# BREAKING THE SILENCE OF A DIVIDED COURT: AN ANALYSIS OF THE FIRST CIRCUIT'S DECISION IN *LOTUS v. BORLAND*

**William F. Porter [n.a1]**

## I. Introduction

Long before the world entered the computer age, one of America's most renown jurists, Learned Hand, struggled with the boundary of copyright law, and reflected some of the frustration practitioners feel today. Hand declared "nobody has ever been able to fix that boundary, and nobody ever can." [n.1] While perhaps Hand overstated the difficulty of copyright analysis, recent copyright litigation lends at least some support to his conclusion.

Throughout the past 60 years, judges in this country have been forced to consider copyrightability in the face of technological progress. Yet in its purest form, the inquiry has not changed significantly and courts often struggle to discern protectable expression and separate it from unprotectable ideas.

Against a long history of copyright jurisprudence, the Lotus Development Corp. v. Borland International, Inc. line of cases promised to clarify the status of copyright protection as applied to the non-literal aspects of computer programs. The case originated in federal district court, where Judge Keeton rendered a series of extensive opinions regarding the copyrightability of various aspects of computer spreadsheets. However, the United States Court of Appeals for the First Circuit *274 reversed, rejecting not only Judge Keeton's conclusions, but also the methodology he employed in reaching those conclusions.

The issues raised in Lotus exemplified the growing discord among courts over how exactly to approach copyright protection problems that did not fit into any of the previous judicial decisions regarding software. When the Supreme Court agreed to review the case, it was thought the resulting decision would serve as a means of reconciling the growing discord regarding the limits of copyright protection in this field.

However, just nine days after oral argument, the Court issued a per curiam opinion, stating simply that "[t]he judgment of the United States Court of Appeals for the First Circuit is affirmed by an equally divided Court." [n.2] Many in the software industry had

hoped for a firm statement by the Court on the proper scope of copyright protection in the software related industry. Instead, the Supreme Court's 4-4 deadlock appears to give as much guidance as if the Court had denied certiorari. The argument by Judge Stahl at the level of the First Circuit Court of Appeals remains only persuasive precedent for those judges looking for guidance outside that jurisdiction.

Because of the continuing failure of the courts to satisfactorily resolve questions relating to software copyrightability, this article will analyze the Lotus decision within its present context to discern the extent of copyrightability of user interfaces. This article will also attempt to provide guidance in understanding the status that computer software technology holds within the body of copyright law.

Since the Supreme Court failed to offer any guidance on the issue of user interfaces, this article will examine the First Circuit's opinion, and will consider the decision's consistency with the intent of the copyright statute and the broader goals of copyright protection. It is the intent of this article to serve as an aid in determining the proper methodology courts should use in approaching modern computer copyright questions. Further, this article will attempt to analyze the degree to which the First Circuit's opinion, now with the smallest bit of support from the nation's highest court, should influence other jurisdictions grappling with these types of issues.

## *275 II. Lotus v. Borland: the Facts

Borland first introduced its Quattro spreadsheet program in 1987, marketing it as an improvement to existing applications, such as Lotus 1-2-3. [n.3] As part of the package, Borland included a menu command structure whereby Quattro users could interact with the spreadsheet either through Quattro's native mode or through the "Lotus Emulation Interface." [n.4] The emulation interface allowed Quattro users to choose how to interact with the Borland spreadsheet, either using Borland commands, or employing Lotus commands through the emulation interface. [n.5]

In 1990, a separatelegal action culminated in a federal district court holding that the protections afforded by copyright to Lotus 1-2-3 extended to the menu command structure (i.e., the structure that Borland had copied in creating its Lotus Emulation Interface). [n.6] That success prompted Lotus to file suit against Borland, alleging substantially similar infractions. [n.7]

In March of 1992, both parties moved for summary judgment, and the district court denied both motions. [n.8] Four months later, upon review of both sides' renewed motions for summary judgment, Judge Keeton denied Borland's motion, but granted the Lotus motion in part. [n.9] The judge held that the expressive elements of the Lotus menu command hierarchy could be distinguished from the functional aspects, thereby *276 allowing copyrightability. [n.10] However, the court reserved for a jury trial the determination of the scope of infringement. [n.11]

Borland responded by removing the emulation interface from its products. [n.12] However, Lotus compatibility could still be achieved through the use of a "key reader" system, whereby Borland's spreadsheet continued to recognize Lotus macros, but the Lotus menu was no longer displayed. [n.13] The district court then allowed Lotus to amend its complaint to include the key reader feature. [n.14]

Judge Keeton conducted two separate trials, one concerning the menu command hierarchy, and the other the key reader feature. [n.15] In both trials, Judge Keeton found in favor of Lotus. [n.16]

## III. Foundations of Copyright Law

All copyright law descends from Article I, Section 8, Clause 8 of the Constitution, which gives Congress the power "[t]o promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."

In the two centuries since the Constitution's ratification, courts have repeatedly recognized that the driving force behind this clause has been the economic belief that the best way to serve the public good would be to afford inventors and writers exclusive rights to their work. [n.17] The *277 Supreme Court has often repeated " s acrificial days devoted to such creative activities deserve rewards commensurate with the services rendered." [n.18] However, the goal of copyright law is also to ensure that the public at large reaps the benefit of an author's work, rather than to simply ensure an author of his just reward. [n.19]

Applying the constitutional mandate, the Supreme Court's first major discussion of copyright protection began with Baker v. Selden. [n.20] In that case, Selden had published Selden's Condensed Ledger, or Bookkeeping Simplified, describing a new system of accounting. [n.21] Included in the sale of the book were blank forms, consisting of lines and headings, to be used in practicing the system explained within the text. [n.22] The Court addressed the issue of whether Selden's copyright to the book imparted an exclusive right to the forms contained in the book. If allowed, copyright protection would have granted Selden patent-like exclusive rights to the method of bookkeeping disclosed within the book. [n.23]

The Court rejected Selden's claim, holding that Selden's copyright extended only to his expression of the disclosed system. [n.24] In contrast, the exclusive right to the method of operation could only be properly be obtained through patent law. Justice Bradley stated:

> 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. The object of the one is explanation; the object of the other is use. The *278 former may be secured by copyright. The latter can be secured, if it can be secured at all, by letters-patent. [n.25]

Decades later, the Supreme Court revisited the Baker question in Mazer v. Stein. [n.26] The issue there concerned the extent of copyright protection for certain statues that were being put to use as lamps. While works of art unquestionably obtain the protection of copyright laws, it was argued that the art's utilitarian ends, and the possibility of obtaining patent protection, precluded protection by copyright. [n.27]

The Supreme Court squarely rejected the option of denying copyright protection to utilitarian works. [n.28] In doing so, the Court emphasized the different objectives of patent and copyright laws. While patents protect novelty or invention, the copyright statute intends only to protect original expression, "conferring only the 'sole right of multiplying copies.'" [n.29]

Following Mazer, relatively few contested the idea that a utilitarian work could receive copyright protection. However, the Mazer Court was less forceful in explaining how lower courts could simultaneously protect the expression inherent in a manufactured article under copyright law while denying protection to the methods and usefulness of the article. The Court left this task to the individual district and circuit courts.

The doctrine that has evolved in response to these pronouncements by the Supreme Court has come to be known as the "merger" doctrine. Although various courts have stated the idea in different ways, a 1967 decision of the First Circuit neatly captured the evolution of this trend and voiced what was perhaps the first modern formulation of the merger concept. [n.30] Drawing on Justice Hand's abstraction analysis from nearly a half century earlier, [n.31] the court refused to extend copyright protection to subject matter so "narrow, so that 'the topic necessarily *279 requires' if not only one form of expression, at best only a limited number." [n.32]

The rationale behind this refusal is clear. Copyright protects the expression of ideas rather than ideas themselves. If copyright laws were extended to protect works where an idea necessarily requires a certain expression, or limited number of expressions, this would indirectly extend copyright protection far beyond its intended bounds. [n.33] In fact, such an extension would protect ideas rather than expression and would constitute a complete reversal of the intended role of copyright protection. The determination of whether idea and expression are merged generally requires a court to decide whether or not the "idea" is capable of expression in various forms. [n.34] If the idea can be expressed in a variety of different ways, one may conclude that the idea and expression have not merged, and that the expressionis entitled to copyright protection.

