] part of the expression."⁴² Necessity, the court declared, would be tested by determining whether more than a small number of ways existed to achieve the function. Although this test has been extensively criticized by commentators,⁴³ it has been used with some frequency in software copyright cases,⁴⁴ for it has the apparent virtue of all "bright line" tests of providing judges faced with complicated legal arguments about factually complex phenomena with a simple basis for making distinctions that will resolve the case before them.⁴⁵

writings, such as spreadsheet programs, generally have a much narrower scope of copyright protection than literature, drama, and other artistic and fanciful works.

39. The aspect of the CONTU Report that both *Whelan* and *Paperback* considered to be a reliable reflection of Congressional intent is that which expresses confidence that the courts could draw the proper line between "idea" and "expression" in computer program cases. See CONTU REPORT, *supra* note 5, at 18-23. Had CONTU and Congress intended for the line between copyrightable and uncopyrightable elements of programs to be drawn between their literal and nonliteral elements, the courts reasoned they would have said so. *Whelan*, 797 F.2d at 1241; *Paperback*, 740 F. Supp. at 54, 73.

40. *Whelan*, 797 F.2d at 1237. For similar reasoning in *Paperback*, see 740 F. Supp. at 75-76.

41. *Whelan*, 797 F.2d at 1240. The *Whelan* test was rejected by the Court of Appeals for the Fifth Circuit in *Plains Cotton*, 807 F.2d 1256. *Whelan's* "SSO" concept was rejected as "technically incoherent" and its test for infringement was rejected as unsound copyright law in *Computer Associates, Inc. v. Altai*, 755 F. Supp. 544, 559-560 (E.D.N.Y. 1991).

42. *Whelan*, 797 F.2d at 1236.

43. See, e.g., Karjala, *supra* note 33, 76-79; *Copyright Protection for Application Programs*, *supra* note 6, at 1045; *LaST Frontier Report*, *supra* note 6, at 90.

44. See, e.g., *Broderbund*, 648 F. Supp. at 1133; *Pearl Systems*, 8 U.S.P.Q.2d (BNA) 1520 (both applying the *Whelan* test).

45. *Paperback* indicates that despite their superficial appeal, "bright line" rules often result in injustice. To reach just results, the opinion states, it is necessary to carefully weigh the facts and circumstances of each case. 740 F. Supp. at 73. Yet, by adopting the *Whelan* approach, the court in *Paperback* seems to have unconsciously slipped into the error of a "bright line" standard, one that is inconsistent with copyright statute and caselaw, as will be demonstrated further in sections IV and V.

C. *Paperback's* Use of the *Whelan* Test

Although the court in *Paperback* purported to model its legal test in the case on Judge Learned Hand's "patterns of abstraction" test,⁴⁶ a close examination of the *Paperback* opinion reveals that the court actually applied the Third Circuit's *Whelan* test, thereby taking that court's shortcut through what would otherwise have been a dense thicket of copyright issues in the case.

Paperback sets forth its "legal test for copyrightability" which the court said it intended to use to judge the protectability of various aspects of the Lotus program.⁴⁷ The first "element" of this test, said to be derived from Judge Learned Hand's "patterns of abstraction" test was that "the decisionmaker must focus upon alternatives that counsel may suggest, or the court may conceive, *along the scale from the most generalized conception to the most particularized*, and choose some formulation—some conception or definition of the 'idea'—for the purpose of distinguishing between the idea and its expression."⁴⁸ The second "element" of the test was that "the decisionmaker must focus upon whether an alleged expression of the idea is limited to elements essential to expression of *that* idea (or is one of only a few ways of expressing the idea) or instead includes identifiable elements of expression not essential to every expression of that idea."⁴⁹ The third was "having identified elements of expression not essential to every expression of the idea, the decisionmaker must focus on whether those elements are a substantial part of the allegedly copyrightable 'work,'" which the court pointed out was to be judged on a qualitative, not just a quantitative, basis.⁵⁰

Yet, in the section of the *Paperback* opinion in which the court purports to apply its legal test to the facts of the case,⁵¹ the court uses an approach that is much more like that of *Whelan*. It begins by identifying "the idea" of the Lotus program as that of "an electronic spreadsheet."⁵² This was, of course, unprotectable by copyright law. More specific aspects of the program, such as the rotated "L" creating the spreadsheet

46. See *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930).

47. The *Paperback* test is a highly idiosyncratic and erroneous analytic procedure for assessing the copyright issues raised by the *Paperback* case. See Copyright Professors Amicus Brief, *supra* note 10, at 1-2. A more appropriate way to frame the copyright issues in *Paperback* would have been to inquire, as part of a copyright infringement analysis, whether the aspects of the Lotus interface for which Lotus was seeking protection were properly considered within the scope of copyright protection available for the Lotus program. That is, were these aspects part of the protectable "expression" of the program and had Paperback copied that expression from the Lotus program? See *Brown Bag*, 960 F.2d 1465; PAUL GOLDSTEIN, COPYRIGHT PRINCIPLES, LAW & PRACTICE, ch. 7 (1989).

48. 740 F. Supp. at 60 (emphasis in the original) (quoting *Nichols*, 45 F.2d at 121).

49. *Paperback*, 740 F. Supp. at 61 (emphasis in the original).

50. *Id.*

51. *Id.* at 65-68.

52. *Id.* at 65.

grid, the "/" key for invoking the menu of commands, and the command structure, were treated, in accordance with the *Whelan* test, as presumptively expressive. Unless the court found them to be necessary elements of a spreadsheet program, they would be established as expressive elements.⁵³

Although recognizing that there were some spreadsheet programs that did not use the rotated "L" as a spreadsheet grid or the "/" to invoke the menu of commands, the court in *Paperback* concluded that there was only a limited number of options for accomplishing the functions these details served.⁵⁴ The court thus concluded that they were "necessary" elements of spreadsheet programs, and consequently, were instances in which idea and expression had "merged," and what would otherwise have been "expression" had become "idea."⁵⁵

Other elements of the Lotus user interface, particularly its command structure, were found by the court to be unnecessary to accomplishing spreadsheet program functions because it was possible to design a spreadsheet program different from Lotus' as to these features, as could be seen by looking at the user interfaces of a number of other spreadsheet programs.⁵⁶ Having determined that these aspects of the Lotus interface were "expression" because they flunked the "necessity" test, the remaining question the court addressed was whether they were substantial components of the Lotus program, to which the court thought it "incontrovertible" that the answer must be yes.⁵⁷

53. *See id.* at 66-67.

54. *Id.* Stern questions the factual basis of the court's assertion that there were a limited number of ways to achieve these functions. *Legal Protection of Screen Displays, supra* note 14, at 335. A better copyright analysis of these details of the Lotus interface, as with *Paperback's* use of "+" to represent addition and "-" to represent subtraction, would have been that these elements, because so commonly found in spreadsheet programs, had become conventional elements of such programs, which should be treated in the same manner as *scenes a faire* typically are in copyright law—as unprotectable by copyright. *See, e.g., Brown Bag, 960 F.2d 1465; Apple Hearing Transcript, supra* note 20, at 84-86. The court in *Paperback* seems to have avoided this way of analyzing the copyright status of these details because of the deep disfavor with which it viewed *Paperback's* "standardization" defense. *See infra* note 126.

55. *Paperback, 740 F. Supp.* at 66-67.

56. The court listed seven spreadsheet programs that had different menu structures from Lotus 1-2-3, but also said that the idea of a menu structure for a spreadsheet program "could be expressed in a great many if not literally unlimited number of ways." *Paperback, 740 F. Supp.* at 67-68. For similar statements in *Whelan, see 797 F.2d* at 1239. What does not seem to have occurred to the court in either case is that the different structures of other programs might reflect use of different "systems" or "methods of operation," rather than differences in expression.

57. *Paperback, 740 F. Supp.* at 68. The court stated, in support of this proposition, that the Lotus interface was "its most unique element, and is the aspect that has made 1-2-3 so popular." *Id.* Further proof of its substantiality was said to be that the defendants had bothered to copy it. *Id.*

What may have caused the court in *Paperback* to use the *Whelan* test instead of a more traditional approach, such as Hand's abstractions test, was that the court got almost no help from the lawyers in conceptualizing what an abstractions scale for a spreadsheet program might look like, and where on such a scale the Lotus interface or various aspects of it might be found.⁵⁸ Lotus' lawyers were apparently arguing that everything about its user interface was "expressive" (because of different ways functions could be done) and Paperback's lawyers were arguing that all aspects of the user interface were "ideas."⁵⁹ So, instead of building a fairly elaborate scale of abstractions, as Hand's test would have suggested was proper,⁶⁰ the court ended up with a dichotomy in which the only perceived generality at the "idea" pole was that of an electronic spreadsheet, and all of the particularities of the Lotus interface were lumped together at the "expression" pole, as would be the result of applying the *Whelan* test.

D. The Errors of the *Whelan* Test

Even if one has no quarrel with the conclusion of *Whelan* and *Paperback* that nonliteral elements of computer programs can sometimes be protected by copyright law, one can nonetheless object to the *Whelan* test for infringement as one "inconsistent with the copyright statute, the copyright caselaw, and traditional principles of copyright law."⁶¹

The principal criticism that has been leveled at the *Whelan* test is that it takes an overly narrow view of what copyright law considers to be an "idea."⁶² On its face, the test begins with the presumption that there is only one idea to be found in every computer program, and that all else in the work is expression unless a necessity test takes it out of the expression category and propels it into the idea category.⁶³ It also fails to distinguish

58. *Paperback*, 740 F. Supp. at 62.

59. *Id.*

60. *See id.* at 60 ("Upon any work, and especially upon a play, a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out," (quoting *Nichols*, 45 F.2d at 121) (emphasis added)).

61. Copyright Professors Amicus Brief, *supra* note 10, at 2.

62. *See, e.g., LaST Frontier Report, supra* note 6, at 20.

63. The *Whelan* test would give "functional works," such as Lotus' spreadsheet program, a far broader scope of copyright protection than is available for artistic and fanciful works, a result out of keeping with traditional principles of copyright law. *See* discussion *infra* in Section V.

The *Whelan* test also contributes to confusion about what is protectable by copyright in a program and what is protectable by patents. The general purpose or function of a program is probably no more patentable than it is copyrightable. A particular way of achieving a program function, however, seems now to be patentable. *See infra* note 136. Only a cursory study of patent law is required to discern that patent specifications routinely describe the different ways that the general function(s) of the invention had been accomplished in the prior art as a basis for establishing the novelty of the claimed inventive new way to do it. Thus, the existence of other ways of doing something is as

between higher level and lower level abstractions, which Hand's abstractions test would do.⁶⁴

The *Whelan* test is particularly unsuitable as a test for copyright infringement for computer programs because programs are often assemblages of components that could be packaged separately instead of being combined together. A spreadsheet program, for example, might include a calculation component, a graphing component, and a data base component. If the *Whelan* test is applied to a software package combining these functions into one program, "the idea" in the program will be that of an electronic spreadsheet, and the three components, because they are more specific details, would be treated as presumptively expressive. If, however, the three components were packaged as separate programs, graphing would now be "the idea" of the graphing program, and only lower level details would be presumptively expressive (even though some of them too may well be methods or processes). This demonstrates how much of a "word game" the *Whelan* test can be.⁶⁵

More importantly, the *Whelan* test fails to take account of the fact that the term "idea" in copyright parlance is not confined in its meaning to abstract generalized conceptions, such as the general purpose or function of a program. Rather, it is a metaphor used in copyright law to describe the unprotectable elements in a copyrighted work, that is, the things that are beyond the scope of protection available to the work.⁶⁶

In particular, the *Whelan* test fails to consider the full text of § 102(b) which indicates that such things as processes, procedures, systems and methods of operation are unprotectable by copyright law even when embodied in the text of a copyrighted work.⁶⁷ It also fails to recognize

likely to indicate that there are other patentable methods for achieving the same function as it is to indicate that there are other copyrightable nonliteral expressions of how to do it.

64. See *LaST Frontier Report*, *supra* note 6, at 20.

65. The court in *Paperback* criticized the defense lawyers for engaging in "word games" in connection with several of its defenses. *Paperback*, 740 F. Supp. at 71-73, 79. Yet the court seems to have engaged in some word games of its own concerning the meaning of § 102(b). See also *Borland Order*, *supra* note 16, at 28-32 (exhibiting an unwillingness to give the words of § 102(b) more content than as a restatement of the idea/expression distinction).

66. *Paperback* seems at one point to recognize this, for it states that the idea/expression distinction "embraces also the process-expression, method-expression, and useful-expressive distinctions." 740 F. Supp. at 53. But the court fails to carry through with this kind of analysis.

67. Section 102(b) is duly quoted in a background section of *Paperback*. *Id.* at 49. Although occasionally referred to in a cursory manner thereafter, neither this section nor its contents (except that referring to the unprotectability of "ideas") is discussed in the subsection of the opinion on functionality issues, *id.* at 54-58, in the subsection describing the legal test for copyrightability to be used in the case, *id.* at 59-62, or in the sections that analyze the "copyrightability" of various elements of the Lotus interface and *Paperback's* copying of them, *id.* at 63-70. Given that the court in *Paperback* was so insistent on the need to follow Congressional "mandates," *id.* at 53, 58, its failure to conduct any meaningful inquiry into § 102(b) issues is very surprising.

that it is in the nature of processes, procedures, systems and methods of operation to have constituent elements, and that the caselaw interpreting § 102(b) or from which its meaning is derived have frequently held constituent elements of a system unprotectable by copyright.⁶⁸

The *Whelan* test also ignores the legislative history of the 1976 Act which indicates that § 102(b) was being added to the statute in part to ensure that copyright protection for programs would not be construed too broadly, that is, to ensure that such things as the "methods" and "processes" of "a program" would not be protected by copyright law.⁶⁹ The text of 102(b), as well as the caselaw from which it was derived and that properly interprets it, demonstrates unequivocally that not all "nonliteral" elements of a program are to be treated equally under copyright law, as they are under the *Whelan* test, and not all should be presumed to be expressive. Because there may well be more than a few methods or systems of achieving some general purpose, the mere existence of alternatives does not demonstrate that a nonliteral aspect of a computer program is "expressive" enough for copyright protection to be available for it.⁷⁰

The proper inquiry in copyright cases involving computer programs must be much broader than the highly protectionist *Whelan* "bright-line" test permits. *Paperback* erred in applying the *Whelan* test for software copyright infringement. The remainder of this article will set forth the kind of the inquiry the court might have made had it not been distracted from doing so by *Paperback's* nonliteral elements defense. Section IV will discuss the kind of copyright inquiry the court should have made about whether any aspects of the Lotus interface that *Paperback* copied—most particularly, the Lotus command structure—were constituent elements of a system for managing spreadsheet functions or for constructing macros. Section V will discuss the kind of inquiry the court should have made about what was expressive in the Lotus interface and whether expressive elements were appropriated by *Paperback*.

68. See Copyright Professors Amicus Brief, *supra* note 10, at 7. See also cases cited *infra* in Section IV.

69. In a background section of the opinion, 740 F. Supp. at 49, *Paperback* quoted the relevant passage from the House Report (see *supra* note 4), but made no effort to give it content. The terms "methods" and "processes" are emphasized in the text because the legislative reports' use of the plural expression demonstrates quite clearly that Congress thought there would be more than one of them per program.

70. *Baker v. Selden*, discussed *infra* in Section IV-A, is one example of this principle. See also *Bibbero Sys. v. Colwell Sys.*, 893 F.2d 1104 (9th Cir. 1990) (existence of alternative arrangements did not mean that plaintiff's form was "expressive" enough to be protectable by copyright law).

IV. WAS THE LOTUS COMMAND STRUCTURE PART OF THE LOTUS SPREADSHEET OR MACRO SYSTEM?

Paperback contains lengthy discussions of the "idea/expression" distinction,⁷¹ but no analysis of whether any aspects of the Lotus interface might have been part of an unprotectable "system" for managing spreadsheet functions or for constructing macros, even though there are statements in the opinion strongly suggesting that this was the case.⁷² Section 102(b) indicates that "systems" are as unprotectable as abstract ideas, in harmony with a long line of copyright cases that go virtually unmentioned in the *Paperback* opinion.⁷³ The Supreme Court's *Baker v. Selden* decision is among the numerous precedents that hold that when an arrangement of words is a constituent part of a system, the arrangement is not within the scope of copyright protection for the work, no matter

71. See 740 F. Supp. at 53-61, 65-68.

72. The strongest of these statements can be found *id.* at 65, discussed *infra* in the text accompanying note 114. See also *id.* at 67 (comparing the "menu command system" of Visicalc with that of Lotus); *id.* at 78 (referring to use of "/" to invoke the "menu command system" and Lotus' "macrocommand facility" and "command facility").

73. See, e.g., *Affiliated Enters. v. Gruber*, 86 F.2d 958 (1st Cir. 1936) (promotional system not protectable by copyright); *Brief English Sys. v. Owen*, 48 F.2d 555 (2d Cir.), *cert. denied*, 283 U.S. 858 (1931) (shorthand system not protectable by copyright on booklets explaining it); *Chautauqua Sch. of Nursing v. National Sch. of Nursing*, 238 F. 151 (2d Cir. 1916) (twelve step hypodermic injection procedure not protectable by copyright in lecture); *Long v. Jordan*, 29 F. Supp. 287 (N.D. Cal. 1939) (old age pension system not protectable by copyright for pamphlet); *Burk v. Johnson*, 146 F. 209 (8th Cir. 1906) (system for organizing mutual burial associations not protected by copyright in pamphlet); *Amberg File & Index Co. v. Shea Smith & Co.*, 82 F. 314 (7th Cir. 1897) (indexing system not copyrightable); *Arica Inst. v. Palmer*, 1991 Copyright L. Rptr. (CCH) ¶ 26,712 (S.D.N.Y. 1991) (spiritual system not protectable by copyright in training manuals); *Kepner-Tregoe, Inc. v. Carabio*, 203 U.S.P.Q. (BNA) 124 (E.D. Mich. 1979) (system for teaching problem-solving techniques not protectable by copyright); *Freedman*, 179 U.S.P.Q. (BNA) 476 (notation system for playing cards not protectable by copyright); *Aldrich v. Remington Rand*, 52 F. Supp. 732 (N.D. Tex. 1942) (tax recording system not protectable by copyright); *Muller v. Triborough Bridge Auth.*, 43 F. Supp. 298 (S.D.N.Y. 1942) (traffic separation system not within scope of copyright on drawings); *Seltser v. Sunbrock*, 22 F. Supp. 621 (S.D. Cal. 1938) (roller derby system not protectable by copyright in book); *Stone & McCarrick, Inc. v. Dugan Piano Co.*, 210 F. 399 (E.D. La. 1914) (system of salesmanship not protectable by copyright on instruction manual); *Simms v. Stanton*, 75 F. 6 (N.D. Cal. 1896) (physiognomy classification systems not protectable by copyrights in books); *Griggs v. Perrin*, 49 F. 15 (C.C.N.D.N.Y. 1892) (stenography system not protected by copyright in book). See also *Healthcare Affiliated Servs. v. Blue Cross of W. Penn.*, 701 F. Supp. 1142 (W.D. Pa. 1988) (use of similar systems and methods in computer program not copyright infringement); *Fishing Concepts, Inc. v. Ross*, 226 U.S.P.Q. (BNA) 692 (D. Minn. 1985) (processes in computer program not protectable by copyright); *Midway Mfg. v. Bandai-America, Inc.*, 546 F. Supp. 125 (D.N.J. 1982) (game rules not protectable by copyright).

None of these decisions is discussed in *Paperback*. Even after eleven copyright law professors urged the court in the *Lotus v. Borland* case to look to *Baker v. Selden* and its progeny in interpreting what the term "system" in § 102(b) means, see Copyright Professors Amicus Brief, *supra* note 10, at 5-9, the court did not do so, saying that *Baker v. Selden* was inapposite because it was decided before computer programs were invented and because the Court didn't have the benefit of the wisdom of Judge Hand's "patterns of abstraction" test. *Borland Order*, *supra* note 16, at 32-33.

how valuable or innovative it might be.⁷⁴ In addition to reviewing *Baker v. Selden* as a representative of the word-arrangement-as-system-elements cases, this section will suggest that, properly construed, Paperback's "language" and "compatibility" defenses were § 102(b) "system" defenses that the court should have taken more seriously.

A. *Baker v. Selden*: The Arrangement of Words As Part of a System

Baker v. Selden is an important precedent to consider in assessing the *Lotus v. Paperback* dispute, not only because it is the venerable Supreme Court decision that ruled that constituent elements of systems embodied in copyrighted works are not protected by the copyright,⁷⁵ but also because it seems to be the only prior copyright case to have involved a claim of copyright infringement based on spreadsheet similarities. It is therefore somewhat surprising how little attention *Paperback* gives to *Baker v. Selden*. That case is discussed in two sentences in the middle of a long paragraph in a part of the opinion remote from the analysis of the merits of Lotus' claim and of Paperback's defenses.⁷⁶ Although *Paperback* makes a passing reference comparing the Lotus electronic spreadsheet to paper spreadsheets,⁷⁷ the larger similarities between the two cases, in terms of their facts, the parties' legal contentions, and the Court's ruling seem to have escaped the court's attention.

74. A number of the "system" decisions involve arrangements of words as elements of the system. *Arica Institute*, 1991 Copyright L. Rptr. (CCH) ¶ 26,712, is a recent example of a case in which the arrangement of words was considered to be outside the scope of copyright because of its role as an element of a system. Palmer had written a book discussing the Arica philosophy, and reproduced diagrams closely resembling those in Arica's manuals. Arica sued for copyright infringement. Palmer defended by asserting that what she had copied were elements of Arica's system for curing ego fixation problems, which were illustrated by the diagrams consisting of nine-pointed stars surrounded by a circle. Each point of the stars was inscribed with a word or phrase that symbolized a component of the system. The ordering of the components was also part of the system. The court stated: "The copyright laws do not confer a monopoly on Arica in the method of describing a particular and interrelated set of characteristics or traits." *Id.* at 24,159.

75. CONTU described *Baker v. Selden* as a "venerable case." CONTU was confident that *Baker v. Selden* and cases like it would provide guidance to the courts in attempting to distinguish what in a program should be regarded as idea or expression. CONTU REPORT, *supra* note 5, at 18.

76. See *Paperback*, 740 F. Supp. at 53-54. The "legal test for copyrightability" is found in the opinion, *id.* at 59-62, and the discussion of the protectable elements of the Lotus interface is found at 65-68.

77. *Id.* at 63.

1. PAPERBACK ON BAKER V. SELDEN

Paperback begins its brief discussion of *Baker v. Selden* by referring to it as a "seminal case,"⁷⁸ and goes on to say that it had held "that the text of a book describing a special method of double-entry accounting on paper spreadsheets . . . was copyrightable *expression*, but that the . . . *idea* of this particular kind of double-entry bookkeeping, was not."⁷⁹ The Supreme Court's statement of its holding in the case was importantly different from the court's description of it in *Paperback*. The Court actually said that Selden's copyright protected his "explanation" of the accounting method, but not the useful "art" (that is, the bookkeeping method or system) explained in the book.⁸⁰ This correct formulation of the rule of *Baker v. Selden* makes clear something that is sometimes forgotten about the case: that *Baker v. Selden* is fundamentally a case about the unprotectability of the functional content of written works, and the right of others to copy that content to make use of it.⁸¹ To speak of the

78. *Id.* at 54.

79. *Id.* (emphasis in the original).

80. *Baker*, 101 U.S. at 105.

81. *Baker v. Selden* has truly been a "seminal case," for from it have grown a number of important doctrines of American copyright law: (1) that the scope of protection for writings embodying functional content is quite narrow, for the functional content is not protectable by copyright, *see, e.g.*, *Continental Casualty Co. v. Beardsley*, 253 F.2d 702, 705-06 (2d Cir.), *cert. denied*, 358 U.S. 816 (1958) and *Kepner-Tregoe*, 203 U.S.P.Q. (BNA) 124, 130-32; (2) that constructing a useful article depicted in a copyrighted work does not infringe the copyright, *see, e.g.*, *Muller*, 43 F. Supp. 298, 300; (3) that blank forms are not copyrightable, *see, e.g.*, *Bibbero*, 893 F.2d 1104, 1106; (4) that when there are significant constraints on the manner in which an idea can be expressed, even using the same expression will not be infringing as an instance in which idea and expression are said to be merged, *see, e.g.*, *Morrissey v. Proctor & Gamble*, 379 F.2d 675, 678-79 (1st Cir. 1967); (5) that when useful elements of a copyrighted work must be copied in order to be used by others, no copyright infringement should be found, *see, e.g.*, *Kepner-Tregoe*, 203 U.S.P.Q. (BNA) 124, 131; and (6) that which is within the subject matter of utility patent law is outside the subject matter of copyright, and that it would be a fraud on the public to give copyright protection to that which has not satisfied the standards and procedures required by the patent system or to that which is the subject of an expired patent. *See, e.g.*, *Brief English Sys.*, 48 F.2d 555, 556; *Muller*, 43 F. Supp. 298, 299; *Korzybski v. Underwood & Underwood, Inc.*, 36 F.2d 727, 729 (2d Cir. 1929).

Some regard this latter proposition to have been called into question by dicta in *Mazer v. Stein*, 347 U.S. 201, 217 (1954) ("Neither the Copyright Statute nor any other says that because a thing is patentable it may not be copyrighted") and by *In re Yardley*, 493 F.2d 1389, 1393 (C.C.P.A. 1974) (design patent can issue on copyrighted work). *See, e.g.*, 1 MELVILLE B. NIMMER & DAVID NIMMER, NIMMER ON COPYRIGHT, § 2.19 (1991) [hereinafter NIMMER ON COPYRIGHT]. Both of these decisions, however, involved design patent and copyright issues, an area in which Congress may have contemplated some degree of overlap in coverage. There is, however, no case holding that utility patents and copyrights can protect the same aspect of the same work. Several cases, *Baker v. Selden* among them, express the contrary view. *See also* *Taylor Instrument Co. v. Fawley-Brost Co.*, 139 F.2d 98, 99 (7th Cir. 1943), *cert. denied*, 321 U.S. 785 (1944). *Mazer* itself cites approvingly not only *Baker v. Selden*, but two other cases in which the courts observed that plaintiffs should have sought a patent if they wanted to protect the innovation which they were trying to protect through copyright law. *Mazer*, 347 U.S. at 217-18. The issue whether utility

case as concerning only the unprotectability of abstract ideas oversimplifies the Court's ruling.⁸²

A more serious mischaracterization of *Baker v. Selden* appears in the next sentence of *Paperback*: "The Court thus concluded that Baker did not

patents and copyrights can protect the same aspect of computer programs is currently hotly debated among intellectual property lawyers. See, e.g., Perle, Meyer, and Siber, *supra* note 34, at 7-8 and Toedt, *Bonito Boats Follow-up*, 6 COMPUTER L. 28 (July 1989).

This author contends that, notwithstanding the dicta quoted above, *Mazer v. Stein* did not effect any significant change in the holding of *Baker v. Selden*. The Court in *Mazer* merely decided that the copyright in a statuette (which qualified for protection as a "work of art") was not invalidated because of the subsequent reproduction of it to serve as a base for a lamp. The statuette served the same aesthetic function as a lamp base as it had as a free-standing sculpture. The Court in *Mazer* made clear that had the sculpture served a utilitarian function, or had any utilitarian function been intermingled with its aesthetic function, it would have regarded the matter differently. 347 U.S. at 212-213. The 1976 Act has codified this aspect of *Mazer*. See 17 U.S.C. § 101 (definitions of "pictorial, sculptural, and graphic work" and "useful article"). See also 17 U.S.C. § 113. If anything, *Mazer* clarified that *Baker v. Selden* should be understood as a case concerning the appropriate scope of protection for copyrighted works, an issue generally arising in the course of an infringement determination. See Reichman, *supra* note 2, at 693-95.

82. A number of decisions have recently cited *Baker v. Selden* as though the only proposition for which it stands concerns the unprotectability of abstract ideas. See, e.g., *Cable/Home Communication Corp. v. Network Prods., Inc.*, 902 F.2d 829, 842 (11th Cir. 1990); *Whelan Assoc. v. Jaslow Dental Lab.* 797 F.2d 1222, 1236 (3d Cir. 1986); *Toro Co. v. R & R Prods. Co.*, 787 F.2d 1208, 1212 (8th Cir. 1986); *Apple Computer, Inc. v. Formula Int'l, Inc.*, 725 F.2d 521, 524 (9th Cir. 1984); *Atari, Inc. v. North Am. Philips Consumer Elec. Corp.*, 672 F.2d 607, 615 (7th Cir. 1982); *Rubin v. Boston Magazine Co.*, 645 F.2d 80, 82 (1st Cir. 1981); *Reyher v. Children's Television Workshop*, 533 F.2d 87, 90 (2d Cir. 1976).

Mazer v. Stein, 347 U.S. at 217, does seem to give *Baker v. Selden* as an illustration of the principle that copyright protection is available only for the expression of an idea, not the idea itself. Professor Nimmer relied on this aspect of *Mazer* to argue that *Baker v. Selden* should be understood as limited to the idea/expression distinction. See 1 NIMMER ON COPYRIGHT, *supra* note 81, § 2.18 [D]. Professor Reichman has pointed out that the idea-expression distinction predated *Baker v. Selden*, and if that was all the case represented, it would have been superfluous at the time it was rendered. Reichman has observed that the important historical role of *Baker v. Selden* "was to override the exclusive reproduction rights as applied to utilitarian works . . . [when] the standard defenses (including idea-expression) appeared insufficient to guarantee a third party's right to use functional features embodied in the work . . ." Reichman, *supra* note 2, at 693-95, n.288.

That *Mazer* did not mean by this reference to *Baker v. Selden* to effect a radical change in the meaning of that case is demonstrated by the fact that, in the same paragraph as its discussion of *Baker v. Selden*, *Mazer* cited approvingly two other cases in which defendants had copied the functional designs depicted in plaintiffs' copyrighted drawings by constructing the functional work depicted in the drawing. The courts in these two cases applied the principles of *Baker v. Selden* in ruling that the copyright had not been infringed. See *Muller*, 43 F. Supp. 298 (design for controlling the flow of automobile traffic approaching a bridge not protectable by copyright in drawings); *Fulmer v. United States*, 103 F. Supp. 1021 (Ct. Cl. 1952) (copyright in drawings depicting design for parachutes not infringed by manufacture of parachutes embodying the design), cited in *Mazer* at 347 U.S. at 217 n.39. These cases indicate that by failing to seek patent protection for the functional designs depicted in the drawings, plaintiffs had lost the right to exercise control over the functional implementation of the design by others. *Muller*, 43 F. Supp. at 299-300; *Fulmer*, 103 F. Supp. at 1022. For further discussion of how *Baker v. Selden* has been understood over time, see Reichman, *supra* note 2, at 693-95, n.288.

infringe Selden's copyright when Baker wrote his own treatise, in his own words, describing the special double-entry method of bookkeeping."⁸³ This statement implies that the court thought that Selden sued Baker for copyright infringement because of similarities in the explanatory material in Baker's and Selden's books. This was not so. Selden sued Baker for copyright infringement because Baker's book contained sample ledger sheets that were substantially similar in arrangement to those found in Selden's book.⁸⁴ *Baker v. Selden* was, in other words, a "nonliteral similarity" or "structure, sequence, and organization" ("SSO") case.⁸⁵ To put it in a slightly different way, Baker's book offered potential users of Selden's accounting system a substantially similar "user interface" to that of Selden's.

2. SELDEN'S AND BAKER'S LEGAL CONTENTIONS

A review of the parties' arguments in *Baker v. Selden* reveals the parallels between the legal contentions in that case and those in *Lotus v. Paperback*. Selden, who had won in the lower courts, was, in effect, arguing that there was original expression in the selection, ordering and arrangement of the columns and headings of the ledger sheets contained in his copyrighted book.⁸⁶ That such elements could be protected by

83. *Paperback*, 740 F. Supp. at 54. It is a small point, but neither Baker nor Selden had written a "treatise" on this accounting method. The Court described Selden's book as consisting of "an introductory essay explaining the system of bookkeeping referred to, to which are annexed certain forms or blanks, consisting of ruled lines and headings, illustrating the system and showing how it is to be used and carried out in practice." *Baker*, 101 U.S. at 100.

Although the *Paperback* opinion characterized *Baker v. Selden* as a "seminal case," the same court, after having been informed in briefs submitted by Borland and eleven copyright law professors of its misunderstanding of the holding in that case, is now of the view that *Baker v. Selden* has no application to computer program cases because the Supreme Court decided that case before computer programs were invented. Borland Order, *supra* note 16, at 32-33. This ignores the fact that Congress codified substantial portions of *Baker v. Selden* in § 102(b) and that CONTU also affirmed the applicability of *Baker v. Selden* and its progeny to computer programs.

84. That Baker's ledger sheets were substantially similar to Selden's can be discerned from the Supreme Court's observation that if Selden's copyright extended to the accounting system, the Court would agree that Baker's book infringed the copyright. *Id.* at 100. The court stated that Baker's work "use[d] a similar plan so far as the results are concerned; but makes a different arrangement of the columns, and uses different headings." *Id.*

85. Neither in *Paperback* nor in the *Whelan* decision was there any recognition that *Baker v. Selden* was a nonliteral similarity or "SSO" case in which the "SSO" was ruled outside the scope of copyright.

86. Selden's arguments are summarized in 25 L. Ed. at 842, as well as described by the Court in the body of the opinion. The Court understood Selden to be contending "that the ruled lines and headings, given to illustrate the system, are a part of the book and, as such, are secured by the copyright; and that no one can make or use similar ruled lines and headings . . . made and arranged on substantially the same system, without violating the

copyright law was evident, Selden asserted, from cases involving maps, charts, and diagrams,⁸⁷ among others.⁸⁸ Selden insisted that Baker's arguments about "uncopyrightable subject matter" simply missed the point.

In support of his "uncopyrightable subject matter" defense, Baker pointed out that Selden had gone to the Patent Office to get a patent for his bookkeeping system.⁸⁹ Baker argued that this demonstrated that the *system* was not the proper subject matter of copyright, but should be protected, if at all, by a patent.⁹⁰ It was a contribution to the useful arts, Baker argued, not to literature. Baker insisted that the ledger sheets conveyed no thought, provided no information, and expressed no idea over and above the system they embodied.⁹¹

Baker relied on some cases denying copyright protection to forms,⁹² as well as on the Court's then very recent decision in *The Trademark Cases*.⁹³ In that set of cases, the Court had very clearly distinguished between those things that were "writings" of "authors," and hence the within the subject matter of copyright, and those that were "inventions" in the "useful arts," which were the province of the patent system.⁹⁴

copyright." *Baker*, 101 U.S. at 101. Section V *infra* will discuss some subsequent copyright "blank forms" cases.