## IV. Copyright Law and Software: Current Formulations

The unique place that computer programs hold, both as written expression and as functional instruments, presents a twist in the body of copyright law. In fact, the issue of whether a computer program deserved copyright protection at all was hotly contested under early law. [n.35] However, Congress settled this debate in 1980 with the passage of the Computer Software Act of 1980. [n.36]

In general, to establish copyright infringement, a plaintiff must prove: (1) ownership of a valid copyright, and (2) copying of constituent elements of the work. [n.37] To satisfy the second prong of this test for *280 copyright infringement, a plaintiff is further required to show that the allegedly infringing work was copied, [n.38] and that the copying is so extensive as to render the two works "substantially similar." [n.39]

That the literal aspects of a program are protected by copyright is no longer contested. [n.40] However, in recent years the courts have also embraced the idea that a program's non-literal aspects may also be protected by copyright. [n.41] The difficulty arises when courts endeavor to ascertain the limits of copyright protection for these necessarily intangible non-literal elements. [n.42]

In an effort to provide a test for determining the copyrightability of non- literal components of a computer program, two courts have formulated similar three part tests to apply to this specific question: Judge Keeton (the District Court Judge in the Lotus-Borland litigation) articulated a test for copyrightability in Lotus Development Corp. v. Paperback Software International, [n.43] while the Second Circuit announced its "Abstraction- Filtration-Comparison" test in Computer Associates International v. Altai, Inc. [n.44] The two tests are similar judicial attempts at *281 providing a framework consistent with the history of copyright law, where non-literal copyrightable expression may be separated from unprotectable material.

## V. User Interfaces: Cases Against Copyrightability

In dealing with the copyrightability of the Lotus menu command hierarchy, the First Circuit states that "we are navigating in uncharted waters." [n.45] While, in fact, the specific issue of menu command hierarchies had yet to be addressed by the courts, Judge Stahl cast aside a significant amount of persuasive precedent dealing with input formats and interfaces. [n.46]

For example, in 1978, a federal district court in Texas handed down what remains perhaps the most influential argument in support of the conclusion reached in the Lotus case in Synercom Technology, Inc. v. University Computing Co. [n.47] At issue were input formats constructed for use in conjunction with computer software designed for engineering applications. [n.48] Judge Higginbotham recognized that these input formats expressed ideas [n.49] and further stated that the formats might be rendered eligible for copyright protection. [n.50] However, Higginbotham could not conclude validseparation of the idea and the expression, and therefore denied copyright protection, holding that the "order and sequence" were ideas rather than expression. [n.51]

*282 Further support for Judge Stahl's opinion in Lotus may be found in the Fifth Circuit's decision Plains Cotton Cooperative v. Goodpasture Computer Service, Inc. [n.52] There the court considered the copyrightability of input and output formats and

held the formats not copyrightable primarily because they were mandated by the externalities of the cotton market, for which the software had been designed. [n.53]

In a case involving the actual commands that were to be included in a computer program's user interface (rather than the sequence, structure and organization of the commands), the Ninth Circuit held that a mere list of commands could not receive protection via copyright law. [n.54]

The discussion of these three cases addressing the question of user interfaces in the context of computer programs is not exhaustive. However, the cases provide a solid foundation to understand why courts might refrain from extending copyright protection to user interfaces. As with any object of copyright protection, Synercom emphasizes that merger of idea and expression will preclude copyright protection. Plains Cotton and Ashton-Tate teach that when the claimed interfaces are dictated by the requirements of the product, these interfaces will not receive copyright protection. [n.55] Interestingly enough, in rejecting copyrightability of the interfaces, none of the three cases used the "method of operation" analysis that Judge Stahl would choose to use in the Lotus case.

*283 VI. User Interfaces: Cases in Favor of Copyrightability

In one of the earliest cases to deal with this issue, the Northern District of California [n.56] confronted the copyrightability of the menu screens and audiovisual displays that controlled a popular graphics application. [n.57] Although the defendants attempted to demonstrate that copyrighting these interfaces would impart exclusive rights to the method of controlling the graphics application, [n.58] the court rejected this contention. [n.59] The district court reasoned that because the choice of interfaces was dictated by aesthetic and artistic considerations, rather than "utilitarian and mechanical" ones, the interfaces deserved copyright protection. [n.60]

The same court that decided Plains Cotton has also shown support for the copyrightability of user interfaces. In Engineering Dynamics, Inc. v. Structural Software, Inc., [n.61] the Fifth Circuit declared that user interfaces, as part of a computer program's non-literal expression, could obtain protection. [n.62] The court explained further that the interfaces in question contained expressive elements [n.63] and explicitly held user interfaces to be within the proper scope of copyright law. [n.64]

The Tenth Circuit also has indicated its support for the copyrightability of user interfaces. In Autoskill Inc. v. National Educational Support Systems, Inc., [n.65] the court held that a keystroke sequence could be protected by copyright in the context of a computer program. [n.66] *284 This case was one of the user interface cases cited by the First Circuit in Lotus, but the accompanying analysis was given little attention by Judge Stahl. [n.67]

What guidance may be taken from these cases tending to support the proposition that user interfaces may be protected by copyright law? In each case, the issue generally

revolved around the extent to which the choice of interfaces was dictated by functional or efficiency conditions. [n.68] If a program's user interface can be chosen, the courts have indicated a willingness to extend copyright protection. On the other hand, where externalities dictate the choice of these interfaces, copyrightability will be defeated. [n.69] Thus, the extent of copyright protection properly includes a merger analysis.

## VII. Lotus v. Borland: First Circuit's Decision

After three years of litigation and four published decisions by Judge Keeton, Lotus Development Corp. v. Borland International, Inc. [n.70] finally made its way up to the Court of Appeals for the First Circuit. On appeal the sole issue concerned Borland's copying of the Lotus menu command hierarchy and the issue of copyrightability.

*285 Because of the facts unique to this appeal, Judge Stahl felt that the general rules governing copyright protection would add little to the case analysis. [n.71] Furthermore, despite the above-mentioned cases, [n.72] Stahl characterized the specific question concerning the copyrightability of the Lotus menu command hierarchy as one of first impression for the First Circuit, [n.73] requiring the court to look elsewhere for guidance.

As with many copyright cases, the point of departure for discussion of relevant authority is Baker v. Selden. [n.74] As previously noted, Baker held that the copyright on a book containing blank forms to be used in conjunction with the book's accounting system did not include an exclusive right to the copying of those blank forms. [n.75] However, despite the apparent similarities between accounting forms and modern spreadsheets, the court recognized that the issues differed significantly. [n.76] Specifically, while Baker argued that his copyright afforded him a monopoly over the forms for implementing his accounting system, Lotus alleged only a monopoly over its specific menu command hierarchy, which would allow competitors to create menu driven spreadsheets by creating their own hierarchies. [n.77]

Next, the court considered Computer Associates International Inc. v. Altai, Inc., [n.78] an influential decision of the Second Circuit addressing the extent to which copyright law protects a computer program's non-literal aspects. [n.79] Altai is well known for its "Abstraction-Filtration- Comparison" test for determining whether the non-literal aspects of two *286 programs are substantially similar. [n.80] However, the court practically dismissed Altai as unhelpful because the issue in Lotus did not involve a question of substantial similarity, but rather the more fundamental inquiry into whether or not a menu command hierarchy could be copyrighted. [n.81]

Given the apparent lack of authoritative case law, the court analyzedthe statutory source, 17 U.S.C. § 102(b), in determining whether the Lotus menu structure could be protected under copyright law. The court formulated a two- pronged analysis. First, Stahl argued that the Lotus menu command hierarchy fell neatly within § 102(b)'s prohibition against copyright protection for methods of operation. [n.82] Secondly, Stahl attempted to

reconcile the decision not to grant copyright protection with the broader aims and policies that have dominated the corpus of copyright law throughout the years. [n.83]

## VIII. Menu Command Hierarchy as a Method of Operation

United States copyright law states that "[i]n no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work." [n.84]

Based on a narrow reading of the statute, the court almost preordained its conclusion that the menu command structure is indeed a "method of operation" and hence uncopyrightable under § 102(b). Rather than considering the menu command hierarchy within the context of the Lotus computer program, the court held that the specific words contained in the Lotus menu command hierarchy constituted expression essential to the operation of the spreadsheet and therefore an unprotectable "method of operation." [n.85] Furthermore, the First Circuit insisted that its characterization of the interface accorded with the intent of *287 Congress behind § 102(b) because offering copyright protection for the menu command hierarchy would force "the user to cause the computer to perform the same operation in a different way." [n.86]

Judge Stahl recognized that the choice of specific command words and phrases necessarily entailed some degree of expressive choice. [n.87] However, he denied the relevance of these choices in determining copyrightability. [n.88] Given the view that a menu command hierarchy was a method of operation that could not be protected by copyright law, it became unnecessary to consider the expressive elements that might have contributed to its evolution. [n.89] With relatively little fanfare, Judge Stahl brushed aside the question of merging idea and expression, stating that it was irrelevant to the question at hand. [n.90]

While admitting that there did exist threshold originality, [n.91] Judge Stahl stated that to characterize the menu command hierarchy as an expression of the "'idea' of operating a computer program with commands arranged hierarchically" [n.92] would be to limit the Lotus method of operation to an abstraction. [n.93] Stahl preferred simply to conclude that the menu structure was a method of operation, therefore forfeiting its copyrightability. [n.94]

Finally, Judge Stahl attempted to draw support for his argument from the Baker decision, [n.95] arguing that its prohibition on copyright *288 protections fit neatly with the decision not to limit the "idea" of the Lotus menu structure to anything more than the lowest level of abstraction. [n.96]

In short, Judge Stahl dedicated the first section of his opinion to demonstrate that the menu command hierarchy constituted an uncopyrightable method of operation. At the same time, Stahl attempted to preclude a number of messy merger questions, admitting

that there was no merger of idea and expression, [n.97] but claiming the merger question bore no relevance to the method of operation inquiry. [n.98]

## IX. Why Methods of Operation are not Copyrightable

Despite the originality inherent in the Lotus structure, this alone was not considered determinative in deciding the issue of copyrightability. [n.99] After stating that the menu command hierarchy could not receive copyright protection despite the fact that its idea and expression were separable, [n.100] Judge Stahl attempted to reconcile this rather curious proposition with the broader goals of copyright law.