87. See synopsis of Selden's argument preceding the Court's opinion, 25 L. Ed. at 842.

88. The Court's opinion indicates that Selden relied heavily on *Drury v. Ewing*, 7 F. Cas. 113 (C.C.S.D. Oh. 1862), a case in which copyright was claimed in a chart of patterns for clothing. *Baker*, 101 U.S. at 107. The Court questioned the *Drury* decision, but concluded that in any event it was not a controlling case. *Id.*

89. That Baker relied on this point is demonstrated in the summary of his argument in 25 L. Ed. at 841. The Court's opinion indicated that no patent issued on Selden's system. 101 U.S. at 104. The opinion does not reveal whether the patent was denied or merely withdrawn.

90. More recent cases would suggest that Selden's system was not patentable subject matter because it was a business method, see, e.g., *Ex Parte Murray*, 9 U.S.P.Q.2d (BNA) 1819 (P.T.O. Bd. App. 1988) although one case suggests the method might be patentable if carried out by computer. See *Paine, Webber, Jackson & Curtis, Inc. v. Merrill Lynch, Pierce, Fenner, & Smith, Inc.*, 564 F. Supp. 1358 (D. Del. 1983).

91. See synopsis of Baker's argument preceding the Court's opinion, 25 L. Ed. at 841.

92. *Id.* at 842; see also 101 U.S. at 106-107 for the court's discussion of a case that had denied copyright protection to a cricket scoring sheet.

93. 100 U.S. 82 (1879). In that opinion, the Court ruled that the Constitutional clause empowering Congress to legislate to give exclusive rights to "authors" for their "writings" and to "inventors" for their "discoveries" in the "useful arts" did not give Congress power to pass a uniform national trademark statute. Baker's reliance on these cases is referred to at 25 L. Ed. at 842.

94. 100 U.S. at 94. Interestingly enough, the Court's opinion in *Baker v. Selden* contains no direct reference to *The Trademark Cases*, although the Court's concern about not allowing copyright law to be used to protect things in the patent domain is evident from the *Baker v. Selden* opinion, 101 U.S. at 102-03. It is worth noting that the Supreme Court's recent decision *Feist Publications, Inc. v. Rural Tel. Serv. Co.*, 111 S. Ct. 1282 (1991) relied heavily on and quoted approvingly from both *The Trademark Cases* and *Baker v. Selden* concerning the meaning of original expression and authorship in copyright law, as well as what copyright law can properly protect in a work.

Baker argued that having been unable to get a patent, Selden should not be able to use copyright to get indirectly the kind of protection that the Patent Office would not grant him directly. Even embodied in a copyrighted book, the system still remained outside the subject matter of copyright. As long as he wrote his own explanatory material about the system, Baker insisted that no copyright liability could arise from having substantially similar ledger sheets in his book, for he needed to be able to reproduce similar ledger sheets in order to illustrate the bookkeeping system in his own book.

3. THE SUPREME COURT DECISION IN BAKER V. SELDEN

The Supreme Court overturned the ruling in Selden's favor, and ordered the complaint against Baker to be dismissed.⁹⁵ The Court agreed with Selden that he held a lawful copyright in the book he wrote explaining his accounting system.⁹⁶ But the Court agreed with Baker that the copyright in the book no more gave its owner exclusive rights in the accounting system than the copyright in a book on the composition and uses of medicines would give its author exclusive rights over manufacture and application of the medicines.⁹⁷ To get exclusive rights for innovations of these sorts, an innovator would have to go to the Patent Office. "To give to the author of a 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 of copyright."⁹⁸ From copyright, the Court insisted, an innovator could get only exclusive rights to print and distribute his book.⁹⁹ The Court pointed out that the novelty of the art being described in the book had no relevance in determining whether copyright protection was available for it.¹⁰⁰

95. *Baker v. Selden*, 101 U.S. at 107.

96. *Id.* at 101-102.

97. *Id.* at 102. The Court noted that a book on an accounting system might "contain[] detailed explanations of the art, [and] it may be a very valuable acquisition to the practical knowledge of the community." *Id.* But this did not make the detailed knowledge concerning the art a protectable element of the copyrighted book. *Id.* The last substantive line of the recent *Feist* decision quotes *Baker v. Selden* on a similar point: "'great praise may be due to the plaintiffs for their industry and enterprise in publishing this paper, yet the law does not contemplate their being rewarded in this way,'" *Feist*, 111 S. Ct. at 1297 (quoting *Baker v. Selden*, 101 U.S. at 105).

98. *Baker*, 102 U.S. at 102.

99. *Id.* at 102-103.

100. *Id.* at 102. This aspect of the Court's opinion is worth noting because *Paperback* emphasizes the "unobviousness" of the Lotus interface as if this had significant bearing on its protectability by copyright. The court even states that "obviousness" was one of five concepts to be considered in analyzing a case such as *Paperback*, 740 F. Supp. at 58. (The others were the idea-expression distinction, functionality, originality, and merger. *Id.* at 58-59). "Obviousness" is an important concept in patent law, but not in copyright law. The court's frequent references to "obviousness" in *Paperback* indicate that the court may

The Court explained further:

The copyright of a work on mathematical science cannot give to the author an exclusive right to the methods of operation which he propounds, or to the diagrams which he employs to explain them, *so as to prevent an engineer from using them whenever occasion requires*. The very object of publishing a book on science or useful arts is to communicate to the world the useful knowledge which it contains. But this object would be frustrated if the knowledge could not be used without incurring the guilt of piracy of the book. And *where the art it teaches cannot be used without employing the methods and diagrams used to illustrate the book, or such as are similar to them, such methods and diagrams are to be considered as necessary incidents to the art, and given therewith to the public . . .*¹⁰¹

The Court noted that the plausibility of Selden's claim arose from the unusual nature of the useful art the case involved: "In describing the art, the illustrations and diagrams employed happen to correspond more closely than usual with the actual work performed by the operator who uses the art."¹⁰² While in most instances useful arts were embodied in wood, metal, or stone, the peculiar art this case involved was embodied in a writing. But, the Court announced, "the principle is the same in all. The description of the art in a book, though entitled to the benefit of copyright, lays no foundation for an exclusive claim to the art itself."¹⁰³

have been confused on this point. See, e.g., *Paperback*, 740 F. Supp. at 58 (asserting that it would be wrong to deny copyright protection to the "most original and least obvious" aspects of a work). *Id.* at 65 (asserting that although the core idea of a spreadsheet program was "functional and obvious," not "every possible method" of designing a spreadsheet program was obvious, and originality in a copyright sense involved "pressing beyond the obvious"). *Id.* at 66 (use of "+" to represent addition was "obvious if not essential" (emphasis in the original)). *Id.* at 68 (referring to the command structure of 1-2-3 as "original and nonobvious"). See also *id.* at 79 (concerning the need to protect "strikingly innovative" aspects of programs by copyright). These statements suggest that the court in *Paperback* had lost sight of a fundamental point of the *Baker v. Selden* opinion that the novelty (or nonobviousness) of Selden's accounting system had no bearing on whether it was protectable by copyright law.

101. *Baker*, 101 U.S. at 104 (emphasis added). Much of this passage was quoted with approval in *Feist*, 111 S.Ct. at 1290.

Cases such as *Whelan*, 797 F.2d 1222, 1236 assert that they are consistent with *Baker v. Selden* because if a particular detail of a program is "necessary" to achievement of a program's general purpose or function, idea/expression merger will be found. Indeed, the Third Circuit in *Whelan* purported to derive its test for infringement from *Baker v. Selden*. *Id.* at 1235-36. *Whelan*, however, fundamentally misconstrues the larger meaning of *Baker v. Selden*, especially its importance in substantially limiting the scope of copyright protection available for functional writings, such as rulebooks, engineering drawings, and nonfanciful computer programs such as spreadsheet programs. See cases cited in note 81.

102. *Baker*, 101 U.S. at 104.

103. *Id.* at 105 (emphasis added). The Court indicated that the "useful art" was not protectable regardless of whether it was explained in a book or illustrated by diagrams: "Those illustrations are the mere language employed by the author to convey his ideas more clearly. Had he used words of description instead of diagrams, which merely stand in the place of words, there could not be the slightest doubt that others, applying the art to

As all this pertained to Selden's claims, the Court reasoned that since Selden's copyright did not give him any exclusive rights in the accounting system, Baker was free to put similar ledger sheets illustrating use of the system in his book as well. To hold otherwise would indirectly give Selden exclusive rights in his system which the Court regarded as improper to recognize directly.¹⁰⁴ In effect, the Court decided that the selection, ordering, and arrangement of these columns and items in the ledger sheets were constituent parts of Selden's accounting system, and hence, not a part of the book's protectable expression.¹⁰⁵

Given that Baker may have made his sample ledger sheets somewhat different from Selden's out of fear that making them identical would almost inevitably bring on a lawsuit, it is worth inquiring whether there is anything in the Court's opinion suggesting that the Court would have ruled differently if Baker had copied Selden's ledger sheets exactly, column for column, heading for heading. The question has a clear bearing on the implications of *Baker v. Selden* for the *Lotus v. Paperback* dispute which involved at least some exact copying. The last substantive statement of *Baker v. Selden* directly answers this question: "The conclusion to which we have come, is that blank account-books are not the subject of copyright; and that the mere copyright of Selden's book did not confer upon him the exclusive right to make and use account-books, ruled and arranged as designated by him and described and illustrated in said book."¹⁰⁶ The Court's decision, then, did not rest on a finding that Baker's ledger sheets were different enough from Selden's that they did not infringe. The Court ruled that Selden's ledger sheets were simply not protectable by copyright law, just as Baker asserted.

Baker v. Selden is one of many U.S. copyright cases in which an arrangement of words has been found to be a constituent element of a "system," and unprotectable by copyright law. It is from such cases that the meaning of § 102(b)'s exclusion of "systems" and "methods of operation" is to be found. Yet because of its use of the erroneous *Whelan* test, the court in *Paperback* neglected to make an inquiry about whether any of the aspects of the Lotus interface copied by Paperback were constituent elements of a "system" that under *Baker v. Selden* and § 102(b), should have put those elements outside the scope of copyright protection available to the Lotus program. A number of statements in *Paperback*,

practical use, might lawfully draw the lines and diagrams which were in the author's mind, and which he thus described by words in his book." *Id.* at 104 (emphasis added).

104. *Id.* at 104-105.

105. Copyright Professors Amicus Brief, *supra* note 10, at 6-7.

106. 101 U.S. at 107 (emphasis added). See also Copyright Professors Amicus Brief, *supra* note 10, at 7 (indicating that in a proper case, even exact copying of system elements can be noninfringing conduct); BENJAMIN KAPLAN, AN UNHURRIED VIEW OF COPYRIGHT LAW 64 (1967). The system or method will typically constrain the range of possible variations by which the system or method can be illustrated.

particularly those dealing with Lotus' macro facility, suggest that the command hierarchy of the Lotus interface was a constituent element of a system.

B. The Macro Facility of Lotus 1-2-3 As a System

The original Lotus complaint asserted that the command language and syntax of the 1-2-3 interface were protectable by copyright law.¹⁰⁷ Although the syntax claim seems not to have been directly pressed at trial,¹⁰⁸ it is evident from *Paperback* that Lotus was still asserting that the command language of 1-2-3 was protectable by copyright law.¹⁰⁹ Paperback attempted to counter this claim by arguing that languages were unprotectable by copyright law, and offering evidence to show that the Lotus macro facility involved a language.¹¹⁰

How little regard the court had for Paperback's language defense is illustrated by the fact that it is only discussed near the end of the opinion in a section entitled "A Postscript on the Nature of Decisionmaking in This Case" in a subsection entitled "Strained Analogies and Word Games."¹¹¹ Curiously, the court acted as though Paperback, rather than Lotus, had introduced the language issue into the case and criticized the

107. See Complaint of Lotus Development Corp., *supra* note 20.

108. Syntax is, by definition, a set of abstract rules which must be followed for statements in a language to be meaningful. Since it is well-established in copyright law that "rules" are not protectable by copyright law, *see, e.g.,* *Morrissey v. Proctor & Gamble*, 379 F.2d 675 (1st Cir. 1967), a direct claim for the Lotus syntax as a protectable element of the program would be on shaky grounds. Because of its importance in the ordering of commands in the Lotus 1-2-3 command structure, syntax remained indirectly in the case. For the court's recognition of this, *see infra* note 114 and accompanying text. That syntax affects semantics can be easily demonstrated with an example in the English language. "The dog bit the man" has a different meaning than "the man bit the dog," notwithstanding the fact that both sentences have the same words. The meanings differ because of English syntactic rules about the placement of subjects of sentences vis-à-vis verbs and other parts of the predicate.

109. *Paperback*, 740 F. Supp. at 63.

110. Professor Harry W. Lewis of the Harvard University Computer Science Department submitted an affidavit directed to this issue to support Paperback's defense. That the "language" defense was one of Paperback's principal defenses is evidenced by the extensive treatment given to the issue in its pretrial brief. *See Paperback Pretrial Brief, supra* note 10, at 30-43. In the course of the litigation, Lotus seems to have downplayed the language claims, focusing more attention on the "command structure" issue for which it may have been easier to find helpful copyright precedents and on its argument that the user interface as a whole was protectable by copyright.

111. *Paperback*, 740 F. Supp. at 71-73. Earlier sections of the *Paperback* opinion mention machine, assembly, and higher order computer languages. *Id.* at 44-45. There is no mention of these languages in the subsection discussing Paperback's language defense. It is possible that one reason the court was so unsympathetic to Paperback's language defense was that the Lotus commands may have seemed too different from these other kinds of languages to be a comparable phenomenon. If, however, the Lotus macro facility, as a matter of computer science, satisfied the formal definition of "language," the court should have taken the defense more seriously.

defense lawyers for even raising it.¹¹² The court failed to perceive that Paperback's language defense was, in fact, a § 102(b) "system" defense, analysis of which should have been integrated into the body of the court's infringement discussion rather than being tacked onto the end of the opinion in a "postscript" section. In similar fashion, the court failed to perceive the connection between the language defense and Paperback's assertion that copyright protection was unavailable to the Lotus command structure.

Paperback contains two paragraphs describing the macro facility of Lotus 1-2-3 in a subsection describing the Lotus interface in a part of the opinion that precedes the section analyzing the "copyrightability" of various aspects of the Lotus interface. These paragraphs reveal that the court understood that users of the Lotus program could use the Lotus commands to construct macros in order to adapt the program to better serve their needs:

Rather than going step-by-step through the same sequence of commands each time there is a need to perform a particular function, the user may store a sequence of command terms as a "macroinstruction," commonly called a "macro," and then, with one command stroke that invokes the macro, cause the programmed computer to execute the entire sequence of commands.¹¹³

The Lotus commands could, in other words, be used as building blocks for construction of these macros, which is why they could accurately be called elements of a language. After noting that macros could be built not only by combining Lotus command terms into sequences, but also by combining function keys and other aspects of the interface into the sequences, the court went on to say:

Because macros may contain many menu choices, the *exact hierarchy*—or structure, sequence, and organization—of the menu system is a *fundamental part of the functionality of the macros*.¹¹⁴

There are several interesting things about this statement. One is that it directly refers to a "menu system" of Lotus 1-2-3, although not explaining what the court meant by this term or probing its possible

112. The court was critical of Paperback's attorneys for failing to make what it regarded as a coherent statement of the defense, and ultimately made its own statement of the argument underlying this defense. *Paperback*, 740 F. Supp. at 72. The court offered several reasons for regarding the argument as flawed. *Id.* at 72-73. The court thought that the argument depended on "arbitrary definitions of words, adopted for undisclosed reasons," and concluded that this defense was "totally without merit." *Id.* The court chastised the defense lawyers for engaging in a "word game" by raising this defense. *Id.*

113. *Id.* at 64.

114. *Id.* at 65 (emphasis added). See also *id.* at 78 (referring to Lotus' "command facility" and "macrocommand facility," and the availability of "translation devices" to allow 1-2-3 macros to be converted to other programs). The statement quoted in the text was made in a section of the opinion describing the Lotus interface, *id.* at 63-65, and before any discussion of what was protectable expression in the interface.

copyright consequences either in that section or elsewhere. A second is that the statement indicates that the command hierarchy of Lotus 1-2-3 is a "fundamental part" of the "functionality" of the macros, again without an inquiry about the possible copyright consequences of this aspect of its statement. Thirdly, the statement seems to indicate that the command hierarchy is a fundamental part of the Lotus macro system, such that the commands must be in exactly the same hierarchical structure in order for macros constructed in the Lotus language to be executed successfully. And indeed, that is what both Paperback and Borland have asserted in defense of the Lotus lawsuits brought against them.

Because copyright law does not provide protection to "systems" embodied in copyrighted programs, nor to constituent elements of systems, one would have expected the court in *Paperback* to reflect on its own characterization of the macro facility, and to perceive the linkage between it, the language defense, and § 102(b). To the extent that the command hierarchy is a constituent part of the Lotus macro system, it would seem to be outside the bounds of copyright protection under § 102(b). Yet no reference to the role of the command hierarchy in the macro system or to § 102(b) can be found in any of the sections of *Paperback* in which the substantive rulings of the case were made. Neither § 102(b) nor the role of the command hierarchy in the macro system is even mentioned in the part of the opinion discussing the language defense.¹¹⁵

Even apart from the court's own statement about the role of the command hierarchy in the macro facility, Paperback's language defense should have been taken more seriously. A language is, by definition, a formal system consisting of three elements: a vocabulary, a syntax, and semantics.¹¹⁶ There was evidence in the record demonstrating why, as a matter of computer science, the Lotus commands could properly be understood to be elements of a language.¹¹⁷ *Paperback* gives no reasons for its rejection of this evidence.

The court instead gives its own tortured statement of what it found to be an incomprehensible defense¹¹⁸ and challenged Paperback to find precedents to show that languages were uncopyrightable.¹¹⁹ From Paperback's failure to find any such precedents, the court seems to have

115. *Paperback*, 740 F. Supp. at 71-73.

116. See INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS ("IEEE"), STANDARD DICTIONARY OF ELECTRICAL AND ELECTRONIC TERMS 478 (3d ed. 1984). See generally Richard H. Stern, *On Copyright in Computer Languages*, 17 RUTGERS COMPUTER & TECH. L.J. 321 (1991).

117. See *supra* note 110. Moreover, Lotus itself had referred to 1-2-3's menu and macrocommand language as protectable elements of the Lotus interface. See *supra* note 20 and accompanying text.

118. *Paperback*, 740 F. Supp. at 71-72.

119. *Id.* at 72.

inferred that the defense had no merit. The court might instead have insisted that Lotus affirmatively prove that languages are copyrightable, or might at least have recognized that Lotus was raising a novel question for it to decide.

Although Paperback may not have found them, there are, in fact, some copyright precedents to support the view that languages are, in fact, not protectable by copyright law even when embodied in the texts of copyrighted works. The "shorthand system" cases, like the *Paperback* case, involve claims to language components used in the practice of the plaintiffs' systems.¹²⁰ The courts in these cases have, with due nods to *Baker v. Selden*, denied plaintiffs the protection they sought from copyright law. None of these cases is discussed in the subsection on the language defense.¹²¹ Nor does the court mention statements by a number of commentators who have opined that computer languages should be regarded as unprotectable by copyright law.¹²² The court's dismissive

120. See, e.g., *Brief English Sys.*, 48 F.2d 555 (finding no copyright infringement where similarities between the plaintiff's and defendant's works were attributable to their being about the same shorthand system (citing *Baker v. Selden*, 101 U.S. 99)); *Griggs*, 49 F. 15 (same). See also *Signo Trading Int'l, Inc. v. Gordon*, 535 F. Supp. 362 (N.D. Cal. 1981) (translation system not protectable by copyright). But see *Nikanov v. Simon & Schuster, Inc.*, 246 F.2d 501 (2d Cir. 1957) (finding infringement of a chart for studying the Russian alphabet).

121. This part of the *Paperback* opinion makes a passing reference to the "coined words" of *Reiss v. National Quotation Bureau, Inc.*, 276 F. 717 (S.D.N.Y. 1921), but does not discuss the case. *Reiss* was an uncharacteristically terse opinion written by then District Judge Learned Hand in which the court upheld the validity of a copyright in a compilation of coined words intended to enable purchasers of the book to encode their communications to other persons via cable. It was, in short, a book consisting of a possible vocabulary without syntax or semantics. *Brief English Sys.*, 48 F.2d 555, decided ten years after *Reiss* and by the Court of Appeals for the Second Circuit, seems a more apposite precedent, given its more linguistic character.

Compilations have long been troublesome for copyright law, and the temptation to protect them on account of the work that was required to prepare them has proved very strong over the years, as demonstrated by *Reiss* and the long line of "sweat of the brow" cases recently rejected as erroneous interpretations of copyright law in *Feist*, 111 S. Ct. 1282, 1292-95. Even so, copyright does have a long history of protecting compilations. See generally Jane C. Ginsburg, *Creation and Commercial Value: Copyright Protection of Works of Information*, 90 COLUM. L. REV. 1865 (1990).

122. Commentators expressed this view before the *Paperback* opinion issued. See, e.g., John P. Sumner, *The Patent/Copyright Interface: Patent Protection for the Structure of Code*, 30 JURIMETRICS J. 107, 112 (1989); Leo J. Raskind, *The Uncertain Case For Special Legislation Protecting Computer Software*, 47 U. PITT. L. REV. 1131, 1174 (1986); Richard H. Stern, *The Bundle of Rights Suited To New Technology*, 47 U. PITT. L. REV. 1229, 1239 n.64 (1986); Steven R. England, Note, *Idea, Process, or Protected Expression?: Determining the Scope of Copyright Protection of the Structure of Computer Programs*, 88 MICH. L. REV. 866 (1990). This Note was cited in *Paperback*, 740 F. Supp. at 53. It asserts that the *Synercom* decision was correct because the input formats for Synercom's structural analysis program were a language: "Thought of in this way, *Synercom* was an easy case, for there can be no more protection for input formats than for the English language itself." *Id.* at 882 n.82. Commentators have expressed the view that computer languages are not protectable by copyright after

treatment of the language defense was improper as a matter of copyright law.

C. The "Compatibility" Defense

A review of the court's discussion of Paperback's "compatibility" defense suggests that it too should either have been integrated with the "language-as-system" defense or given more serious treatment as a "merger" defense.¹²³ Paperback argued that in order to develop a commercially viable spreadsheet program, it was "necessary" for its program to be "compatible" with Lotus because of user dependence on the Lotus macro facility. Copying the Lotus command hierarchy was said to be necessary to achieve this compatibility. Hence, argued Paperback, this was an instance in which "idea/expression merger" should be found, as it had been in *Synercom*.¹²⁴

Paperback pointed out that the macro facility of the Lotus program allows users with complex computational needs to create "libraries" of macros that allow rapid single stroke execution of the sequences necessary to accomplish these tasks. Whole books have been written to advise users on how to construct macros for specific kinds of functions for Lotus 1-2-3.¹²⁵ After investing considerable time and energy in the creation of macros, users will naturally want to continue to be able to use these macros, and perhaps to share them with others with whom they may be working and who may or may not have a Lotus program on their computers.¹²⁶

the *Paperback* decision. See, e.g., *Legal Protection of Screen Displays*, *supra* note 14, at 283; and Lowry, *supra* note 14, at 1293.

123. Discussion of Paperback's compatibility defense can be found in *Paperback*, 740 F. Supp. at 69 and 77-79.

124. *Synercom*, 462 F. Supp. 1003 (input formats for structural analysis program inseparable from "idea" and hence unprotectable by copyright law).

125. See, e.g., DAVID P. EWING, 1-2-3 MACRO LIBRARY (3d ed. 1990).

126. Like the "compatibility" defense, Paperback's "standardization" defense aimed at injecting the interests of users of spreadsheet programs into the court's copyright analysis. As with the "compatibility" defense, the court rejected the "standardization" defense, and indeed, the court treated these two defenses as intertwined. *Paperback*, 740 F. Supp. at 77-79.

It is beyond the scope of this article to deal at length with the "standardization" defense. Other commentators have dealt with standardization defenses in computer program copyright cases at greater length. See, e.g., Karjala, *supra* note 33, at 35; *Copyright Protection for Application Programs*, *supra* note 6, at 1045; *Legal Protection of Screen Displays*, *supra* note 14, at 283; and Lewis, *supra* note 14, at 689. Economists have argued that copyright law should be applied to computer program user interfaces so as to promote standardization. See, e.g., Joseph Farrell, *Standardization and Intellectual Property*, 30 *JURIMETRICS J.* 35, 47-48 (1989); *Copyright Protection for Application Programs*, *supra* note 6, at 1045.

The court in *Paperback* was surely wrong to say that there was no copyright precedent to support recognition of a standardization defense. 740 F. Supp. at 79. A number of computer program copyright cases have regarded elements of user interfaces as

"Compatibility"¹²⁷ with the Lotus program, in essence, means that users with macro libraries can continue to enjoy the fruits of their own labor by allowing them to port over to another spreadsheet program the macros they have constructed using Lotus 1-2-3. As the court's description of the Lotus macro facility reveals, the exact same hierarchy of commands as 1-2-3 must exist in another program for macros built in the 1-2-3 system to be executed there. If, from the users' standpoint, the Lotus macro facility is a system on which users depend so heavily that they won't use a spreadsheet program that doesn't have this facility,

unprotectable for reasons consistent with a standardization defense: some because the elements had become conventional to programs of the sort the case involved, *see, e.g.*, *Telemarketing Resources v. Symantec Corp.*, 12 U.S.P.Q.2d (BNA) 1991, 1995 (N.D. Cal. 1989); some out of concern for users who otherwise would have to be retrained, *see, e.g.*, *Synercom*, 462 F. Supp. 1003, 1013; and some because market factors required commonality in user interfaces, *see, e.g.*, *Plains Cotton*, 807 F.2d 1256. *See also Kepner-Tregoe*, 203 U.S.P.Q. (BNA) at 132 (discussing how market factors can narrow the range of available expressions). In the context of traditional kinds of literary works (novels and plays), the *scenes a faire* doctrine is a comparable defense. *See Paperback*, 740 F. Supp. at 59 (recognizing the *scenes a faire* doctrine).

What may have made Paperback's argument so difficult for the court to accept was that it was an argument that virtually all of the Lotus interface had become a standard. Intellectual property scholars who reached consensus on many user interface issues at the LaST Frontier Conference were unable to reach consensus on what if any weight should be given to standardization concerns in analyzing copyright infringement claims involving user interfaces. *LaST Frontier Report*, *supra* note 6, at 28, 31.

127. "Compatibility" has been a heatedly debated computer program copyright issue both in the United States and abroad. The issue arises both as to internal interfaces of computer programs and as to user interfaces of programs. From the standpoint of the technical community, the compatibility issues raised by both are much the same, even though copyright lawyers tend to treat them as somewhat more distinguishable. *See, e.g.*, *LaST Frontier Report*, *supra* note 6, at 21-22, 26-33.

Some think that copyright law should be interpreted so that "interfaces" would be "ideas" and only the code implementing them should be considered copyrightable "expression." Under this view, if one firm copies the interface of another firm (at least nonfanciful aspects of it) in order to make a compatible product, no copyright infringement should be found. *See, e.g.*, *Karjala*, *supra* note 33, at 33; Michael A. Jacobs, *Copyright and Compatibility*, 30 JURIMETRICS J. 91 (1989). Others regard interfaces as valuable "nonliteral" elements of copyrighted programs, the copying of which should be treated as an appropriation of "expression." There is, in this view, no "right" under copyright law to make a "compatible" product. *See, e.g.*, William T. Lake, John H. Harwood, & Thomas Olson, *Tampering With Fundamentals: A Critique of Proposed Changes in EC Software Protection*, 6 COMPUTER L. 1 (Dec. 1989). *See also* NRC REPORT, *supra* note 17, at 67-69 (discussion of opposing views on copyright protection for interfaces).

The recently adopted European Directive on copyright protection for computer programs recognizes that interfaces may be unprotectable "ideas" of computer programs, and allows such interfaces as are necessary to achieve interoperability to be copied without copyright liability. *See* Council Of The European Communities Directive On The Protection Of Computer Programs, reproduced at 42 Pat., Trademark & Copyright J. (BNA) 109 (May 23, 1991). The caselaw in the United States is somewhat more mixed on this question. *See* cases discussed in the articles cited earlier in this note. The court in *Paperback* must have been aware of the larger debate over the "compatibility" issue, yet the opinion does not refer to this debate or to the caselaw that addresses the issue.

Paperback's compatibility defense would seem to present serious copyright issues.

The court's first response to Paperback's compatibility defense was to point out that some spreadsheet developers had attained commercial viability without copying the command structure of the Lotus interface, and hence Paperback had not proven the necessity of copying.¹²⁸ Yet the opinion goes on to suggest that if Paperback had copied the Lotus commands in a more indirect or inefficient manner—as by creating a help facility to inform users what VP-Planner command was equivalent to each Lotus command, or having a macrocommand conversion facility, such as Microsoft's Excel program had—this might have been an acceptable way to achieve compatibility.¹²⁹

By identifying only two ways to achieve compatibility—both of which would have involved some copying from the Lotus program, albeit having a different mode of presentation—the court seemed to forget that it had previously recognized that the existence of only a small number of ways to do something would satisfy the “necessity” test for “idea/expression merger” purposes.¹³⁰ The court had, for example, found “merger” as to some features of the Lotus interface, such as the use of the “/” to invoke the command menus and the rotated “L” grid, despite the fact that there were programs on the market that used different means than these.¹³¹ If there were only two ways, apart from that chosen by Paperback, to achieve compatibility with the Lotus program and both alternatives involved significant copying of the Lotus commands, perhaps the command structure was a good candidate for “idea/expression merger.”

An additional factor that might have supported the “merger” argument was the inefficiency from the user's standpoint in having to call up a help screen each time one wished to identify what VP-Planner commands were equivalent to what Lotus commands. This would seem to require difference for difference's sake, rather than because of some expressive quality in the command terms and their arrangement.¹³² From a competitor's standpoint, having to develop a macro conversion facility to achieve compatibility also might be more inefficient than just presenting the Lotus commands as Paperback did. The Supreme Court recently ruled that competitors were not required to engage in inefficient

128. *Paperback*, 740 F. Supp. at 69.

129. *Id.*

130. *Id.* at 61.

131. *Id.* at 66.

132. See *Kepner-Tregoe*, 203 U.S.P.Q. (BNA) 124 (difference for difference's sake not necessary).

copying when traditional principles of federal intellectual property law would regard the copied aspect of the work as unprotectable.¹³³

How unwilling the court was to consider "merger" when a commercially valuable aspect of the Lotus interface was at stake is best revealed by this statement:

[E]ven if VP-Planner otherwise would have been a commercial failure, and even if no other technological ways of achieving macro and menu compatibility existed, the desire to achieve "compatibility" or "standardization" cannot override the rights of authors to a limited monopoly in the expression embodied in their intellectual "work."¹³⁴

This would seem to reflect a willingness to let Lotus enjoy a complete monopoly in the electronic spreadsheet market, a result hardly in keeping with CONTU's assurances that copyright protection for programs would not lead to monopolization of the market for software products or deter entry of competitors.¹³⁵

Paperback's treatment of the "compatibility" defense, like its treatment of the "language" defense, was unsatisfactory as a matter of copyright law. The two defenses were strands of a § 102(b) defense pertaining to the macro facility of 1-2-3, even though the court failed to perceive this. While the macro facility of 1-2-3 may be a highly useful aspect of the Lotus program—one that might well be patentable under today's standards for computer program-related inventions¹³⁶—if it is, as the court's description seems to indicate, a "system," it should be deemed to be outside the scope of copyright protection. To the extent the command structure is an essential component of the macro facility, it too may be outside the scope of copyright. Only if the court determined that there was "expression" in the Lotus interface, over and above the role of significant components of it in the macro system, could the court, consistent with traditional principles of copyright law, properly find infringement in the *Paperback* case.

133. *Bonito Boats, Inc. v. Thunder Craft Boats, Inc.*, 489 U.S. 141 (1989).

134. *Paperback*, 740 F. Supp. at 69.

135. See CONTU REPORT, *supra* note 5, at 23-24.

136. Although the caselaw on the patentability of computer program-related inventions is somewhat unclear, the U.S. Patent and Trademark Office has been taking a broad view of the patentability of such inventions. The Lotus macro facility would seem to qualify for patent protection under recently issued guidelines. See P.T.O. Report on Patentable Subject Matter: Mathematical Algorithms and Computer Programs, 38 Pat., Trademark, & Copyright J. (BNA) 563, 569-71 (1989). See also Donald S. Chisum, *The Patentability of Algorithms*, 47 U. PITT. L. REV. 959 (1986); cf. Pamela Samuelson, Benson Revisited: *The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions*, 39 EMORY L.J. 1025 (1990).

V. WHAT WAS "EXPRESSIVE" ABOUT THE LOTUS INTERFACE?

Determining what in a copyrighted work is "idea" and what is "expression" can be among the most difficult conceptual tasks faced by federal judges. For many kinds of works, such as novels and dramatic plays, there are at least numerous precedents through which a judge can search to find comparable situations to serve as a basis for making a judgment on the matter before him or her. In some cases, however, there is very little apart from general principles of copyright law to give a court guidance about how a particular case should be resolved.

Judge Keeton perceived *Paperback* to be such a case. The court found nothing in the statute,¹³⁷ the legislative history or the CONTU Report that gave more than indirect guidance on the issues.¹³⁸ The court seemed also to think that there were no prior cases that presented anything more than general similarities to the idea/expression problem the court found in *Paperback*.¹³⁹ The court regarded itself as left only with general principles

137. Notwithstanding the court's statements early in *Paperback* about the starting place of any analysis being the statutory language itself, the court only mentions the statutory definition of "computer program" in a background section of the opinion, 740 F. Supp. at 50, and pays very little attention to § 102(b). The statutory definition of computer program more clearly supports Paperback's position in the litigation than Lotus', as does the text of § 102(b) and the caselaw underlying it.

138. *Id.* (CONTU did not directly address the issue presented in *Paperback*). The opinion indicates that the court was aware of the differences of opinion among CONTU Commissioners and staff about CONTU's views as to whether nonliteral elements of programs would be protected by copyright law. *Id.* The opinion does not contain any reference to the legislative history of the 1980 software amendments. The court does refer in the legal background section to some statements from the legislative history of the 1976 Act concerning § 102(b), *id.* at 49, but thereafter pays very little attention to them, even though the text of 102(b) and the legislative history concerning it were directly relevant to the controversy.