Judge Stahl offered two rationales to support his decision not to extend copyright protection to the Lotus menu command hierarchy. The first was a policy argument where he deemed the idea of an end user having to learn more than one command to perform a given function "absurd." [n.101] Furthermore, Judge Stahl argued that forcing users to perform the same operations in different ways would be inconsistent with the Congressional intent of prohibiting the copyright of "methods of operation." [n.102]

*289 Stahl can be understood to have stated that the protection of the Lotus menu structure through copyright would lead to a system whereby Lotus users are "trapped" and unable to abandon Lotus 1-2-3 in favor of other superior spreadsheets. [n.103] In Stahl's view, not only would this copyright protection inhibit interoperability, but would also appropriate some of the end user's property. Lotus users would be forced to rewrite macros to interact with the Quattro spreadsheet, despite the fact that the a macro is "clearly the user's own work product." [n.104]

Second, Stahl argued that his ruling regarding the Lotus menu command structure was in line with the Supreme Court's most recent pronouncements regarding the purposes of copyright law. Citing the Court's decision in Feist Publications, Inc. v. Rural Telephone Service Company Co., [n.105] Stahl noted that copyright law encourages the development of science and the "useful Arts" by protecting original expression and encouraging others to build upon the ideas contained in that protected expression. [n.106]

As the originality of expression in this case was not at issue, Stahl moved to the question of whether others could freely build upon the ideas inherent in the Lotus menu command hierarchy. [n.107] For Stahl, the answer to the question was facilitated by applying 17 U.S.C. § 102(b) and determining whether the subject matter in question falls within one of its prohibited categories. [n.108]

The unique question presented by the Borland appeal, as Stahl viewed it, was that although most expression need not be copied for the underlying ideas to be built upon, [n.109] the copying of the menu command *290 hierarchy, or more generally, copying a method of operation, was essential to build on the Lotus ideas. [n.110] In other words, Stahl concluded that in order to "build" upon the ideas inherent in Lotus, Borland was required to copy the Lotus expression as well.

This argument is familiar to those versed in the ways of copyright law, as it reflects a straightforward merger analysis. [n.111] Stahl, in describing what Borland was required to do in order to "build" upon the Lotus ideas, concluded that there the expression was inseparable from the idea, and hence the original expression at issue could not be afforded copyright protection.

Following the logic in Judge Stahl's argument throughout the opinion, one traces a complete reversal in direction. At the beginning of his opinion, Stahl offers two seemingly contradictory propositions: that the Lotus menu command hierarchy did not deserve copyright protection because of 17 U.S.C. § 102(b)'s prohibitions, [n.112] and that there was no merger between the idea and the expression contained in the menu structure. [n.113] However, by the time Stahl reached the end of his argument, he appeared to reverse course. While maintaining that the menu command hierarchy was undeserving of copyright protection, [n.114] he was arguing that the idea and expression had merged in support of his conclusion. [n.115]

## X. ANALYSIS

Assessing how far the Lotus copyright protection extends is difficult because the question of the copyrightability of menu command hierarchies has received little attention in legal circles. As noted above, the copyrightability of both literal and non-literal aspects of computer programs is well-settled. [n.116] Further, recognizing that there are difficulties *291 associated with analyzing these non-literal aspects, there seems to be general acceptance of the Altai approach. [n.117]

That copyright protection extends beyond the literal limits of expression is nothing new in the law. [n.118] In fact, the idea is commonsensical. Learned Hand realized decades ago that if copyright protection were limited to the exact words of an author, "a plagiarist would escape liability by immaterial violations." [n.119]

When dealing with computer programs, however, the task of determining what constitutes a literal element as distinguished from a non-literal element becomes complicated. The literal elements clearly refer to the program's source code, i.e., the instructions as programmed in any one of a number of programming languages. [n.120] The term "literal element" has also been extended to a program's object code, which comprises the binary instructions that the source code is translated into in order to directly control the computer. [n.121] Since Whelan Associates v. Jaslow Dental Laboratory, most courts have also recognized that copyright law protects some of a program's non-literal elements. [n.122] The question courts are still grappling with is exactly how far protection for non-literal elements should extend.

The Lotus cases present a novel question as the menu command hierarchy and key reader functions can be difficult to categorize as either literal or non-literal elements. Lotus never contended that Borland copied either its source or object code in creating the

Lotus Emulation *292 Interface. [n.123] On the other hand, Borland did duplicate the exact menu command structure of Lotus, as displayed by the program and used by the key reader function. [n.124]

It would seem then, at least at first glance, unfair to characterize the copied elements as "non-literal" given Borland's deliberate, literal copying of the Lotus menu command hierarchy. The court states that its task is to determine "not whether nonliteral copying occurred in some amorphous sense, but rather whether the literal copying of the Lotus menu command hierarchy constitutes copyright infringement." [n.125] However, because the copying most certainly did not cover the literal computer code, the court is left with only a non-literal analysis.

As the First Circuit deemed the non-literal analysis unsatisfactory, [n.126] it chose a third option. Rather than considering the menu command hierarchy within the realm of non-literal elements, the court decided that it constituted a "method of operation," undeserving of copyright protection under 17 U.S.C. § 102. [n.127]

Although the court downplayed the importance of this analytical move, the question of how to characterize user interfaces is one that may exercise great impact on the computer industry at large. Given that user interfaces are not part of the literal computer code, [n.128] courts must decide the same question that the First Circuit dealt with, yet did not explicitly address: whether user interfaces are properly considered part of a computer program's non-literal aspects, or whether they are a separate entity entirely and should not be afforded the benefit of copyright protection that applies to computer programs.

It should be recognized that the extension of copyright protection to computer programs represents an exception to the general rules governing copyright law. Congress amended Title 17 U.S.C. governing copyright law in the United States to define a computer program as a set *293 of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result. [n.129]

The legislative history behind the 1976 Amendments to the Copyright Act makes clear that computer programs were not to be excluded from copyright protection simply because of their functional nature. [n.130] In fact, the House Report explicitly includes computer programs as properly within the scope of copyright protection. [n.131]

In apparent contrast to the extension of copyright protection to computer programs, 17 U.S.C. § 102(b) states that "[i]n no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work." [n.132]

Without considering the definitions and legislative history, this section might induce one to assume that computer programs (since they all represent simply a method of operating computers) fall outside the scope of copyright protection. This approach certainly convinced the First Circuit.

In opting to characterize Borland's user interface as an uncopyrightable method of expression, the First Circuit attempted to draw a distinction between the § 102(b) prohibition against extending copyright protection to methods of operation and the inquiry fundamental to any question of copyright: namely, an exploration of the idea and expression dichotomy as applied to a specific fact pattern. [n.133] Stahl's intentions on *294 this point are unequivocal, as he wrote that "the fact that Lotus developers could have designed the Lotus menu command hierarchy differently is immaterial to the question of whether it is a 'method of operation.'" [n.134]

By characterizing the menu command hierarchy as a method of operation, the First Circuit deftly avoided the difficult question of whether there was a distinction between the "idea" of the menu command hierarchy, and the "expression" of it, and the relation of these notions to the § 102 exclusion of methods of operation from copyright protection, since these processes and methods are more properly the province of patent law. [n.135]

Because each computer program represents a method of operating a computer, it follows that in a given case protection for a program will necessarily extend copyright law to a method of operation. To characterize the user interface as a method of operation therefore creates the larger question of whether the extension of copyright protection to these interfaces can be made without imparting patent-like exclusive rights to the underlying system or method.

Unfortunately, at least for the supporters of Borland's position, Judge Stahl's opinion stands in direct conflict with the intent of § 102(b) as explicitly stated in the legislative history. While restructuring copyright law with the 1976 revisions, Congress made very clear its intentions in enacting 17 U.S.C. § 102(b), stating that the section "in no way enlarges or contracts the scope of copyright protection under the present law. Its purpose is to restate, in the context of the new, single federal system of copyright, that the basic dichotomy between expression and idea remains unchanged." [n.136]

Of even further interest to the present discussion is the fact that this statement of purpose regarding § 102(b) was offered in response to questions raised regarding the copyrightability of computer software. [n.137]

*295 In short, there can be no question that section § 102(b) was intended by Congress as a codification of the idea that the touchstone for copyright protection remains the idea/expression dichotomy. [n.138] Further, the scope of material excluded from the sphere of copyright under § 102(b) was to be exactly the same as the scope excluded by an idea/expression analysis. [n.139]

In Lotus, Judge Stahl began his argument stating that merger between the idea and that idea's expression had not occurred. Yet, in his subsequent § 102(b) analysis of the software interfaces, Judge Stahl extended the prohibitions of § 102(b) to include non-merged expressions. It appears that Judge Stahl viewed § 102(b) as expanding the range

of uncopyrightable matter beyond the boundaries demarcated by the merger doctrine. This belief is clearly erroneous, as it contradicts the explicit purposes of § 102(b).