There is only one place in *Paperback* in which the court makes clear use of the legislative history of the 1976 Act. In rejecting Paperback's standardization defense, the court referred to statements made by a witness at one of the legislative hearings preceding enactment of the 1976 Act. This witness had warned that copyright protection for computer programs might have disastrous consequences for standardization in that field. *Id.* at 76. Because Congress decided to protect computer programs despite such warnings, the court regarded Paperback's standardization argument to be contrary to Congressional intent. This statement, however, was made at the same hearing at which another witness, Professor Miller, warned of dire consequences from protecting computer programs by copyright law unless Congress adopted what became § 102(b) of the 1976 Act. *See supra* note 4. Given that Congress adopted § 102(b) in response to the concerns stated at this hearing, it may be that Congress thought § 102(b) could be used to address standardization concerns as well.

139. Although the *Paperback* opinion contains some sections in which prior computer program copyright cases are either briefly summarized or quoted parenthetically, *see, e.g., Paperback*, 740 F. Supp. at 55, there is remarkably little analysis of prior cases in the substantive sections of the opinion analyzing the "copyrightability" of the Lotus interface and Paperback's copying of what the court found to be copyrightable elements of the Lotus interface. Occasional references are made to other cases, but only at a very general level. *See, e.g., id.* at 65 (brief reference to *Nichols*); *id.* at 66 (citation to *Morrissey*); *id.* at 67

of copyright law and overall Congressional objectives concerning the protection of computer programs with which to resolve the *Lotus v. Paperback* dispute.¹⁴⁰

This section will show that there were, in fact, a number of copyright cases and principles to which the court could—and should—have looked for guidance in resolving the *Lotus v. Paperback* dispute. In addition, this section will point out how little explanation *Paperback* offers concerning the court's rationale for finding the Lotus interface, or certain components of it such as the Lotus command structure, to be "expressive" in a copyright sense.

Notwithstanding *Paperback's* lengthy abstract discussions of the idea/expression distinction, and notwithstanding the court's willingness to explain why it regarded certain features of the Lotus interface, such as the use of "+" to represent addition, as unprotectable features of the interface,¹⁴¹ the court does not explain its theory about the expressiveness of Lotus command terms such as "worksheet," "range," and the like, or about the expressiveness to be found in the ordering of "range" after "worksheet" in the menu structure.¹⁴² Indeed, the only comment the court made about why commands were ordered as they were in the Lotus interface casts doubt on the view that the arrangement was for expressive purposes, for the opinion notes that the command terms were "presented in order of predicted frequency of use rather than alphabetically."¹⁴³ This suggests the ordering was for functional purposes.¹⁴⁴

(brief reference to *Whelan*); *id.* at 68 (brief reference to *Midway/Bandai*). The only prior computer program case which is discussed in some depth was *Softklone*, 659 F.2d 449, whose holding that computer program menu screens were separate works from the program, was rejected in a section of the opinion discussing *Paperback's* subject matter jurisdiction defense. *Paperback*, 740 F. Supp. at 80-81. Even the *Whelan* decision whose test for infringement heavily influenced the court's analysis of the *Paperback* case (see *supra* Section III) is not much discussed in the opinion.

140. The court discusses what it considered to be relevant general principles of copyright law at 740 F. Supp. at 51-52, 58-62 and relevant Congressional objectives and policies at 52-53, 73-79.

141. 740 F. Supp. at 66-67.

142. The second line of the Lotus and Paperback command menu displays was known as a "long prompt." *Id.* at 64. As a user moves the cursor along the first line of the menu, thereby highlighting different first line commands, the long prompt line would display different information for each highlighted command. Some long prompts were explanations of highlighted command terms. Other long prompts displayed the next array of command choices that would be available to the user if the user cared to invoke them. *Id.* The explanatory long prompts were more clearly "expressive" in character than were long prompts that displayed further menus of command choices. Yet, *Paperback* does not differentiate between the explanatory long prompts and submenu long prompts in assessing the expressiveness of the Lotus interface. The court does, however, observe that *Paperback's* explanatory long prompts were different from Lotus'. *Id.* at 70.

143. *Paperback*, 740 F. Supp. at 67.

144. See Copyright Professors Amicus Brief, *supra* note 10, at 9-10 (indicating that arranging commands in order of predicted frequency would be an arrangement for functional purposes, not expressive purposes).

Nor was there any inquiry concerning whether there were functional reasons for grouping certain commands together, or putting some in certain levels of the hierarchy; nor about whether there might have been other functional factors constraining the design of the command structure; nor about what elements of the command structure might, in fact, not have been "original" to Lotus.¹⁴⁵ The court relies heavily on the fact that Paperback could have done things differently to support the conclusion that the Lotus command structure was "expressive."¹⁴⁶ That something can be done differently may be relevant to an "expressiveness" inquiry, but on its own, it is insufficient to demonstrate the presence of "original expression," as the "blank forms" line of cases discussed *infra* illustrates. This line of cases is among the copyright precedents and principles that the court in *Paperback* should have examined and discussed in assessing the expressiveness of various aspects of the Lotus interface.

A. Copyright Cases and Principles That *Paperback* Should Have Discussed

1. THE BLANK FORMS CASES

Over the years, there have been many copyright cases in which copyright protection has been asserted for "blank forms," that is, forms designed for the recording of information of particular sorts. In these cases, judges have generally examined the forms quite closely to determine what if any expressiveness they contain. If the forms include explanatory material as an integral part of the form, the courts will generally find sufficient "expression" to support a copyright in the

145. Although the court recognizes that the use of "/" to invoke command menus was actually original to Dan Bricklin, a co-developer of the Visicalc program which was the first successful electronic spreadsheet program, the court disposes of Lotus' claim to the "/" on merger grounds rather than on the ground that it was not original to Lotus. *Paperback*, 740 F. Supp. at 66. Although Lotus later acquired the copyright in Bricklin's Visicalc program, at the time Lotus 1-2-3 was developed, this feature of Visicalc was copied by the developers of Lotus 1-2-3. See Affidavit of Daniel S. Bricklin at 22, *Lotus Dev. Corp. v. Paperback Software Int'l* (CA No. 87-0076-K) (June 28, 1990). The developers of Lotus 1-2-3 copied a number of other elements of the Visicalc user interface as well. *Id.* at 34-36. "In most cases, Lotus 1-2-3 uses the same [command] terms as Visicalc . . ." *Id.* at 34. See also Christopher Barr, *From Visicalc to 1-2-3*, PC MAG. (May 26, 1987) (showing how Lotus built on the command structure of Visicalc).

146. *Paperback*, 740 F. Supp. at 67-68. Although the court recognized that some command terms in the Lotus interface were "obvious or merge[d] with the idea," the court stated that this did "not preclude copyrightability for the command structure as a whole." *Id.* at 67. Without quarreling with this statement, it is still a fair question to ask what was expressive about the command structure.

form.¹⁴⁷ When the only text in the forms consists of a set of terse instructions or words merely identifying the categories of information to be elicited when the form is used, the courts will generally find that there is insufficient "original expression" to support copyright protection for the form.¹⁴⁸

Even though it may require some intellectual effort to decide which categories of information should appear on a form and how best to arrange the categories, and even though there may be more than a few ways to do these things, the courts still require more from a form designer before the work can be considered an "original work of authorship" within the meaning of the copyright statute.¹⁴⁹

Those who design command-driven user interfaces such as Lotus 1-2-3, like those who design blank forms, generally aim to maximize system efficiency and ease of use.¹⁵⁰ They will choose command names indicating the function the term represents.¹⁵¹ When a program will have

147. See, e.g., *Beardsley*, 253 F.2d 702 (pamphlet containing forms with explanatory material held copyrightable).

148. See, e.g., *Bibbero*, 893 F.2d 1104 (medical billing forms ruled uncopyrightable for lack of original expression); *John H. Harland Co. v. Clarke Checks, Inc.*, 711 F.2d 966 (11th Cir. 1983) (denying copyright protection to checkbook format); *Janus Mktg. Communications, Inc. v. Doubleday & Co.*, 569 F. Supp. 76 (S.D.N.Y. 1981) (daily activity charts ruled uncopyrightable). See also *Safeguard Business Sys. v. Reynolds & Reynolds Co.*, 14 U.S.P.Q.2d (BNA) 1829, 1832 (E.D. Pa. 1990) ("The Safeguard day sheets are more innovative and sophisticated than the ledger pages in *Baker v. Selden*, and they do convey a certain amount of information. These forms include certain visual guides to where some numbers should be recorded, but they are not sufficiently informative to be subject to copyright."), *aff'd*, 919 F.2d 136 (3rd Cir. 1990). These cases are among the many that trace the rule against copyright protection for "blank forms" to *Baker v. Selden*. As shown *supra* in Section IV-A, the court in *Paperback* perceived *Baker v. Selden* to be a case only about the unprotectability of abstract ideas.

149. See, e.g., *Bibbero*, 893 F.2d 1104 (despite a considerable amount of printing on a medical form and the fact that the information could be arranged differently, the court found no copyrightable expression). *Bibbero* acknowledged that the cases interpreting the blank forms rule "do not yield a consistent line of reasoning." *Id.* at 1107. It distinguished one of the cases on which the plaintiff relied, and stated its disagreement with the ruling in another. *Id.* The court noted that the Copyright Office had recently restudied the blank forms regulation, 37 C.F.R. § 202.1(c). Despite arguments made by blank forms suppliers about their need for copyright protection, the Office found no persuasive argument for repealing this regulation, and reaffirmed the continuing importance of *Baker v. Selden* as authority for the regulation. *Bibbero*, 893 F.2d at 1107. The blank form suppliers were most likely relying on the "sweat of the brow" cases which tested copyrightability by the hard work involved in compiling facts. The "sweat of the brow" cases were, however, recently spurned by the Supreme Court in *Feist*, 111 S. Ct. 1282. In *Feist*, as in *Bibbero*, the Court took seriously the requirement that there be something "expressive" to support copyright protection for the work.

150. See Bill Curtis, *Engineering Computer "Look and Feel": User Interface Technology and Human Factors Engineering*, 30 JURIMETRICS J. 51, 74 (1989) ("The importance of aesthetics relative to other user interfaces increases with the importance of playfulness and decreases with the importance of productivity.").

151. See, e.g., Gary Perlman, *Natural Artificial Languages: Low Level Processes*, 20 INT'L J. MAN-MACHINE STUD. 373 (1984) (discussing choices of mnemonic names for commands).

more than a few commands, designers will tend to limit the number of command terms that are displayed at one time so as not to overwhelm and confuse users with choices.¹⁵² This necessitates the creation of some kind of hierarchy of commands (with those of a more general character typically being made available higher up in the menu of choices) and the grouping in submenus of commands with related functions.¹⁵³ This requires intellectual effort, but it is a kind of intellectual effort that resembles the designing of forms or of human-machine interfaces that are outside the bounds of copyright protection.¹⁵⁴ Yet *Paperback* takes no note of the similarities between "blank forms" and the command-driven Lotus interface; nor does it discuss reasons why the court might have regarded these precedents as inapposite.

2. ASHTON-TATE V. ROSS

Nor did *Paperback* take notice of a California trial court ruling that the command structure for the user interface of a spreadsheet program was insufficiently expressive to be protectable by copyright law. The Ninth Circuit Court of Appeals has affirmed this decision in *Ashton-Tate Corp. v. Ross*.¹⁵⁵ This ruling is directly contrary to *Paperback*.

The case arose out of an unsuccessful collaboration between Ross and a colleague for development of a spreadsheet program for the Macintosh computer. Ross's job was to develop the "engine" for the program, and his colleague's to design the user interface. In the process of developing the engine, Ross wrote out a list of commands that he thought should be included in the user interface, grouped in certain arrangements to represent the various menus the program might have. After Ross and his colleague had a falling out, the colleague went to work for Ashton-Tate, a software development firm, where he worked on a Macintosh spreadsheet program project.¹⁵⁶ The Ashton-Tate product

152. See, e.g., APPLYING COGNITIVE PSYCHOLOGY TO USER INTERFACE DESIGN 268-69 (Margaret Gardner & Bruce Christie eds., 1987) (guidelines for screen design and organization).

153. *Id.*

154. See *infra* note 173 for a discussion of the kinds of human-machine interfaces that are not protectable by copyright law because of the "useful article" doctrine.

155. 728 F. Supp. 597 (N.D. Cal. 1989), *aff'd* 916 F.2d 516 (9th Cir. 1990).

156. There are a couple of interesting ironies about the *Ashton-Tate v. Ross* case. One is that in 1988 Ashton-Tate charged Fox Software with copyright infringement for copying the command structure of the user interface of its popular data base program known as "dBase." (The complaint in that case was modeled after that which Lotus filed against Paperback.) It is difficult to distinguish the claims made by Ashton-Tate against Fox from the claims Ross was making against Ashton-Tate. Thus, Ashton-Tate found itself, in two different litigations, on both sides of the issue as to whether the command structure of a user interface is protectable by copyright law. A second irony is that Ashton-Tate has recently been acquired by another software development firm, Borland International. Borland is a defendant in a similar lawsuit brought by Lotus Development Corp. By virtue of its ownership of Ashton-Tate, Borland is now in litigation on both sides of the

deriving from this project had a user interface in which, Ross alleged, "[n]ot only are the individual commands identical to those of Full Impact, [but] the order in which they are displayed and the menus in which they are contained are identical to the command set of Full Impact."¹⁵⁷

The District Court ruled that Ross's "list of commands is only an idea that is not protected under federal law."¹⁵⁸ On appeal, Ross argued that in a number of decisions, the ordering and arrangement of user interface commands had been protected by copyright law, and emphasized, as those cases had, the large number of different arrangements that were possible as a basis for asserting the arrangement was expressive.¹⁵⁹

The Ninth Circuit affirmed the District Court ruling on the unprotectability of Ross's command hierarchy with approval of the lower court's reasoning.¹⁶⁰ There was, in the appellate court's view, not even a triable issue of fact about whether there was "expressiveness" in this set of user interface commands. The commands were simply the names of functions that the program was capable of performing grouped in a way to promote efficiency in the utilization of the program. It may require intellectual effort to identify what functions the program should perform,

same issue. Because of concerns about the market power Borland would have in the database market if it owned copyrights in both dBase and its own database product and could exclude competitors like Fox from offering commercially viable substitute programs, the Justice Department conditioned its approval of Borland's acquisition of Ashton-Tate on its seeking a prompt resolution of the two user interface copyright lawsuits in a manner that will not give Borland undue market power.

157. See Borland's Redacted Memorandum of Law Regarding the Phasing of the Case at 51 (CA No. 90-11662-K) (June 7, 1991) [hereinafter Borland Brief] (quoting Ross's Brief in Opposition to Ashton-Tate's Motion for Summary Judgment at 14). Following page 51 of the Borland brief is a photocopy of the handwritten list Ross had developed.

158. *Ashton Tate v. Ross*, 728 F. Supp. at 602. The district court stated:

The document given to Wigginton is only a list of labels for user commands, many of which are common commands that were already available on other software programs. There is nothing innovative or novel about the labels that Ross proposed Wigginton use for the program or the order in which they are listed on the document. The single sheet of paper does not contain any source code. The document clearly falls short of the threshold separating ideas from expressions Ross merely told Wigginton what tasks he believed the interface should allow the user to perform.

Id. After saying that the list of commands was only an idea that was not protected by federal law, the court cited to 17 U.S.C. § 102(b). 728 F. Supp. at 602. See also NRC REPORT, *supra* note 17, at 54 (expressing the doubts of some software developers that a distinction between idea and expression can be made in program user interfaces).

159. Borland Brief, *supra* note 157, at 52. The cases include: *Manufacturers Technologies*, 706 F. Supp. 984 (arrangement of items for user interface for cost estimating software held protectable by copyright because of number of alternative ways to arrange the commands); *Softklone*, 659 F. Supp. 449 (grouping and arrangement of command terms in menu screen for communications software held protectable by copyright because of large number of arrangements possible); *Broderbund*, 648 F. Supp. 1127 (arrangement of items on menu screens of a printing program held protectable by copyright because of the possibility of using different arrangements).

160. *Ashton-Tate v. Ross*, 916 F.2d at 521-22.

and to group the commands to facilitate efficient accomplishment of these tasks, but that does not make the arrangement of commands "expressive" in the way they must be for copyright protection to attach. *Paperback* did not even cite the trial court ruling in the *Ross* case, let alone grapple with its contrary ruling. The Ninth Circuit's affirmance of the ruling gives further weight to the trial court's decision in *Ross*.¹⁶¹

3. "THIN" PROTECTION FOR FUNCTIONAL WRITINGS

Among the other traditional principles of copyright law that the court in *Paperback* did not—but should have—employed in constructing its framework for analysis of the copyright issues in *Paperback* was that which recognizes that the scope of a copyright—that is, the breadth of protection provided to a work by copyright law and how far down the hierarchy of abstractions it is appropriate to draw the line between idea and expression—tends to vary according to the nature of the work under consideration. Over the years, courts have come to perceive that there are differing levels of "expressive" content in different kinds of works.