In enacting § 102(b), Congress intended no more than to codify the rule as expounded by the courts prior to 1976 that the expression of an idea may receive copyright protection, but the idea itself cannot. Rather than considering § 102(b) in light of the relevant history, the First Circuit used it as a shield, protecting the court from having to undertake the unquestionably difficult task of separating those expressive elements deserving of copyright protection from the ideas and processes behind them. [n.140] Given the unique place computer programs occupy within the corpus of copyright law, there is no question that the First Circuit erred in applying its method of operation analysis to the menu command hierarchy.

*296 XI. Lotus v. Borland: Policy Considerations

The foregoing discussion provides an analysis into the judicial interpretation of 17 U.S.C. § 102(b), which prohibits the copyrighting of "any idea, procedure, process, system, [or] method of operation...." As applied to computer programs, the House Report stated that § 102(b) "intended to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law." [n.141] This passage makes clear that § 102(b)'s bar on copyright protection for a method of operation operates to ensure the distinction between patent law and copyright law.

Computer software copyrights present a unique twist, however, since copyright protection of a computer program may be applied to methods of operation without granting rights similar to those granted by patent law. As defined by Congressional statute, a computer program is "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." [n.142]

Therefore, it is axiomatic to state that computer programs are methods of operation. How then, is the tension between the copyright protection of computer programs and the § 102(b) proscription of such protection to be resolved?

The simple answer is that the nature of computer programs enables the extension of copyright protection without monopolizing the underlying process. [n.143] The extension of copyright protection therefore protects original expression contained in computer programs while simultaneously respecting the division between copyright and patent law.

Judge Stahl, in writing for the First Circuit, failed to appreciate the fundamental principles behind § 102(b) and their application to computer software copyrights. He considered the inquiry into the range of options in designing the Lotus menu command

hierarchy irrelevant, because the method of operation analysis had already transformed the interface into uncopyrightable material. [n.144]

*297 The method of analysis employed by Judge Stahl is flawed as a tool for considering the copyrightability of user interfaces for a number of reasons. First, and most importantly, the baseline dismissal of the copyrightability of user interfaces is contrary to the intent of Congress to protect computer programs. Computer programs are clearly entitled to copyright protection, subject to the restriction that their copyright not protect the processes or ideas embodied by the expression. [n.145] As a component of computer programs, the First Circuit offers no explanation why a user interface's functional aspects should remove the possibility of copyright protection, while the computer program's functional aspects do not.

Given the dismissal of Congressional intent by the Lotus court, the assertion by some commentators that the decision was driven by policy concerns argued in amicus briefs is not surprising. [n.146] Judge Stahl attempted to characterize his decision as a victory for computer software users around the country. [n.147] Further insight into this view can be seen in the concurrence by Judge Boudin. In joining Judge Stahl, Boudin writes that "it is hard to see why customers who have learned the Lotus menu and devised macros for it should remain captives of Lotus because of an investment in learning made by the users and not by Lotus." [n.148]

Boudin, in making this statement, appears to make an interesting set of economic assumptions. His argument is premised on the idea that Borland constructed a superior spreadsheet, and without allowing them to copy the Lotus menu command hierarchy, the potential benefits to the general public will be lost as a result of consumer inertia. [n.149] In essence, Boudin argues that in order to best serve the interests of the general *298 public, for whom the copyright laws exist, the field ought to be leveled to allow Borland and Lotus to compete. [n.150] Furthermore, inherent to this argument is the assumption that a computer spreadsheet's utility, or whether one is "better" than other, ought to be evaluated independently from the interface or the menu structure.

The problem with Judge Boudin's analysis is the implicit assumption that the forces normally at work in a properly functioning market are failing, because without the Lotus Emulation Interface, Borland's superior product has not garnered its deserved market share. [n.151] For Boudin, the inertia created by spreadsheet users unwilling to make the switch to a new interface represents a waste of a superior product (i.e., Borland's spreadsheet), rather than the choice of users making informed decisions about competing products. Instead of considering the menu command structure as enhancing the competitive viability of Lotus 1-2-3, Boudin regards it as a valueless, arbitrary and yet essential component of any computer spreadsheet. Therefore, Boudin views as positive the idea that he might enable Lotus users to switch to a competing product he deems superior. [n.152]

However, the fact that Lotus users, having grown accustomed to the menu structure in Lotus 1-2-3, are now generally reluctant to expend the energy or cost of learning a new

user interface is not proof that the market has failed. On the contrary, it is prima facie evidence that Lotus 1-2-3 remains competitively viable, and that the structure of the menu command hierarchy is integral to the spreadsheet's commercial success. If Borland wanted to convince Lotus users to cross the line and change spreadsheets, they should have offered a spreadsheet program with benefits commensurate with the costs that learning a different user interface would entail.

Recent developments in the computer spreadsheet market tend to allay the First Circuit's fears that copyright protection for the Lotus menu commandhierarchy would inhibit the development of other products and "lock-in" Lotus users to their current spreadsheet. In fact, *299 between 1992 and 1994, [n.153] the Lotus share of spreadsheet sales fell from 70 percent to 32 percent. [n.154] In addition, much of the decline was attributable to the growth of Microsoft's Excel spreadsheet, which incorporated none of the Lotus product. [n.155]

Not only is the economic rationale offered by Judge Boudin (and implicitly by Stahl) built upon faulty assumptions, but there are strong reasons to doubt that the decision in favor of Borland represents an unqualified victory for spreadsheet users throughout the computing world. While Judges Boudin and Stahl champion their decision as an unqualified victory for these end users, they ignore the effects of their decision. The popularity of the Lotus user interface is without question a competitive advantage of the product over other spreadsheets, which presumably offer at least comparable functionality. Because of the refusal to protect this interface through copyright law, future spreadsheet developers have little reason to create command structures that may become popular in the marketplace.

In terms of development costs, companies are better off surveying existing technology and copying the most commercially successful. Unless the premise is that the state of computer user interfaces could never possibly be improved, this effect is undesirable to the public at large. [n.156] The success of the Lotus menu command hierarchy should have alerted the First Circuit that the public would be better served if incentives to create popular user interfaces were preserved. The situation envisioned by Stahl and Boudin parallels the concerns Judge Keeton raised in Paperback. Keeton stated that by denying copyright protection to works which gain widespread popularity because of their utility, "the moment of creative triumph is also a moment of devastating financial loss--because the triumph destroys copyrightability of all expressive elements that would have been protected if only they had not *300 contributed so much to the public interest by helping to make some article useful." [n.157]

## XII. User Interfaces and Modern Copyright Law: A Superior Alternative that Already Exists

When faced with the question of user interfaces, courts properly look to the range of choices available to a programmer. Where a programmer has exercised wide discretion in the creation of his or her program, the court is more likely to find protectable expression

within that program. [n.158] In contrast, where the programmer's options are severely limited, a court will likely hold that the expression has merged with the idea of the program, and therefore that the subject of contention is not copyrightable. [n.159] In determining the extent to which external forces dictate the choice of certain interfaces, courts are able to better analyze user interfaces within the idea/expression dichotomy.

In Plains Cotton, the court recognized that a programmer could not write software for the cotton industry without using the input and output formats in question. [n.160] Therefore, to extend copyright protection to cover user interfaces within certain cotton marketing software would be to extend monopoly rights to the process of presenting this agricultural data. [n.161] Clearly this exclusive right, if to be granted at all, must be given through our patent system. Because the extension of copyright protection to these interfaces would have blurred thedistinction between patent and copyright law, the court refused to grant protection. [n.162]

*301 It should be clear that the First Circuit's insistence on using the "methods of operation" tool for deciding this case was severely flawed. However, a solid criticism of Judge Stahl's opinion should offer the proper analytical framework for assessing the copyrightability of the Lotus menu command hierarchy.

First, the menu command hierarchy is properly considered, at least at law, as a non-literal element of the Lotus 1-2-3 program. [n.163] Therefore, rather than dismissing Altai's three part test for substantial similarity, [n.164] the First Circuit should have employed the Abstraction-Filtration-Comparison test, as Judge Keeton did at the district court level, [n.165] in an effort to discern any protectable elements contained by the menu command hierarchy. In fact, Judge Keeton's analysis of the issues through application of a three part test similar to Altai presented a solid framework for analyzing the user interface under current law.

The Lotus Emulation Interface, included in the original offering of Quattro spreadsheets, constituted an arrangement of computer commands organized in a hierarchical menu structure. [n.166] Viewed against the idea/expression dichotomy, there seems little problem separating functional aspects from expressive ones. Lotus claimed no exclusive right to the "copy" or "print" commands contained in their menu trees. What Lotus did seek to protect, however, was the organization of these commands in the specific manner presented in the Lotus 1-2-3 spreadsheet. [n.167]

In offering protection to the Lotus menu tree, there exists no danger of accidentally imparting patent-like rights to the process of operating computer spreadsheets to Lotus. At every level, it is recognized that there are a myriad of possible menu trees for operating similar spreadsheets offering substantially similar functionality. [n.168] In fact, functional considerations simply do not control the choice of a menu *302 tree. [n.169] Because Borland could easily create a spreadsheet offering exactly the same functions as Lotus 1-2-3 with a completely different menu tree, it becomes difficult to sustain the argument that functional considerations drive the creation of menu trees. [n.170]

It is therefore fair to characterize the Lotus menu tree as an expressive choice, reflecting any number of concerns, from aesthetic and artistic to concerns about the user-friendliness of their program. Regardless of why the menu tree might have been constructed, however, it cannot be said that its functionality is so caught up with its expression that copyright law must deny protection to its expressive aspects. [n.171]

As presented to the spreadsheet user, then, it can be said that Judge Keeton's decision to afford the menu tree copyright protection remains faithful to the purposes of copyright law in general. Given the relevant case law and Congressional intent regarding copyright protection, a computer program's menu trees should be eligible for copyright protection to the extent that protection can be provided without granting patent-like rights to the process controlled by the interfaces, i.e., the operation of the spreadsheet. Furthermore, the above discussion should make clear that, at least in the context of spreadsheet user interfaces, there is no danger of creating a monopoly.

Given that protectable expression exists, the question then becomes whether the menu command hierarchy may somehow have lost its protectability. For the First Circuit, the interface lost its right to gain copyright protection because it could be characterized as a "method of operation." However, the relevant question should not be whether the interface is a method of operation, but rather whether the extension of copyright protection to encompass the interface would impart patent-like rights to the underlying process. [n.172] In essence, the method of operation analysis should not differ in substance from a merger inquiry.

*303 The difference in phrasing the judicial inquiry is not trivial. In asking whether or not the interfaces constituted a method of operation, the First Circuit refused to recognize that as expression the interfaces were entitled to at least a baseline assumption of copyrightability. [n.173] The choice and arrangement of commands within a menu structure could represent the expression of any number of desires, including artistic, aesthetic and efficient. [n.174] In determining whether a user interface is copyrightable, the courts must consider whether to deny copyright protection to original expression, precisely what the laws have been designed to protect. In the context of input formats, the courts have denied protection only when copyright protection could not be awarded to an interface without creating a monopoly over the process presented in the computer program. [n.175]

Although the computer industry continues to present courts with new and sometimes confusing innovations, the First Circuit was incorrect in breaking with precedent and developing a new rationale in order to deal with computer user interfaces. The idea/expression dichotomy remains the touchstone for copyright analysis. Whether applying the Altai test or terming components a method of operation, these analytical tools serve only as proxies for the judicial task at hand, i.e., determining whether the expression in question has lost its right to claim copyright protection. Affording too much protection would blur the distinction between patent and copyright law, and unfairly impart to the copyright *304 holder patent-like rights to the processes disclosed within the limits of expression. [n.176]

XIII. Fair Use

Judge Boudin came close to recognizing that a court addressing the copyrightability of a menu command hierarchy is mistaken in rejecting copyrightability outright when he discussed fair use as an alternative ground upon which the decision to allow Borland's copying to stand. [n.177] The preceding discussion has hopefully dispatched the argument that a menu command hierarchy, and user interfaces in general, may not as a matter of law receive copyright protection.

However, after inquiring into the idea/expression dichotomy within the context of the Lotus menu structure, a thorough analysis should explore the issue of whether Borland's use of the Lotus material copyrighted was privileged. Judge Boudin concluded that perhaps the most likely place to look, if one wished to decide in favor of Borland, was 17 U.S.C. § 107, the fair use exception to copyright protection. The statute provides that "fair use" of a copyrighted work does not constitute copyright infringement and lists four factors to be considered in any fair use analysis. [n.178]

As a codification of many years of common law development, the fair use exception to copyright continues to exhibit a flexibility rare even for judicial interpretation. Indeed, the legislative history behind § 107 clarifies the intent of Congress for fair use to be a malleable concept. [n.179] Nevertheless, the four statutory factors, as applied to the facts in the Lotus case, appear to weigh heavily in favor of Lotus and the protection of the menu command structure.

*305 The first factor of § 107, the purpose of the use, stands clearly in favor of Lotus. The Supreme Court has stated that "[t]he fact a publication was commercial as opposed to non-profit is a separate factor that tends to weigh against a finding of fair use." [n.180] Commercial use, like Borland's copying of the menu structure into its Quattro spreadsheet, creates a presumption against fair use, although this presumption is rebuttable. [n.181] Since Borland wished simply to draw customers away from Lotus and to its own spreadsheet, there appears little doubt that this factor would favor Lotus.

The second factor, the nature of the copyrighted work, reflects the idea that while copyright law is broad in scope, not every copyrightable work will gain equal protection. [n.182] In fact, the inquiry under this second factor appears similar to the idea/expression discussion above. [n.183] As the above discussion concludes that Lotus 1-2-3 contains separable, copyrightable elements, the second factor in a fair use analysis also weighs in favor of Lotus.

The third factor for fair use, the amount and substantiality of the portion copied, presents perhaps the most contentious issue. There is no simple equation for determining what constitutes "substantial" in the context of fair use. [n.184] However, the fact that Borland deemed the Lotus menu command hierarchy so important to its own success lends validity to the claim that the copying constituted a substantial portion of the Lotus

spreadsheet. [n.185] While we cannot proclaim that the third factor falls *306 decisively within the Lotus or Borland camp, Borland's desire to incorporate the hierarchy into its own spreadsheet is prima facie evidence of the substantiality of the menu command hierarchy in relation to the Lotus copyrighted spreadsheet.

Finally, the fourth factor laid out for consideration in § 107 is the effect of Borland's use on the value of Lotus 1-2-3. Again, the inquiry is highly speculative. However, the complex litigation that has resulted regarding the menu command hierarchy, as well as Borland's perception of the necessity for its Quattro spreadsheet to include the hierarchy, would appear to lend support to the proposition that the Lotus menu structure constituted a significant portion of its competitive viability. Judge Boudin writes that Lotus users were "locked-in" to Lotus 1-2-3, "captives of Lotus because of an investment in learning made by the users and not by Lotus." [n.186] Therefore Lotus users were unable to reap the benefits of a superior spreadsheet because that superior spreadsheet failed to contain the Lotus menu hierarchy. [n.187]

The First Circuit appears to believe that Borland's spreadsheet is superior to Lotus 1-2-3 in all ways except for the menu structure. This premise leads to the conclusion that taking away the menu structure would destroy the last marketable feature of the Lotus program. Given the court's characterization, one is left wondering why any person would even consider buying Lotus 1-2-3 if themenu structure could be obtained through the purchase of a competing spreadsheet. On the other hand, today the spreadsheet market is dominated by Microsoft's Excel, [n.188] a product without the benefit of the Lotus menu structure. This would seem to indicate that the menu command hierarchy either does not possess the commercial vitality that it once did, or that Lotus and Borland were both mistaken as to its marketing importance.

While the four factor test discloses a manner in which to approach the fair use problem, the most important lesson to take from § 107 is the fact that fair use is intended to provide a flexible framework to consider broad, equitable concerns. [n.189] The foregoing discussion should *307 make clear that the spreadsheet market in general, and end-users specifically, are best served by the protection of these types of user interfaces. In addition, employing the four factor fair use test leads to the conclusion that an exception to the normal copyright provisions should not be granted for Borland.

## XIV. Conclusion

In the past decade, the explosion of copyright infringement litigation in the computer industry has touched almost every facet of the relevant technology. With respect to user interfaces, there is near universal recognition that the choice and presentation of user interfaces contain original expression, and therefore these interfaces are entitled at least to an assumption that they are copyrightable. However, the merger doctrine, as developed within the body of copyright law, prohibits the extension of copyright protection where the protection would encompass both the expression and the idea underlying it.

The debate over user interfaces, therefore, properly involves a determination whether extending protection to the interface would grant the copyright holder an exclusive right to the idea embodied in his or her software. In cases where the interface is mandated by external conditions, rather than chosen from a large range of viable alternatives, copyrightability is properly denied. On the other hand, where the user interface is the product of decisions by the programmer drawing on a wide variety of possibilities, and the exclusive right to that interface in no way affects the ability of competitors to duplicate the process through different expression, copyright protection is generally afforded.

In Lotus Development Corp. v. Borland International, Inc., [n.190] however, the First Circuit avoided this scholarly debate by citing a section of the United States Code for which their opinion evinces little understanding. Dismissing the copyrightability of the Lotus menu command hierarchy because it is a "method of operation" represents a step back in the evolution of copyright law as applied to computer programs.