Fanciful or artistic works are generally regarded as enjoying a broad scope of copyright protection because of their predominantly expressive character.¹⁶² Factual works generally have a narrower scope of protection, for in order to further the constitutional purpose of copyright of promoting the growth and dissemination of knowledge, the facts, theories, and other discoveries such works contain are considered outside the scope of the copyright.¹⁶³ Because of this, there is generally less expressive content to be found in factual works than in artistic or fanciful works. An even narrower scope of copyright protection is available for functional writings, such as rulebooks, forms, manuals for operating power plants, engineering drawings, and the like.¹⁶⁴ In general, only

161. Nor is *Ashton-Tate v. Ross* mentioned in Judge Keeton's Memorandum and Order denying cross-motions for summary judgment in the *Lotus v. Borland* case, although *Borland* brought this decision to the court's attention. See *Borland Order*, *supra* note 16.

162. Novels, dramatic plays, and cartoons are examples of works generally enjoying such a broad scope of protection. See, e.g., *LaST Frontier Report*, *supra* note 6, at 18.

163. Biographies, histories, scientific reports, and fact compilations are among the works that enjoy this "thinner" protection under copyright law. *Id.* See also *Feist*, 111 S. Ct. at 1289 (indicating that copyright protection for fact compilations is "thin").

164. Elsewhere the author has distinguished between the truly functional character of computer programs and the kinds of "functional writings" mentioned in the text. *CONTU Revisited*, *supra* note 5, at 727-49. Manuals for operating power plants explain how the plant should be operated; they do not in themselves operate the plant. The wording used to explain the plant operations is the copyrightable "expression" in the manual. The details of plant operations described in the manual are among the work's "ideas." The only "function" of the manual is to convey information, a kind of function that does not disqualify a work from copyright protection. This is in contrast to a computer program written to control the plant's operations; it actually controls the functioning of the plant. The inherently functional nature of programs is what make them so different from traditional categories of copyrightable works.

exact or near-exact copying of such works will be infringing, for the content of these kinds of works tends to be predominantly functional.¹⁶⁵ Under the principles of *Baker v. Selden* and its progeny, this functional content is outside the scope of copyright protection.¹⁶⁶ Only if and to the extent such works contain some “expressive” content is copyright protection available to their authors.

Although user interfaces of computer programs can be fanciful or artistic in character—videogames being perhaps the clearest example—many are more functional in character. Some are too functional to be protectable by copyright.¹⁶⁷ That this should be so is not surprising in view of the fact that computer programs themselves are properly regarded as functional writings,¹⁶⁸ and the role of user interfaces is to provide users with access to program functionalities.¹⁶⁹

Although *Paperback* repeatedly cites copyright cases involving novels, dramatic plays, and fabric designs (that is, artistic or fanciful work cases),¹⁷⁰ it is clear that the court did not regard the Lotus interface as an artistic or fanciful aspect of the program. Indeed, the court agreed with *Paperback* that the Lotus interface was functional.¹⁷¹ Yet the court seems

165. *LaST Frontier Report*, *supra* note 6, at 18-19.

166. *See supra* note 81.

167. One example of this might be the digital display panels of gas pumps. *See LaST Frontier Report*, *supra* note 6, at 27.

168. *Id.* at 19.

169. *See, e.g.*, Curtis, *supra* note 150, at 51.

170. *Paperback*, 740 F. Supp. at 51-52 (citing eight artistic and fanciful work cases for the proposition that nonliteral elements of such works can be protected by copyright law); *id.* at 54 (concluding that protecting nonliteral elements of copyrighted computer programs was “consistent with the treatment of other kinds of intellectual works—specifically, with the treatment of nonliteral elements of expression in musical, dramatic, and motion picture works, and works of literature”); *id.* at 60 (citing artistic or fanciful work cases in support of use of the “patterns of abstraction” test and other standard principles of copyright law); *id.* at 70 (comparing the similarities between the Lotus and *Paperback* interfaces to those in a case involving fabric designs).

171. *Id.* at 54-58. The court states: “Lotus 1-2-3 is surely useful.” *Id.* at 57. The court compares the usefulness of computer programs to that of dictionaries, directories, and maps to illustrate that just because something is useful doesn’t necessarily mean it isn’t copyrightable. *Id.* at 58. This is an inappropriate comparison because dictionaries, directories, and maps are not considered to be “useful” in a copyright sense because their sole function is to convey information or portray an appearance. *See CONTU Revisited*, *supra* note 5, at 727-49. Computer programs are useful because they can be executed in machines in order to perform functional tasks that might otherwise be done in special purpose hardware.

It is clear that by enacting the provisions admitting computer programs to the copyright realm that Congress changed somewhat the copyright rule against the protection of technological processes, a rule for which *Baker v. Selden* might well be cited. The question is how much change did Congress intend. By enacting § 102(b) and explicitly stating that only the “expression” of the programmer, and not the methods, processes, and the like that might be embodied in programs, were to be protected, it would seem that Congress intended to make only minimal changes to the principles of *Baker v. Selden*.

to have found Paperback's arguments about the proper copyright consequences of this functionality to be both confusing and unacceptable.

B. *Paperback's Unwillingness To Accept the Proper Copyright Consequences of The Functionality of Computer Programs And User Interfaces*

Only one subsection of the *Paperback* opinion discusses the functionality of the Lotus interface and the possible copyright consequences of it.¹⁷² In this subsection, the court indicates that Paperback had made a number of "functionality" defenses. The only one the court addressed was that which likened the Lotus interface to other human-machine interfaces that were unprotectable by copyright law.¹⁷³

172. *Paperback*, 740 F. Supp. at 54-58.

173. Paperback's "useful article" defense might be stated in this manner: the Lotus 1-2-3 interface is a human-machine interface; human-machine interfaces have traditionally been proper subject matter for patent—not copyright—protection; traditional copyright law would reject human-machine interfaces from its domain under the "useful article" doctrine; and because there was, in the Lotus interface, no separable "artistic" or "expressive" aspect that could be protected by Lotus' copyright in the program, the Lotus interface failed to satisfy the conditions permitting copyright protection for some aspects of "useful articles." Every aspect of the Lotus interface is directed to achievement of spreadsheet functionality, just as were the columns and headings of Selden's ledger sheets.

In support of this defense, Paperback relied principally on *Synercom*, 462 F. Supp. 1003. In *Synercom*, the court had analogized the input formats for Synercom's statistical analysis program to the unprotectability of the "H" design for automobile gearshifts. Even though particular drawings of this gearshift design might be copyrighted, that would not make the H pattern itself protectable by copyright law such that its subsequent embodiment in an automobile would infringe the copyright in the drawing. Like the gearshift pattern, the input formats were an interface to a machine. The court in *Synercom* found no expression in these formats that was "separable" from the idea they embodied.

It is somewhat surprising that the court in *Paperback* did not reject the "useful article" defense as inapplicable because the program was registered as a "literary work." Some would say that the "useful article" doctrine only applies to pictorial, sculptural, or graphic works. See *Harper House, Inc. v. Thomas Nelson, Inc.*, 889 F.2d 197 (9th Cir. 1989) (stating that the "useful article" doctrine does not apply to textual elements of copyrighted works, although ruling that some aspects of the Harper House organizers were unprotectable because of their utilitarian character). The court in *Paperback*, however, seems to accept that the doctrine had some relevance in computer program cases—an aspect of its ruling quite consistent with *Baker v. Selden*—because it states its understanding of the doctrine more broadly than construction of the statutory provision might suggest: "those elements of a useful article that can exist independently of the utilitarian aspects of the article are potentially copyrightable because those elements are elements of *expression* that can be distinguished from the utilitarian functions of the article." *Paperback*, 740 F. Supp. at 52 (emphasis in the original). Yet the court does not identify what separable expressive elements exist in the Lotus interface.

The subsection on functionality issues in *Paperback* does not display awareness that the "useful article" rule can be traced to *Baker v. Selden*. The subsection cites to only one "useful article" case, which is cited for the proposition that a copyrighted work does not lose copyright status when subsequently put to functional use. *Id.* at 58 (citing *Brandir Int'l Inc. v. Cascade*, 834 F.2d 1142 (2d Cir. 1987)). It is odd that *Paperback* relied on this case because in it, the Second Circuit ruled that a slightly modified version of the plaintiff's sculpture was unprotectable by copyright because changes had been made to the

In parallel to its treatment of Paperback's language defense, the court criticized Paperback's lawyers for failing to make a comprehensible statement of this defense, and after making several of its own statements of the defense, the court rejected it as meritless and based on a "word game."¹⁷⁴

Curiously, the subsection on Paperback's functionality defenses makes no reference to § 102(b), *Baker v Selden*, its progeny, or any of the cases in which the courts have recognized that the scope of copyright protection for functional writings is quite "thin."¹⁷⁵ Through a misconstruction of the "central proposition" of *Synercom*,¹⁷⁶ the court in *Paperback* converted the functionality defense into a "uncopyrightability

sculpture in order to make it more functional, thus causing the work to become an unprotectable "useful article." See *Brandir Int'l, Inc. v. Cascade Pacific Lumber Co.*, 834 F.2d 1142 (2d Cir. 1987).

174. See *Paperback*, 740 F. Supp. at 56, 71-73. The court made its own statement of the defense in three different forms. *Id.* at 56-57.

175. The court treats Paperback's argument that the scope of protection for computer programs should be narrow as if it were a "policy argument," and consigned discussion of it to the "postscript" section of the opinion. *Paperback*, 740 F. Supp. at 77-79. Conceiving of the argument in this way allowed the court to consider it as an argument better addressed to Congress than to the courts which were bound to follow the law as it was. The court also regarded this argument as contrary to Congressional objectives of providing incentives to software innovators.

The "narrow scope for functional writing" principle should instead have been integrated into the body of the court's infringement analysis, for it is a well-established principle of copyright law which applied to the *Lotus v. Paperback* dispute just as clearly as it would to any other functional writing case. *LaST Frontier Report*, *supra* note 6, at 18-19. The CONTU Report and legislative history relevant to the scope of copyright protection for computer programs reflect Congressional intent that standard principles of copyright law of this sort were to be applied in computer program cases. See CONTU REPORT, *supra* note 5; *supra* note 4 and accompanying text.

176. *Paperback*, 740 F. Supp. at 55, regarded that the "central proposition" of *Synercom* was that nonliteral sequencing should always be treated as a circumstance in which idea and expression have merged. This is not the case, for in a footnote, the court in *Synercom*, 462 F. Supp. at 1013 n.5, indicated that in a proper case, nonliteral elements of computer programs might be protectable by copyright. In the usual case, said the court, "sequence, choice, and arrangement have only *stylistic significance*, rather than constituting as they would here the essence of the expression." *Id.* at 1014 (emphasis added).

Had the court in *Paperback* had a proper appreciation of *Baker v. Selden*, it would have realized why the court in *Synercom* ruled that the manner in which Synercom had formatted data for input to its structural analysis program for engineering projects was unprotectable by copyright law. There were, as the court in *Synercom* noted, more than three million ways to order the data for input into the structural analysis program. *Id.* at 1012. That did not, however, mean that there were three million ways to order the data that would be sound from an engineering standpoint. If the data were ordered in accordance with a method Synercom had devised, the ordering of the data would very likely be inextricably interconnected with this method. Under *Baker v. Selden*, that would make the ordering of the data a constituent element of the work's "idea." This appears to be the basis of the court's ruling in *Synercom*.

of nonliteral elements" defense which the court found easy to reject by citations to *Whelan* and its progeny.¹⁷⁷

Close examination of the court's discussion of the functionality issue reveals the heart of the court's concerns with a defense of this sort:

If, in a context, such as that of *Synercom* or of this case, an idea and its expression were taken to be inseparable and the expression therefore not copyrightable, copyright law would never, as a practical matter, provide computer programs with protection as substantial as Congress has mandated—protection designed to extend to original elements of expression however embodied.

I credit the testimony of expert witnesses that the bulk of the creative work is in the conceptualization of a computer program and its user interface, rather than in its encoding, and that creating a suitable user interface is a more difficult intellectual task, requiring greater creativity, originality, and insight, than converting the user interface design into instructions to the machine.

Defendants' contentions would attribute to the statute a purpose to protect only a narrowly defined segment of the creative development of computer programs, and to preclude from protection even more significant creative elements of the process.¹⁷⁸

Even though agreeing with Paperback that the Lotus interface was functional, the court insisted that:

[i]t does not follow that when an intellectual work achieves the feat of being useful as well as expressive and original, the moment of creative triumph is also a moment of devastating financial loss—because the triumph destroys copyrightability of all the expressive elements that would have been protected if only they had not contributed so much to the public interest by helping to make some article useful.¹⁷⁹

177. *Paperback*, 740 F. Supp. at 55. After citing a number of computer program cases that had accepted copyright protection for the "nonliteral elements" of programs, *id.*, the court acknowledged that *Softklone*, 659 F. Supp. 449, and *Plains Cotton*, 807 F.2d 1256, had taken a somewhat different view. *Paperback* does not mention that both had cited approvingly to *Synercom*, as had *E.F. Johnson v. Uniden*, 623 F. Supp. 1485, 1497 (D. Minn. 1985); *SAS Inst. v. S & H Computer Sys.*, 605 F. Supp. 816, 826 (M.D. Tenn 1985); and *Kepner-Tregoe*, 203 U.S.P.Q. (BNA) 124, 133. Thereafter, *Paperback* asserts that the *Synercom* input formats were "quite different" from the nonliteral elements of Lotus 1-2-3, although without saying how or why. 740 F. Supp. at 55.

178. *Paperback*, 740 F. Supp. at 56 (citations omitted). To clarify my analysis of the court's concerns, I have separated in the text these three statements. In *Paperback*, they appear as consecutive sentences in one paragraph of the opinion.

179. *Id.* at 57. Similar statements to that quoted in the text are found in the same section:

To hold [that a work was uncopyrightable because it was associated in the marketplace with a high degree of usefulness] would be to deny copyright protection to the most original and least obvious products of the creative mind merely because the marketplace accepts them as distinctively 'functional.' Such a rule would grant copyright protection for only those products that fall far short of being the best available." Such a rule "would offer incentives to market only the second, or third, or tenth best, and to hold back the best for fear that it is too good for copyrightability.

Id. at 58.

Following these statements, the court's analysis shifts to the idea/expression distinction as interpreted in *Whelan*,¹⁸⁰ and no further mention is made about the functionality of computer programs, user interfaces, or any particular aspects of them or the copyright consequences of their functionality.

These statements in *Paperback* reflect some fundamental misunderstandings of copyright law and principles. It is, for example, inappropriate to say that user interfaces should be protected by copyright law because the bulk of the creativity they embody resides in the "conceptualization" of them. The text of § 102(b) indicates that copyright does not protect "concepts" or "conceptualizations", no matter how creative or original they are.¹⁸¹ The most creative thing about Selden's book was undoubtedly the bookkeeping system explained in it, not the prose he used to describe it. Yet the Court nonetheless ruled that the scope of Selden's copyright was limited to his explanation of the system.¹⁸²

Nor is it proper to test whether an aspect of a copyrighted work is protectable by measuring how much hard work went into either creating the idea or implementing it in some concrete form. The Supreme Court's recent *Feist* decision demonstrates that "sweat of the brow" does not automatically signify the presence of "original expression" protectable by copyright law.¹⁸³ That creativity at the conceptual level does not always indicate that copyrightable expressiveness will be present in all aspects of the written implementation of the concept is shown by *Baker v. Selden*.¹⁸⁴

Near the end of *Paperback*, where the standardization defense was discussed, the court expressed similar concerns:

By arguing that 1-2-3 was so innovative that it occupied the field and set a *de facto* industry standard, and that, therefore, defendants were free to copy plaintiff's expression, defendants have flipped copyright on its head. Copyright protection would be perverse if it only protected mundane increments while leaving unprotected as part of the public domain those advancements that are more strikingly innovative.

Id. at 79. *But see supra* note 100 (discussion of the irrelevance of the novelty or nonobviousness of a work, or specific aspects of it, in determining the availability of copyright protection for it).

180. The functionality subsection concludes with this statement which shows how the court shifts away from functionality and into idea/expression:

[A] court, in determining whether a particular element is copyrightable, must not allow one statutory mandate—that functionality or usefulness is not itself a basis for copyrightability—to absorb and destroy another statutory mandate—that elements of expression are copyrightable. Elements of expression, even if embodied in useful articles, are copyrightable if capable of identification and recognition independently of the functional ideas that make the article useful. This mandate may be viewed as a corollary of the central distinction of copyright law between idea and expression

Paperback, 740 F. Supp. at 58.

181. 17 U.S.C. § 102(b) (1988), quoted *supra* in the text accompanying note 3.

182. *Baker*, 101 U.S. at 105.

183. *Id.* at 102; *Feist*, 111 S. Ct. at 1292-95.

184. *Baker*, 101 U.S. at 107 (ledger sheets not protectable by copyright law).