The Supreme Court's recent decision [n.191] has disappointed the users and producers within the computer software industry. Rather than either throwing its weight behind Judge Stahl or reversing what amounted to *308 nothing more than a poorly devised judicial opinion, the Court failed to clarify the boundaries of copyright law. With the deadlocked decision, the Court did not offer a statement to help define the limits of copyright law of software interfaces, but instead cast further confusion on an already clouded question. The Court missed a perfect opportunity to recoup at least some of the lag time often seen between the advent of cutting edge technology and the extension of legal protection to these new developments.

The First Circuit's opinion lives on, but questions remain with regard to the extent to which copyright law will protect user interfaces. This article has described the tools for examining computer copyright protection in this context. These tools are not to be found in obfuscated readings of the copyright statute, but rather in the very teachings that have supported copyright jurisprudence since its inception.

The First Circuit's opinion incorrectly interprets the proper scope of copyright protection in this field, and also promises to stunt rather than expand the range of alternatives available to software users if it is adopted by additional jurisdictions.

The Supreme Court, in affirming the First Circuit by default, has neither rendered an opinion on the subject nor offered guidance to Circuits unwilling to summarily adopt Judge Stahl's reasoning. However, until the Supreme Court finally fulfills its obligations by offering an opinion on the subject, the debate that Judge Hand referred to decades ago promises to rage on.

[n.a1]. William F. Porter is a second year law student at Duke University School of Law. The author would like to thank Marc A. Ehrlich for his helpful comments and encouragement.

[n.1]. Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930).

[n.2]. Lotus Dev. Corp. v. Borland Int'l, Inc., No. 94-2003, 1996 U.S. LEXIS 470, at * 1 (Jan. 16, 1996). Justice Stevens took no part in the consideration of the case, for reasons that have not been disclosed, to date, allowing for the 4-4 decision. See also Linda Greenhouse, Supreme Court Deadlocks in Key Case on Software, N. Y. Times, Jan. 16, 1996, at D2. ("A closely watched Supreme Court case that was expected to define the boundaries of copyright protection for computer software fizzled today as the eight justices who heard the case last week deadlocked four to four.").

[n.3]. 49 F.3d 807, 810, 34 U.S.P.Q.2d (BNA) 1014, 1016 (1st Cir. 1995).

[n.4]. Id. Borland admitted that in copying the menu command structure it desired to "obtain the benefit of its functionality." Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 209 (D.Mass. 1992). Presumably the marketplace success of Lotus rendered it profitable for Borland to present a spreadsheet with which those familiar with Lotus 1-2-3 could easily interact.

[n.5]. 49 F.3d at 810, 34 U.S.P.Q.2d (BNA) at 1016.

[n.6]. Lotus Dev. Corp. v. Paperback Software Int'l, 740 F.Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (D.Mass. 1990).

[n.7]. Id.

[n.8]. Lotus Dev. Corp. v. Borland Int'l, Inc., 788 F.Supp. 78, 80, 99, 22 U.S.P.Q.2d (BNA) 1641, 1642, 1656 (D.Mass. 1992).

[n.9]. Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 205 (D.Mass. 1992).

[n.10]. Id. at 216-20, 223. Courts in general have had a difficult time in dealing with the question of computer user interfaces in the context of copyright law. See infra notes 43-67, and accompanying text.

[n.11]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 811, 34 U.S.P.Q.2d (BNA) 1014, 1017 (1st Cir. 1995).

[n.12]. Id.

[n.13]. Id. The term "macro" refers to a feature by which a user may substitute a relatively short keystroke for a longer sequence. Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 206 (D.Mass. 1992).

[n.14]. 49 F.3d at 812, 34 U.S.P.Q.2d (BNA) at 1018.

[n.15]. Id.

[n.16]. Id. See Lotus Dev. Corp. v. Borland Int'l, Inc., 831 F.Supp. 202, 30 U.S.P.Q.2d (BNA) 1081 (D.Mass. 1993) for the opinion on the menu command structure, and Lotus Dev. Corp. v. Borland Int'l, Inc., 831 F.Supp. 223 (D.Mass. 1993), for discussion of the key reader function.

[n.17]. See, e.g., Mazer v. Stein, 347 U.S. 201, 219, 100 U.S.P.Q. (BNA) 325, 333 (1954), stating that "the economic philosophy behind the clause empowering Congress to grant patents and copyrights is the conviction that encouragement of individual effort by personal gain is the best way to advance public welfare through the talents of authors and inventors in 'Science and useful Arts.'"

[n.18]. Id.

[n.19]. Lotus Dev. Corp. v. Paperback Software Int'l, 740 F.Supp. 37, 52, 15 U.S.P.Q.2d (BNA) 1577, 1587 (D.Mass. 1990). The Supreme Court addressed this idea recently, in the context of copyrighting telephone directories. Justice O'Connor rejected the "sweat of the brow" analysis of copyright protection, stating that "the 1976 revisions to the Copyright Act leave no doubt that originality, not 'sweat of the brow,' is the touchstone of copyright protection." Feist Publications, Inc. v. Rural Tel. Serv. Co., 499 U.S. 340, 359-60, 18 U.S.P.Q.2d (BNA) 1275, 1283 (1991). While this exposition does not eliminate the idea of rewarding the author through copyright law, it emphasizes the notion that a copyright is awarded for originality, regardless of the effort expended by an author.

[n.20]. 101 U.S. 99 (1879).

[n.21]. Id. at 99-100.

[n.22]. Id. at 100.

[n.23]. Id.

[n.24]. Id. at 107.

[n.25]. Id. at 105

[n.26]. 347 U.S. 201, 100 U.S.P.Q. (BNA) 325 (1954).

[n.27]. Id. at 215, 217, 100 U.S.P.Q. (BNA) at 332, 333.

[n.28]. Id. at 218, 100 U.S.P.Q. (BNA) at 333. The Court stated "[w]e find nothing in the copyright statute to support the argument that the intended use or use in industry of an article eligible for copyright bars or invalidates its registration. We do not read such a limitation into the copyright law." Id.

[n.29]. Id. at 217, 100 U.S.P.Q. (BNA) at 333 (citing Jewelers Circular Pub. Co. v. Keystone Pub. Co., 281 F. 83, 94 (2d Cir. 1922)).

[n.30]. Morrissey v. Proctor & Gamble Co., 379 F.2d 675, 154 U.S.P.Q. (BNA) 193 (1st Cir. 1967).

[n.31]. See Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930).

[n.32]. 379 F.2d at 678, 154 U.S.P.Q. (BNA) at 195 (quoting Sampson & Murdock Co. v. Seaver-Redford Co., 140 F. 539, 541 (1st Cir. 1905)).

[n.33]. The court stated that "[i]n such circumstances it does not seem accurate to say that any particular form of expression comes from the subject matter. However, it is necessary

to say that the subject matter would be appropriated by permitting the copyrighting of its expression. We cannot recognize copyright as a game of chess in which the public can be checkmated." 379 F.2d at 678-79, 154 U.S.P.Q. (BNA) at 195.

[n.34]. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 219 U.S.P.Q. (BNA) 113 (3rd Cir. 1983). See also Lotus Dev. Corp. v. Paperback Software Int'l, 740 F.Supp. 37, 67, 15 U.S.P.Q.2d (BNA) 1577, 1599 (D.Mass. 1990).

[n.35]. 1 M. Nimmer & D. Nimmer, Nimmer on Copyright § 2.04[C] (1993).

[n.36]. Id. Congress amended the Copyright statute to include a definition of computer programs, in order to end the ongoing debate and place programs within the protective cloak of copyright law.

[n.37]. Feist Publications, Inc. v. Rural Tel. Serv. Co., 499 U.S. 340, 361, 18 U.S.P.Q.2d (BNA) 1275, 1284 (1991).

[n.38]. Copying can be proven in either of two ways: either through direct evidence of copying, or by evidence that the alleged infringer "had access to the copyrighted work and that the offending and copyrighted works are so similar that the court may infer that there was factual copying." Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 813, 34 U.S.P.Q.2d (BNA) 1014, 1019 (1st Cir. 1995) (citing Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335, 1340, 31 U.S.P.Q.2d (BNA) 1641, 1644 (5th Cir. 1994)).

[n.39]. Id. The concept of "substantially similar" is important for copyright questions in general, but unimportant in the context of this decision, as Borland admitted that it had copied Lotus's menu command structure. See Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 208 (D.Mass. 1992).

[n.40]. Id.

[n.41]. The literal aspects of a computer program are the source and object codes, while the non-literal aspects extend to a program's overall structure and other intangible qualities. See Whelan Assocs. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1233-34, 230 U.S.P.Q. (BNA) 481, 488-89 (3rd Cir. 1986). The extension is consistent with copyright law in regard to literary and artistic expressions, as an author's copyright may be

infringed through plot similarities or other devices, although the literal expressions are not exact copies. Id.

[n.42]. See, e.g., Computer Assocs., Int'l, Inc. v. Altai, Inc., 982 F.2d 693 (2d Cir. 1992); Whelan Assocs. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 230 U.S.P.Q. (BNA) 481 (3rd Cir. 1986).