The "otherwise not enough protection" argument is also not a proper copyright argument.¹⁸⁵ Had such an argument been made in *Baker v. Selden*, the Court's likely response would have been "that's what patents are for."¹⁸⁶ Although there is some uncertainty at present about the patentability of computer program-related inventions, user interface patents are issuing with some frequency nowadays. The Lotus interface, or at least some elements of it, might well be eligible for a patent.¹⁸⁷ But even if patent protection was unavailable for certain valuable aspects of the Lotus interface, the "otherwise not enough protection" argument may more properly be construed as an argument for some *sui generis* protection for user interface features such as command hierarchies.¹⁸⁸

Although the court in *Paperback* did not accept the idea that achieving a optimally useful user interface for a computer program could result in a "devastating" loss of protection,¹⁸⁹ this result is consistent with traditional principles of copyright law. In a report on the application of copyright principles to computer programs, ten intellectual property scholars reached a consensus that

copyright should not protect aspects of an interface that optimizes, in a way for which there is no viable substitute, such design goals as rapid execution, accuracy of results, error reduction, number and speed of keystroke functions, or time, effort, or cost of becoming skilled at using the program. Such functionally optimal aspects of an interface should not be protected regardless of whether the original designer consciously employed systematic design analysis aimed at optimization or simply discovered an optimal interface by intuition.¹⁹⁰

185. See *Feist*, 111 S. Ct. at 1289-90. The strongest argument for "bending" the originality requirement of copyright law to allow "sweat of the brow" protection for compilations was that without copyright protection, there would be inadequate incentives to invest in the socially desirable activity of compiling information. See Ginsburg, *supra* note 121, at 1899. Yet, the Supreme Court in *Feist* expressly rejected this rationale for recognizing copyright protection for these kinds of works. See *Feist*, 111 S. Ct. at 1295.

186. *Baker*, 101 U.S. at 102.

187. In litigation with Lotus, Borland has argued that because Lotus had sought patent protection for a similar command structure to that of 1-2-3, Lotus has recognized that command structures were patentable rather than copyrightable subject matter. The court has rejected this as inconsistent with *Mazer*, 347 U.S. 201. Borland Order, *supra* note 16, at 27-29. For a critique of the court's reasoning on this issue, see *supra* notes 81-82.

188. See, e.g., *Legal Protection of Screen Displays*, *supra* note 14, at 355; Abramson, *supra* note 14, at 9-10.

189. *Paperback*, 740 F. Supp. at 57. Although there are places in *Paperback* where it might seem that the court regarded the Lotus interface as an optimally functional (see, e.g., *id.* at 57-58, 79), there was no finding on this point. It is not asserted here that the Lotus 1-2-3 user interface was functionally optimal. The point is rather that the court in *Paperback* was unable to accept the idea that a functionally optimal interface might be unprotected by copyright law despite the fact that it would be consistent with traditional principles of copyright law to hold that a functionally optimal interface is unprotectable by copyright.

190. *LaST Frontier Report*, *supra* note 6, at 28.

An optimal computer program user interface would, in these scholars' views, be an instance of "idea/expression merger."¹⁹¹ The conferees did not expect it would be easy to establish a "functional optimality" defense, but regarded it as consistent with copyright principles to recognize it.¹⁹²

That copyright law does not protect creative concepts, hard work, optimally efficient expressions, or other valuable elements of works failing to satisfy copyright standards may, on occasion, seem to lead to unfair results. But as the Supreme Court recently observed:

It may seem unfair that much of the fruit of a compiler's labor may be used by others without compensation. As Justice Brennan has correctly observed, however, this is not "some unforeseen byproduct of a statutory scheme." It is, rather, "the essence of copyright", and a constitutional requirement. The primary objective of copyright is not to reward the labor of authors, but "[t]o promote the Progress of Science and useful Arts." Art. I, sec. 8, cl. 8 To this end, copyright assures authors the right to their original expression, but encourages others to build freely upon the ideas and information [or other uncopyrightable elements] conveyed in a work.¹⁹³

The limiting principles of copyright law reflected in the text of § 102(b) are also not unintended byproducts of the law, but of its very essence.

Although the functionality of the Lotus program and some aspects of its user interface means that the work will have a narrower scope of copyright protection than more fanciful works would have, there is no question but that a computer program copyright can be infringed if expressive aspects of the work are copied.

C. Two Kinds of Expressiveness Inquiries That *Paperback* Might Have Made

Thus far, this section has mainly criticized *Paperback* for its inadequate analysis of the "expressiveness" of the Lotus 1-2-3 interface and for its failure to consider a number of copyright precedents and principles. This subsection will present two ways that the court might

191. *Id.* at 27-28.

192. *Id.* at 28. The "moment of achievement as moment of loss" argument is also clearly inconsistent with *Baker v. Selden*. The moment of Selden's crowning achievement was the moment of devastating loss, for the Supreme Court ruled that by publishing his book, Selden had dedicated the useful knowledge in the book to the public domain (unless the system was patented). *Baker*, 101 U.S. at 104. Baker was free to copy the most important and valuable elements of the book because they were constituent elements of the useful system described in it. That the elements could be said to be "original" and "expressive of the system," that they were made up of lines on paper and arrangements of words, did not change the Court's thinking, either about their utility or about the inextricable interconnection of idea and expression in the ledger sheets.

193. *Feist*, 111 S. Ct. at 1289-90 (citation omitted).

have analyzed the expressiveness problem in *Paperback*, either of which would have been more consistent with traditional principles of copyright law than the one the court employed.

Two analytic paths seem necessary because *Paperback* does not seem to be consistent about whether it conceived of the Lotus interface as a "literal" or "nonliteral" element of the program, or the case as involving "literal" or "nonliteral" copying.¹⁹⁴ In many places in the opinion, the court seems to have accepted Paperback's characterization of the Lotus interface as a nonliteral element of the program. Indeed, much of the opinion discusses at length the protectability of nonliteral elements of programs.¹⁹⁵ Yet, the court also seems to rely on near verbatim copying when ruling that Paperback infringed Lotus' copyright.¹⁹⁶

This subsection will first discuss how an expressiveness inquiry might have proceeded in *Paperback* had the court consistently conceived of the user interface as a nonliteral element of the Lotus program and the case as one involving nonliteral copying. It will then discuss how the case might have been analyzed if the court had consistently conceived of the user interface as a literal element of a program and the case as one involving literal copying.

1. PAPERBACK AS A "NONLITERAL COPYING" CASE

Copyright law regards "literal copying" as the reproduction of the whole or a substantial portion of the text of a copyrighted work. Copyright statute defines "computer program" as "a set of statements or instructions to be used directly or indirectly in a computer to bring about a certain result."¹⁹⁷ Thus, the reproduction of the whole or a substantial portion of the source or object code of a program involves "literal copying."

The user interface of a computer program is not among the set of statements or instructions constituting the program. It is among the "results" that can be brought about when a computer executes the program instructions.¹⁹⁸ Neither the statutory definition of computer

194. *Paperback* is not alone in having a confused conception of the relationship between computer programs and user interfaces, or as some courts have perceived the issue, between programs and the screen displays they produce. See *infra* Appendix.

195. See, e.g., *Paperback*, 740 F. Supp. at 50-53.

196. *Id.* at 70 (observing that VP-Planner uses the same command tree as 1-2-3 and stating that the two works were strikingly similar, so much so that users could be confused about which program was being operated).

197. 17 U.S.C. § 101 (definition of "computer program").

198. GOLDSTEIN, *supra* note 47, § 2.16. The unusual nature of computer programs is such that not only is the user interface of a program among the results of executing the program instructions, but it is also the means by which a user can cause the program to bring about additional results (such as printing a copy of a chart or graph that the user of the spreadsheet program might have created for a report he or she needed for a meeting the next morning).

program nor the legislative history of the provision indicates that the "results" achieved when program instructions are processed in the computer are protectable by copyright law. One reason to regard a user interface as a "nonliteral" element of the program is that one can read the entire source code of the program and not find the user interface there as a literal part of the text.

This is not to say that there is nothing in the text of a program from which one could discern things about the user interface. But the user interface is embodied in the program text as an abstraction. This is why it is appropriate to conceive of a user interface as a nonliteral element of the program. Lawyers will describe a case as involving "nonliteral copying" when more abstract elements of the work than the text have been copied. The classic example is the copying of plot elements from a dramatic play. In a proper case, the copying of a detailed sequence of events within the play can give rise to copyright liability, even though all of the lines of dialogue in the defendant's play might be different from those in the plaintiff's play. Under Judge Hand's "patterns of abstractions" test, some nonliteral elements of a work, however, are too abstract or at too high a level of generality to be protectable by copyright law.

The abstractness of a user interface such as Lotus 1-2-3 may not be readily apparent to those who have not developed computer programs. The abstract character of command names displayed in a menu of commands for a spreadsheet program's user interface can be understood by considering that each command stands at the peak of the hierarchy of abstractions for the code that will implement that particular function. For example, assume that command "move" was associated with certain lines of the program source code. These instructions would be part of the "literal text" of the program. A detailed line-by-line summary of this sequence of instructions would be a structural abstraction for those lines of code. They could, therefore, be accurately described as a "nonliteral" element of the program. Even more abstract (that is, less detailed) representations of this sequence might also be made; these too would be nonliteral in nature. The most abstract representation of the function for that segment of the program would be the command name that would appear in the user interface. In this way of understanding the command name, it is a "nonliteral" element of the program, but one of a much more abstract character than those that might represent the underlying structure of the program.¹⁹⁹

So if one were constructing a pattern of abstractions for a spreadsheet program, one would have to say that the commands for its user interface would be among the most abstract elements of the

199. See Alfred Z. Spector, *Software, Interface, and Implementation*, 30 JURIMETRICS J. 79, 86 (1989) (a computer scientist's discussion of the abstractness of user interfaces).

program.²⁰⁰ The copying of a command, or even a set of commands, is the copying of a far more abstract element of a program than a line-by-line synopsis of the source code of a program. Lotus was not claiming that Paperback had copied the source or object code of its spreadsheet program. Nor was it claiming that Paperback had copied the detailed internal "SSO" of the Lotus program. Lotus' main complaint was that Paperback had copied the command structure of the user interface of its program.

If one conceives of a command structure as a nonliteral element of the program which identifies the functions that the program can perform, it is understandable that it would be regarded, as it is in the Ninth Circuit, as the copying of a list of ideas which is not actionable under copyright law.²⁰¹ Treating the command structure as an "idea" is also consistent with Hand's "pattern of abstractions" test which treats higher level generalities as "ideas."²⁰² In addition, this result would be

200. Although the first letter of the command term might be an even more abstract representation than the word itself, it has become standard practice in user interface design to represent commands by names rather than by one-letter representations because the mnemonics of the names aid users to remember with which function they are associated. See Perlman, *supra* note 151, at 373.

201. See *supra* notes 158-61 and accompanying text.

202. Another sign of how abstract the Lotus interface is, and how little Lotus has treated the 1-2-3 command structure as part of the program's expressive authorship, is the large number of books written about Lotus 1-2-3. These books reproduce the Lotus commands and their arrangements, and go to great lengths to explain how the commands can be used to accomplish certain tasks. There are more than two hundred listings in the current issue of Books in Print having "Lotus 1-2-3" in the title, among them: BILL KLING, *THE ABC'S OF LOTUS 1-2-3* (3d ed. 1991); Ira Krakow, *LOTUS 1-2-3: SELF TAUGHT* (1989); QUE CORP., *USING 1-2-3* (1987); ALAN SIMPSON & PAUL LICHTMAN, *THE FIRST BOOK OF LOTUS 1-2-3, RELEASE 2.3* (H.W. Sams & Co. 2d ed. 1991); MARK M. WILLIAMS & NELDA M. RINCKENBERGER, *EXPERT ADVISOR 1-2-3, RELEASE 3.0* (1990); and JEFF WOODWARD, *TEACH YOURSELF LOTUS 1-2-3, RELEASE 2.2* (1990).

If the Lotus command structure was truly "expressive" in a copyright sense, it would seem unnecessary for there to be so many books with so many hundreds of pages of text to explain how to use the commands, and navigate the structure, to accomplish different kinds of spreadsheet functions. This in itself suggests that the Lotus command structure is part of a "system" the program's developers had devised for performing a set of spreadsheet functions, even apart from the role of the command structure in the macrocommand facility. Whole chapters of the books listed above are given over to discussion of specific commands. See, e.g., KLING, *supra* note 202, ch. 9 (lessons on moving, inserting, and deleting in 1-2-3).

Lotus, who against Paperback has asserted that reproducing these aspects of its interface is copyright infringement, has not sued the authors of these books, even though these authors have reproduced many more of the Lotus commands and their arrangement than Paperback which reproduced the Lotus command words only once per program. See, e.g., QUE CORP., *supra* note 202, at 496-97 (reproducing Lotus screen displays) and WILLIAMS & RINCKENBERGER, *supra* note 202, at app. A (showing representations of Lotus menu screens). If Lotus sued the authors of these books for copyright infringement because they reproduced Lotus' commands, command structure, and mode of presenting the commands, the book authors would surely defend such a lawsuit by asserting that they were merely explaining the Lotus spreadsheet system to users. Lotus is, of course,

consistent with the copyright cases giving a narrow scope of copyright protection to functional writings, of which the Lotus spreadsheet program would be one.

2. PAPERBACK AS A "LITERAL COPYING" CASE

One can also, however, conceive of the user interface of a program as an aspect of the work is capable of having expressive elements that can be literally copied. Consider, for example, that a user interface may present (and often does) textual explanations of program functions or colorful pictorial images that if printed in a book would clearly be protectable by copyright law. If one software developer copies the exact explanatory text or pictorial images from another firm's user interface, there might well be the kind of literal copying that would be actionable as copyright infringement.²⁰³

Although Judge Keeton has characterized *Paperback* as a case in which "without dispute, the 1-2-3 user interface as a whole was copied,"²⁰⁴ this is not really so. Consider the differences *Paperback* identifies between the Lotus and VP-Planner user interfaces: VP-Planner had different opening screen displays than Lotus 1-2-3. VP-Planner's help screens were also organized differently than Lotus' and contained different explanatory materials. VP-Planner's screens were generally wider than Lotus'; VP-Planner allowed users to hide certain columns of data in ways that the Lotus program did not permit. The menu bar of VP-Planner initially appeared at the bottom of the screen, whereas the menu bar of Lotus 1-2-3 was at the top of the screen (although users of VP-Planner could move its menu bar to the top of the screen if they preferred to have it there). *Paperback's* menu of commands also generally listed "help" as the first command in the set. For some submenus, VP-Planner had some additional commands relating to the data base capability VP-Planner provided that 1-2-3 did not have. VP-Planner also used somewhat different wording for explanatory long prompts.²⁰⁵

There are clearly some aspects of the Lotus 1-2-3 user interface that are "expressive" in a copyright sense. The text of the opening screens, the

unlikely to sue these authors. Precisely because the 1-2-3 interface is so unexpressive (in a copyright sense), Lotus benefits from the efforts of authors whose books explain to users how to implement various spreadsheet functions by interacting with the Lotus interface. Lotus seems to be more concerned about protecting itself from competition in the electronic spreadsheet market than in protecting the command structure more generally.

203. Even if the whole of a highly expressive user interface was copied, it is improper to regard the case as though the whole of the copyrighted work had been copied. At most, such a case should be described as a "fragmented literal similarity" case if the program itself has not been copied. See NIMMER ON COPYRIGHT, *supra* note 81, at § 13.03 [A][2].

204. Borland Order, *supra* note 16, at 4.

205. *Paperback*, 740 F. Supp. at 70.

explanatory text in the on-line help facility, and the explanatory long prompts would all seem to satisfy copyright standards for originality of expression.²⁰⁶ Paperback, however, did not copy these aspects of the Lotus interface, which is why this article has questioned the *Paperback* ruling.

The court identifies only four aspects of the Lotus user interface as having been copied by Paperback: (1) the command hierarchy of 1-2-3, (2) the mode 1-2-3 used to present the command hierarchy on the screen (that is, a two-line moving cursor menu displayed at the top of the screen), (3) the rotated "L" spreadsheet grid, and (4) the use of the "/" key to invoke the menu of commands.²⁰⁷ Even leaving aside the serious questions raised earlier in this article concerning whether the command hierarchy is unprotectable as a constituent part of Lotus' macro system, the blank forms cases and *Ashton-Tate v. Ross* cast doubt on Lotus' claim there is sufficient "expressiveness" in the Lotus command hierarchy for copyright infringement to be found based on the copying of it alone.²⁰⁸ Because the court regarded the other three things copied by Paperback to be unprotectable "ideas," it is questionable whether a court following traditional principles of copyright law could be persuaded that merely combining four unprotectable ideas of this sort into a user interface would convert the interface into an "expressive" compilation.²⁰⁹

While the author has questioned whether Paperback copied expressive elements of the Lotus interface, it may be that upon appropriate detailed findings and analysis, the court might have found that Paperback infringed the Lotus copyright. The most clearly objectionable thing about VP-Planner was that it was a "clone."²¹⁰ Although the court did not speak of VP-Planner as a clone, Lotus offered evidence in the case to show that it was.²¹¹ The court was clearly impressed by the fact that users could move the VP-Planner menu bar from the bottom of the screen to the top, and because some of the VP-

206. The scope of copyright protection afforded to these expressive aspects would likely be quite narrow in line with the "narrow scope for functional writing" rule discussed *supra* in subsection V-A. See, e.g., *Kepner-Tregoe*, 203 U.S.P.Q. (BNA) at 134 (minor changes in wording necessary to avoid infringement in functional writing case because of the unprotectability of most of the content).

207. *Paperback*, 740 F. Supp. at 65-70.

208. This issue is presented in Lotus' copyright infringement suit against Borland International.

209. See *Brown Bag*, 960 F.2d 1465.

210. As the name of the defendant alone might suggest, *Softklone*, 659 F. Supp. 449, also involved copyright infringement claims against the producer of a clone of a popular software product.

211. Interestingly, *Paperback* never refers to VP-Planner as a "clone," even though Lotus complained of this to the court. One of Lotus' expert witnesses offered a definition of "clones" of software products which the Paperback product may have satisfied. See Affidavit of Bernard Galler para. 91, *Lotus Dev. Corp. v. Paperback Software Int'l* (CA No. 87-0076-K).