[n.43]. 740 F.Supp. 37, 15 U.S.P.Q.2d (BNA) 1577 (D.Mass. 1990).

[n.44]. 982 F.2d 693 (2d Cir. 1992). In Step 1, the computer program is analyzed in terms of increasing levels of abstraction, beginning with the source code and ultimately reaching the level of describing the program's ultimate purpose. Id. at 707. Next, these levels of abstraction are "filtered" in an exercise to determine which elements are "idea" and which were incidental to that idea. Id. Finally, the protectable elements that have been filtered out by the court are compared to the allegedly infringing expression in order to assess substantial similarity. Id. at 710. The discussion by Judge Keeton in Paperback is roughly equivalent.

[n.45]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 813, 34 U.S.P.Q.2d (BNA) 1014, 1019 (1st Cir. 1995).

[n.46]. Stahl stated that "[w]hile some other courts appear to have touched on [whether a computer menu command hierarchy is copyrightable] briefly in dicta, we know of no cases that deal with directly with the copyrightability of a menu command hierarchy standing on its own (i.e., without other elements of the user interface, such as screen displays, in issue)." Id. (citations omitted).

[n.47]. 462 F.Supp. 1003, 199 U.S.P.Q. (BNA) 537 (N.D. Tex. 1978).

[n.48]. Id. at 1011, 199 U.S.P.Q. (BNA) at 544-45.

[n.49]. Judge Higginbotham stated that "[a]t first glance these input formats are simply devices for the assistance of the user to facilitate his task-forms. On reflection, however, one must conclude that they indeed express ideas." Id. at 1011, 199 U.S.P.Q. (BNA) at 545.

[n.50]. "It follows that the formats are copyrightable if the ideas they express are separable from their expression." Id. at 1012, 199 U.S.P.Q. (BNA) at 545.

[n.51]. Id. at 1013, 199 U.S.P.Q. (BNA) at 546.

[n.52]. 807 F.2d 1256, 1 U.S.P.Q.2d (BNA) 1635 (5th Cir. 1987).

[n.53]. Id. at 1262, 1 U.S.P.Q.2d (BNA) at 1640. The persuasive authority of Plains Cotton, however, has to some extent eroded. The Fifth Circuit explicitly recognized that the case could not be extended as authority for the proposition that non-literal expression within computer programs is unprotectable, as it joined the emerging consensus of courts recognizing the extension of copyright law to encompass non-literal expression. Kepner-Tregoe, Inc. v. Leadership Software, Inc., 12 F.3d 527, 536, 29 U.S.P.Q.2d (BNA) 1747, 1752 (5th Cir. 1994).

[n.54]. See Ashton-Tate Corp. v. Ross, 916 F.2d 516, 16 U.S.P.Q.2d (BNA) 1541 (9th Cir. 1990), aff'g Ashton-Tate v. Ross, 728 F.Supp. 597, 12 U.S.P.Q.2d (BNA) 1734 (N.D. Cal. 1989) (commands list only idea and not protected under federal law).

[n.55]. The fear, at least from the view of the courts, is that extending copyright protection in situations where the choice of interfaces are dictated by externalities would grant monopoly rights to the process described. See Manufacturers Technologies, Inc. v. Cams, Inc., 706 F.Supp. 984, 995, 10 U.S.P.Q.2d (BNA) 1321, 1329 (D. Conn. 1989) ("To give the plaintiff copyright protection for this aspect of its screen displays, would come dangerously close to allowing it to monopolize a significant portion of the easy-to-use internal navigation conventions for computers.... For the same reasons, the idea of using a menu driven approach to effect external navigation between various program components and specific screen displays is likewise not copyrightable." (citations omitted)).

[n.56]. This is the same district court that decided Ashton-Tate v. Ross, 728 F.Supp. 597, 12 U.S.P.Q.2d (BNA) 1734 (N.D. Cal. 1989), aff'd 916 F.2d 516, 16 U.S.P.Q.2d (BNA) 1541 (9th Cir. 1990).

[n.57]. See Broderbund Software, Inc. v. Unison World, Inc., 648 F.Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1986) where the applications in question were, respectively, "Print Shop" and "Printmaster."

[n.58]. Id. at 1132, 231 U.S.P.Q. (BNA) at 702.

[n.59]. The court stated that "[t]here is no danger in the present case that affording copyright protection to the 'instructions' of 'Print Shop' will amount to awarding a plaintiff a monopoly over the idea of a menu-driven program that prints greeting cards, banners, signs, and posters." Id. at 1134, 231 U.S.P.Q. (BNA) at 704.

[n.60]. Id.

[n.61]. 26 F.3d 1335, 31 U.S.P.Q.2d (BNA) 1641 (5th Cir. 1994).

[n.62]. Id. at 1342, 31 U.S.P.Q.2d (BNA) at 1645.

[n.63]. Id. at 1346, 31 U.S.P.Q.2d (BNA) at 1649.

[n.64]. Id. at 1351, 31 U.S.P.Q.2d (BNA) at 1653.

[n.65]. 994 F.2d 1476, 26 U.S.P.Q.2d (BNA) 1828 (10th Cir. 1993).

[n.66]. The court addressed the question in a much quoted footnote, stating that "the record showed that the keying procedure reflected at least a minimal degree of creativity. Further, NESS has not pointed to substantial evidence that this procedure was such a common practice, or that it was dictated by efficiency considerations, so that it should have been filtered out of the analysis." Id. at 1495 n.23, 26 U.S.P.Q.2d (BNA) at 1842 n.23.

[n.67]. Judge Stahl, with apparent incredulity, stated that "we fail to see how 'a student selecting a response by pressing the 1, 2, or 3 keys' can be anything but an unprotectable method of operation." Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 819, 34 U.S.P.Q.2d (BNA) 1014, 1024 (1st Cir. 1995) (citations omitted).

[n.68]. In Autoskill, the fact that the interfaces were not "dictated by efficiency considerations" was important. Autoskill, Inc. v. National Educ. Support Sys., Inc., 994 F.2d 1476, 1495 n.23, 26 U.S.P.Q.2d (BNA) 1828, 1842 n.23 (10th Cir. 1993). Similarly, Broderbund mentions "utilitarian and mechanical" considerations. Broderbund Software, Inc. v. Unison World, Inc., 648 F.Supp. 1127, 1134, 231 U.S.P.Q. (BNA) 700, 704 (N.D.

Cal. 1986). See also Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335, 1346- 47, 31 U.S.P.Q.2d (BNA) 1641, 1649 (5th Cir. 1994).

[n.69]. See Plains Cotton Coop. v. Goodpasture Computer Serv., 807 F.2d 1256, 1262, 1 U.S.P.Q.2d (BNA) 1635, 1640 (5th Cir. 1987) ("The record supports the inference that market factors play a significant role in determining the sequence and organization of cotton marketing software ...."). See also Ashton-Tate v. Ross, 916 F.2d 516, 16 U.S.P.Q.2d (BNA) 1541 (9th Cir. 1990).

[n.70]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 812, 34 U.S.P.Q.2d (BNA) 1014, 1018 (1st Cir. 1995).

[n.71]. The court noted that "this appeal is in a very different posture from most copyright-infringement cases, for copyright infringement generally turns on whether the defendant has copied protected matter as a factual matter." Id. at 813, 34 U.S.P.Q.2d (BNA) at 1019.

[n.72]. See supra notes 47-69.

[n.73]. 49 F.3d at 813, 34 U.S.P.Q.2d (BNA) at 1019.

[n.74]. 101 U.S. 99 (1879).

[n.75]. Id. at 107.

[n.76]. 49 F.3d at 814, 34 U.S.P.Q.2d (BNA) at 1019-20.

[n.77]. Id.

[n.78]. 982 F.2d 693 (2d Cir. 1992).

[n.79]. Id. at 706. A program's literal aspects would be its source code (i.e., the computer program written in BASIC, FORTRAN, or some other programming language) or object code (the program translated into binary code understandable to the computer). In contrast, it's non-literal aspects extend to overall structure and other intangible qualities.

See Whelan Assocs. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1233-34, 230 U.S.P.Q. (BNA) 481, 488 (3rd Cir. 1986).

[n.80]. Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 706 (2d Cir. 1992). See also supra note 44.

[n.81]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 815, 34 U.S.P.Q.2d (BNA) 1014, 1020 (1st Cir. 1995). The court stated that "[w]hile the Altai test may provide a useful framework for assessing the alleged nonliteral copying of computer code, we find it to be of little help in assessing whether the literal copying of a menu command hierarchy constitutes copyright infringement."

[n.82]. Id.

[n.83]. Id. at 818, 34 U.S.P.Q.2d (BNA) at 1022.

[n.84]. 17 U.S.C. § 102(b) (1976).

[n.85]. 49 F.3d at 816, 34 U.S.P.Q.2d (BNA) at 1021.

[n.86]. Id. at 818, 34 U.S.P.Q.2d (BNA) at 1023.