Planner screens displayed an array of commands identical to Lotus', "a user could easily think 1-2-3 rather than VP-Planner was the program in use."²¹² Although this statement seems to reflect a more trademark-like than traditional copyright analysis of the case,²¹³ had the court ruled against Paperback because VP-Planner was a "clone" and had the court defined "clone" with precision, it might have developed a persuasive rationale for interpreting copyright law to make clones illegal. There was some exact copying in *Paperback*, and exact or near-exact copying of aspects of a functional writing can be a copyright infringement. A ruling of this sort in the *Paperback* case might not only have been reconciled with traditional principles of copyright law, it could also have provided significant guidance to the software industry about what aspects of user interfaces could properly be protected by copyright law and why.

VI. CONCLUSION

Whether an aspect of a copyrighted work will be protected by copyright law should be tested by long-standing principles of copyright law which require that the aspect be "expressive." The court in *Paperback* did not offer a convincing explanation of what was expressive about the aspects of the Lotus interface on which the finding of infringement was based. In particular, it neglected to consider the implications of its own statements which indicated that an important element of the Lotus interface, namely, the Lotus command structure, was a constituent element of the Lotus macro system.

A user interface is among the "results" generated when program code is executed in a computer. Some program results, such as highly fanciful videogame graphics, may have sufficient "expressive" content that copyright will and should protect them, but a user interface (or other program result) should not be automatically categorized as "expression" merely because it is a result, merely because it is valuable, or merely because hard work or creativity was required to bring it into being.

212. *Paperback*, 740 F. Supp. at 70.

213. That is, it seems to view the problem the case presented as though Paperback had copied the trade dress of 1-2-3, and infringement should be found because the two works were confusingly similar. Trade dress theory for protecting the "look and feel" of computer programs has been suggested by some commentators. See, e.g., Richard A. Beutel, *Trade Dress Protection For the "Look and Feel" of Software: A New Source of Proprietary Rights Protection for the Software Industry?*, 5 *COMPUTER L.* 1 (Oct. 1988).

One of the grounds on which Borland has sought to distinguish its dispute with Lotus from the Paperback dispute is that no one could mistake its Lotus-compatible alternative interface for the Lotus interface because of some distinctive characteristics of the Borland screens and their presentations of the commands. Borland Brief, *supra* note 157, at 11.

"Results" of computer programs will often be functional in nature and beyond the scope of copyright protection.²¹⁴

Although insisting on the necessity to follow the "mandates" of Congress, the court in *Paperback* failed to do so. The CONTU Report makes clear that the Commission expected the courts to look to *Baker v. Selden*, other functional writing cases, and § 102(b) in interpreting copyright law and applying it to computer programs.²¹⁵ CONTU's assurances to Congress that the principles embodied in these sources would yield a proper balance of the interests of program authors, competitors, and the public are consistent with the views that led Congress to make § 102(b) a part of the statute a few years before.

The overbroad ruling in *Paperback* continues rather than resolves the controversy about the protectability of user interfaces by copyright law. It is unfortunate that only by many more years of litigation can the software industry get definitive guidance on these important issues.

214. A recent article reports that advances in computer modelling and tools for analysis of data from materials research labs have led to significant advances in the development of materials such as "extra-tough steel . . . for bearings in space shuttles." Otis Port, *The New Alchemy*, *BUS. WK.*, July 29, 1992, at 48. These new materials may be the "results" of processing programs in computers, but it would be improper to consider them derivative works within the scope of the programmer's copyright. See generally Pamela Samuelson, *Allocating Ownership Rights in Computer-Generated Works*, 47 *U. PITT. L. REV.* 1185 (1986).

215. CONTU REPORT, *supra* note 5, at 18-23.

APPENDIX

On The Relationship Between Computer Programs and Their User Interfaces and Copyright Law's Confused Conception of It

Few issues of computer program copyright law have confused the courts and the U.S. Copyright Office more than attempting to grasp the relationship between a computer program and its user interface. To enable readers to understand the potential significance of the *Paperback* ruling that "user interfaces" are "copyrightable" parts of computer programs, it may be useful to review how the matter had previously been dealt with by the courts and the Copyright Office.

The story starts with the Copyright Office decision in the early 1980's to begin registering videogame programs as audiovisual works.²¹⁶ Somewhat later, the Office began to issue separate registration certificates for the videogame programs.²¹⁷ Videogame caselaw then began to distinguish between program and audiovisual copyrights, for in some cases, a defendant would be found to have infringed the audiovisual but not the program copyright, in other cases to have infringed the program copyright but not the audiovisual copyright, and sometimes both would be infringed.²¹⁸ Registration practices and caselaw of this sort seemed to suggest that a firm needed one copyright to cover the program, and a second for whatever aspects of the program might be displayed on the screen.

Whelan was the first nonvideogame computer program copyright case in which user interface similarities became an issue. The trial court in *Whelan* relied heavily on similarities in the screen displays produced by the plaintiff's and defendant's programs as a basis for concluding that *Whelan's* copyright had been infringed.²¹⁹

On appeal, Jaslow pointed out that *Whelan* was not charging infringement based on screen display similarities. Rather, her claim was that Jaslow had copied aspects of the underlying program. Jaslow noted that very differently structured programs could produce substantially similar or identical screen displays, and hence, the trial judge had erred in relying on screen display similarities as the basis for finding copyright infringement as to the underlying program.²²⁰

The Third Circuit agreed with Jaslow's argument, but only to a point. The court agreed that relying solely on screen display similarities

216. See, e.g., *Stern Elec., Inc. v. Kaufman*, 669 F.2d 852 (2d Cir. 1982).

217. See, e.g., *Williams Elec., Inc. v. Artic Int'l, Inc.*, 685 F.2d 870 (3d Cir. 1982).

218. See, e.g., *Williams Elec.*, 685 F.2d 870; *M. Kramer Mfg. Co. v. Andrews*, 783 F.2d 421 (4th Cir. 1986); and *Midway Mfg. Co. v. Strohon*, 564 F. Supp. 741 (N.D. Ill. 1983).

219. See *Whelan*, 609 F. Supp. at 1322.

220. See *Whelan*, 797 F.2d at 1242-45.

to prove infringement as to structural elements of the underlying program would be error, but decided it was not error to consider screen similarities as some evidence of copying of underlying program elements. Because there was other evidence in the record that the court regarded as supporting Whelan's claim of copying of underlying structural elements of the program, the appellate court affirmed the infringement ruling.

Broderbund was the next nonvideogame software copyright case in which user interface/screen display similarities were discussed.²²¹ In *Broderbund*, the infringement claim was based exclusively on screen display similarities, specifically, the choice and arrangement of command terms on a series of menu screens for a printing program. Although the court was somewhat equivocal about whether *Broderbund* had an audiovisual or a program copyright or both, it found comfort in the *Whelan* decision which it interpreted as a screen display similarity case. The court in *Broderbund* used the *Whelan* test for software copyright infringement. This allowed the court to sidestep the issue of what kind of copyright it was dealing with.

Softklone, like *Broderbund*, involved claims of infringement based solely on user interface/screen display similarities, that is, in the arrangement of command terms on menu screens.²²² The judge in *Softklone* rightly pointed out that the court in *Broderbund* had misinterpreted *Whelan* by characterizing it as a screen display similarity case. Taking more seriously than had the appellate court in *Whelan* the point that different programs could produce the same screen displays, and substantially similar programs could produce different screen displays, the court in *Softklone* ruled that the program and the user interface were separate works requiring separate copyrights, a result consistent with the videogame cases. Fortunately for the plaintiff, it had obtained a separate copyright for its main menu screen as a compilation of terms. The court found infringement because the defendant had arranged the command terms in the same way as the plaintiff on one portion of the menu screen, and because the defendant had also used the same highlighting and capitalization of the first two letters of the command terms as had the plaintiff.²²³

Not surprisingly, the *Softklone* decision, insofar as it ruled that separate copyrights were necessary to cover user interface elements of programs, made some software developers quite nervous. Soon thereafter, the Copyright Office initiated an inquiry about the separate

221. 648 F. Supp. 1127.

222. 659 F. Supp. 449.

223. Interestingly, the court ruled that it was not an infringement to have the same set of command terms as another program, but only to arrange them in the same way when they could be arranged differently. *Id.* at 460.

registration issue.²²⁴ Although the Office held hearings at which witnesses discussed reasons that separate registrations of programs and screen displays might be desirable, the Office decided against separate registrations.²²⁵ Current Copyright Office policy is that if the predominant character of a program is audiovisual (such as a videogame), the program should be registered as an audiovisual work. If not, it should be registered as a literary work. Regardless of which registration form is used, the Office considers the program and its screen displays to be one work.²²⁶

This, then, was the state of the law on this issue when *Paperback* was decided. *Paperback* relied on *Softklone* to support its contention that the Lotus interface was unprotected by the copyright Lotus had registered for the program. Lotus, of course, thought that the Copyright Office "one work" policy statement reflected the proper legal approach. Although some aspects of the prior user interface caselaw supported Lotus' position, none of the prior cases seems to have been completely suitable to its purposes. Lotus seems to have decided to take a bolder and more direct approach to litigating its copyright infringement claim against *Paperback*, seeking protection not for the screen displays 1-2-3 generated (the issue as to which the separate registration controversy had arisen), but rather for its "user interface" as an important part of the program.

The more conventional path for the court to have taken in *Paperback* would have been to compare the screen displays of the two programs and to have regarded the screen displays as literal parts of the program under the Copyright Office's "one work" theory. The screen displays then might have been protected from copying by a competitor for reasons given in *Broderbund* and *Softklone*. (For the sake of this discussion, I will leave aside the § 102(b) problem with protecting aspects of the Lotus interface and the conflict between *Broderbund* and *Ross*). The reason prior caselaw had focused attention on similarities in screen displays was because the aspects of user interfaces that are most readily analyzable as copyright problems are those that involve the display of words and symbols on computer screens.²²⁷ More abstract elements of user interfaces, such as the pairing of particular functions to particular keys on

224. See Notice of Inquiry, 52 Fed. Reg. 28,311 (1987).

225. This may have been more as a matter of administrative convenience (e.g., it would minimize the number of forms the Office would have to process) than because the Office had definitively resolved the copyright dilemma about the proper characterization of the relationship between computer programs and user interfaces. See *Hearings Before U.S. Copyright Office on Registration and Deposit of Computer Screen Displays* (Sept. 9-10, 1987).

226. The Office has left to the courts the task of deciding what elements of programs are protectable by copyright law. Notice of Registration Decision: Registration and Deposit of Computer Screen Displays, 53 Fed. Reg. 21,817 (1988).

227. See, e.g., *Whelan*, 797 F.2d 1222; *Manufacturers Technologies*, 706 F. Supp. 984; *Broderbund*, 648 F. Supp. 1127.

the computer keyboard, have generally been viewed as outside the copyright realm.²²⁸ By abjuring the prior caselaw's focus on "screen displays" and embracing the copyrightability of "user interfaces,"²²⁹ *Paperback* may have opened the door to a considerable broadening of the scope of copyright protection for user interfaces.²³⁰

Although the court in *Paperback* does not offer its own definition of "user interface,"²³¹ it quotes Lotus' description of its user interface as including "such elements as 'the menus (and their structure and organization), the long prompts, the screens on which they appear, the function key assignments, [and] the macro commands and language.'"²³² *Paperback* does not explicitly say that function key assignments are protectable by copyright law, but given some of the expansive language in the opinion, it would be consistent with the approach in *Paperback* to say that they were.²³³

228. See, e.g., *Manufacturers Technologies*, 706 F. Supp. 984 (similarities in navigational elements of the user interface ruled not protectable by copyright); *Symantec*, 12 U.S.P.Q.2d (BNA) 1991 (rejecting claims of copyright infringement for similarities in the pairing of certain keys to certain functions).

229. *Paperback*, 740 F. Supp. at 79-80 (infringement said not to be based on screen display similarities, but upon similarities in user interfaces). *Paperback* makes a point of distinguishing its ruling in this respect from the prior user interface cases, particularly *Softklone*. *Id.* The decision to conceptualize the user interface as a protectable element of the Lotus program may help to explain why there is so little discussion of or reference to the other user interface/screen display cases in the *Paperback* opinion.

230. Had the court not found the Lotus interface to be a "copyrightable" element of the Lotus program, *Paperback* indicates that the defendant might still not have won the case as a whole. The second phase of the trial would have determined whether Paperback copied copyrightable elements from the source or object code of the Lotus program. See *Paperback*, 740 F. Supp. at 42. This reflects a view of the relationship between the program and the user interface consistent with that in the *Whelan* case (wherein user interface similarities might be considered as some evidence of copying the underlying structure of the program).

231. A recent decision by Judge Keeton in the *Lotus v. Borland* case suggests that he is still confused about what a user interface is. The opinion refers to *Paperback* as a case in which "without dispute, the user interface as a whole was copied." Borland Order, *supra* note 16, at 4. As Section V-C *supra* has pointed out, this was not really true.

Further evidence of the court's confusion about what a "user interface as a whole" is can be seen from the court's recent statement that a reasonable trier of fact could find on disputed facts that Borland took the Lotus user interface as a whole. *Id.* at 5. In fact, it would be impossible for a reasonable trier of fact to find that Borland had copied the whole of the Lotus user interface. What Borland copied from the Lotus interface was the 1-2-3 command hierarchy which is used in Quattro Pro's 1-2-3 emulation interface. This is an important component of the Lotus interface but not the whole of it.

232. *Paperback*, 740 F. Supp. at 63 (emphasis added).

233. Because function key assignments are more specific than the general purpose or function of the program, the court's test for copyrightability would treat them as presumptively expressive. If there are more than a small number of ways to assign the function keys (as would be likely with a user interface like Lotus'), this would make them conclusively expressive. The only question then would be whether they were a substantial component of the program, and to use *Paperback's* reasoning, if the defendant bothered to copy them, they must be substantial parts of the program. See *supra* Section III-C.

The court in *Paperback* deserves some praise for redirecting the copyright discourse on this subject away from a discussion of the protectability of "screen displays" and toward the protectability of "user interfaces." The latter phrase is the accepted term in the technical community. Copyright law should describe computer programs and user interfaces in a technically appropriate manner.²³⁴

At the same time, one can question *Paperback's* insistence that a "user interface as a whole" is copyrightable.²³⁵ Although some computer program user interfaces may consist principally of text and graphics, other user interfaces may consist of devices such as joy sticks, power gloves, and/or dials and switches.²³⁶ These user interface elements are too functional to be protectable by copyright law.²³⁷

As for the issue of how to conceive of the relationship between a computer program and a user interface, it will generally suffice to do as the Copyright Office has recommended: to treat programs and their user interfaces as one work (if indeed they are so), and in the event someone copies parts of a program or user interface, to apply traditional principles of copyright law to determine whether what has been copied is "expressive" in a copyright sense.

That it is appropriate to consider computer programs and their user interfaces as "one work" and to apply copyright law only to "expressive" aspects of them can be illustrated with an example of another kind of machine and its user interface. Consider, for example, the relationship between the internal working parts of a wristwatch machine and its user interface. The wristwatch—which consumers tend to consider in an integrated way as a watch, rather than as two things, a machine and its interface—often has a user interface consisting of a face with numbers on it and two hands fixed at the center of the face.²³⁸ These aspects of the watch's user interface present a visual appearance. They also display information in an efficient manner (which we refer to as "telling" us "the time"). Another part of the user interface of the watch is the externally visible device with which one can wind and set the watch to the proper hour. This controls the functioning of the internal working parts, rather than displaying information.

234. See, e.g., *Altai*, 755 F. Supp. at 559-60 (rejecting certain aspects of *Whelan*, 797 F.2d 1222, as technically incoherent as well as at odds with copyright doctrine).

235. Not all computer programs have user interfaces. Some interface only with other programs or with computer hardware.

236. See, e.g., Donald A. Norman, *Turn Signals Are The Facial Expressions of Automobiles* 59-71 (1992).

237. They may, however, be patentable.

238. As with computer programs and their user interfaces, it is possible for a watch with the same user interface as another to have different internal working parts, and for a watch with the same internal working parts to have a different user interface. In this sense, there is no necessary relationship between these two aspects of the watch.

Some parts of the user interface of a watch may be expressive in a traditional copyright sense; some may not. If one puts, for example, a picture on the face of a watch (Mickey Mouse, for example), the picture may be protectable by copyright law as an original expression. Some parts of the user interface of a watch, although conveying information or displaying an appearance, will be unprotectable by copyright law because they are conventional (i.e., standard) elements of watches (for example, the long "hand" indicating minutes, and the shorter "hand" hours). Other parts of a watch user interface, however, such as the device one uses to set the watch to the proper time, are too functional to be protectable by copyright law.

Similar distinctions can be drawn between expressive and unexpressive aspects of computer program user interfaces. *Paperback* is to be commended for conceiving of computer screen displays as part of computer program user interfaces, but criticized for its failure to apply traditional principles of copyright law when judging the protectability of particular features of user interfaces. Congress did not intend for courts to remake copyright law to render all valuable aspects of computer programs protectable by copyright law. Other user interface cases have recognized that traditional principles of copyright law are to be applied in computer program cases,²³⁹ including the principles embodied in the limiting principles of § 102(b). It is their example, not *Paperback's*, that should be followed.

239. *Brown Bag*, 960 F.2d 1465.