[n.87]. Id.

[n.88]. The court stated that the "'expressive' choices of what to name the command terms and how to arrange them do not magically change the uncopyrightable menu command hierarchy into copyrightable subject matter." Id.

[n.89]. Id. This is an odd statement, given the fact that copyright protection extends to computer source and object code. Computer programmers wishing to create a new spreadsheet may copy the functions present in Lotus 1-2-3 (e.g., print, copy, etc.), but the copyright law prevents them from achieving this result by copying the Lotus source and object code.

[n.90]. Id.

[n.91]. Id.

[n.92]. Id. at 816, 34 U.S.P.Q.2d (BNA) at 1021.

[n.93]. Id. at 816, 34 U.S.P.Q.2d (BNA) at 1022.

[n.94]. The obvious implication in Stahl's analysis is that, although the menu command hierarchy was the product of some expressive choices, any choice made by the Lotus programmers would have been unprotectable under copyright law, since all possibilities would similarly be characterized as "methods of operation."

[n.95]. It is interesting that Judge Stahl looked to Baker v. Selden for support since he had distinguished the case from the facts at hand not more than a few paragraphs earlier. See Id. at 814, 34 U.S.P.Q.2d (BNA) at 1019.

[n.96]. Id. at 817, 34 U.S.P.Q.2d (BNA) at 1023. Viewing the levels of abstraction, the menu command hierarchy would, at its lowest level of abstraction, be the exact command structure embodied in the Lotus software. At the district court level, Judge Keeton chose a slightly more abstract level, arguing that the proper lens to view the menu structure would be a user interface with hierarchically arranged command terms. See Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 216 (D.Mass. 1992). It should be noted, however, that Judge Stahl at no time undertook an abstractions analysis.

[n.97]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 816, 34 U.S.P.Q.2d (BNA) 1014, 1022 (1st Cir. 1995).

[n.98]. Id.

[n.99]. Id. at 818, 34 U.S.P.Q.2d (BNA) at 1023.

[n.100]. See supra note 88.

[n.101]. 49 F.3d at 818, 34 U.S.P.Q.2d (BNA) at 1023.

[n.102]. Id.

[n.103]. Stahl wrote "if a user uses several different programs, he or she must learn how to perform the same operation in a different way for each program used. For example, if the user wanted the computer to print material, then the user would have to learn not just one method of operating the computer such that it prints, but many different methods. We find this absurd." Id.

[n.104]. Id.

[n.105]. 499 U.S. 340 (1991).

[n.106]. 49 F.3d at 818, 34 U.S.P.Q.2d (BNA) at 1023.

[n.107]. Id.

[n.108]. Id. Stahl wrote that "while original expression is necessary for copyright protection, we do not think that it is alone sufficient. Courts must still inquire whether original expression falls within one of the categories foreclosed from copyright protection by § 102(b), such as being a 'method of operation.'" Id.

[n.109]. Id. Stahl wrote that "in most contexts, there is no need to 'build' upon other people's expression, for the ideas conveyed by that expression can be conveyed by someone else without copying the first author's expression. In the context of methods of operation, however, 'building' requires the use of the precise method of operation already employed; otherwise, 'building' would require dismantling, too." Id.

[n.110]. Id.

[n.111]. Judge Stahl cites Morrissey, explicitly mentioning the merger concept. Id.

[n.112]. Id. at 815, 34 U.S.P.Q.2d (BNA) at 1021.

[n.113]. Id. at 816, 34 U.S.P.Q.2d (BNA) at 1022.

[n.114]. Id. at 818, 34 U.S.P.Q.2d (BNA) at 1023.

[n.115]. Id.

[n.116]. See supra notes 39-42.

[n.117]. See Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 706 (2d Cir. 1992).

[n.118]. See Id. at 701. (list of cases supporting the statement that "copyright protection extends beyond a literary work's strictly textual form to its non-literal components.")

[n.119]. Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930).

[n.120]. Whelan Assocs. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1237, 230 U.S.P.Q. (BNA) 481, 491-92 (3rd Cir. 1986). See also SAS Inst., Inc. v. S & H Computer Sys., Inc., 605 F.Supp. 816, 830, 225 U.S.P.Q. (BNA) 916, 926 (M.D. Tenn. 1985).

[n.121]. Williams Elecs., Inc. v. Artic Int'l, Inc., 685 F.2d 870, 876- 77, 215 U.S.P.Q. (BNA) 405, 410 (3rd Cir. 1982). See also Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1249, 219 U.S.P.Q. (BNA) 113, 121 (3rd Cir. 1983), concluding that "a computer program, whether in object code or source code, is a 'literary work' and is protected from unauthorized copying, whether from its object or source code version."

[n.122]. Whelan Assocs. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1239, 230 U.S.P.Q. (BNA) 481, 492-93 (3rd Cir. 1986). The court intended the term "non-literal" to cover a computer program's structure, sequence, and organization. See also SAS Inst. Inc. v. S & H Computer Sys., Inc., 605 F.Supp. 816, 225 U.S.P.Q. (BNA) 916 (M.D. Tenn. 1985).

[n.123]. Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 208 (D.Mass. 1992).

[n.124]. Id.

[n.125]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 816, 34 U.S.P.Q.2d (BNA) 1014, 1020 (1st Cir. 1995).

[n.126]. Judge Stahl wrote that "[i]n the instant appeal, we are not confronted with alleged nonliteral copying of computer code. Rather, we are faced with Borland's deliberate, literal copying of the Lotus menu command hierarchy." Id. at 814, 34 U.S.P.Q.2d (BNA) at 1020.

[n.127]. Id. at 815, 34 U.S.P.Q.2d (BNA) at 1021.

[n.128]. Borland did not need to copy any of the Lotus source or object code in order to create the emulation interface.

[n.129]. 17 U.S.C. § 101 (1976).

[n.130]. The House Report states "[s]ome concern has been expressed lest copyright in computer programs should extend protection to the methodology adopted by the programmer, rather than merely to the 'writing' expressing his ideas. Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law." Notes of Committee on the Judiciary, H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.131]. The Report states "[t]he term 'literary works' does not connote any criterion of literary merit of qualitative value: it includes catalogs, directories, and similar factual, reference, or instructional works and compilations of data. It also includes computer data bases, and computer programs to the extent that they incorporate authorship in the programmer's expression of original ideas, as distinguished from the ideas themselves." H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.132]. 17 U.S.C. § 102(b) (1976) (emphasis added).

[n.133]. See Baker v. Selden, 101 U.S. 99, 101 (1879). Justice Bradley stated that the key question was "whether, in obtaining the copyright of his books, he secured the exclusive right to the use of the system or method of book-keeping which the said books are intended to illustrate and explain." Id.

[n.134]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 816, 34 U.S.P.Q.2d (BNA) 1014, 1022 (1st Cir. 1995). Stahl also stated that despite "the district court's finding that the Lotus developers made some expressive choices in choosing and arranging the Lotus command terms, we nonetheless hold that that expression is not copyrightable because it is a part of Lotus 1-2- 3's 'method of operation.'" Id.

[n.135]. "It is a fundamental principle of copyright law that a copyright does not protect an idea, but only the expression of the idea." Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 703 (2d Cir. 1992).

[n.136]. H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.137]. Just before the section quoted, the report states that "[s]ome concern has been expressed lest copyright in computer programs should extend protection to the methodology or processes adopted by the programmer, rather than merely to the 'writing' expressing his ideas. Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law. Section 102(b) in no way enlarges...." Id.

[n.138]. See Data East USA, Inc. v. Epyx, Inc., 862 F.2d 204, 207, 9 U.S.P.Q.2d (BNA) 1322, 1325 (9th Cir. 1988).

[n.139]. See also Toro Co. v. R & R Prods. Co., 787 F.2d 1208, 1211, 229 U.S.P.Q. (BNA) 282, 284 (8th Cir. 1986).

[n.140]. The Tenth Circuit was presented with a similar question. In Autoskill, Inc. v. National Educ. Support Sys., Inc., the court reviewed a district court decision which included the conclusion that a keying procedure (where the programmer required a user to press "1," "2," or "3" in response to certain questions) could be copyrighted. Defendant in the action made the same argument as Borland in this case, stating that the keying procedure constituted a "method of operation." However, the Tenth Circuit recognized that to resolve the issue, "we must go beyond the literal language of the statute and apply the idea/expression distinction." Autoskill, Inc. v. National Educ. Support Sys., Inc., 994 F.2d 1476, 1495 n.23, 26 U.S.P.Q.2d (BNA) 1828, 1842 n.23 (10th Cir. 1993).

[n.141]. H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.142]. 17 U.S.C. § 101 (1980).

[n.143]. This is true since a programmer can write a variety of different computer programs in response to a single problem, all of which contain similar functionality but express this functionality in different ways.

[n.144]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 816, 34 U.S.P.Q.2d (BNA) 1014, 1022 (1st Cir. 1995). Stahl stated that "[t]he 'expressive' choices of what to name the command terms and how to arrange them do not magically change the uncopyrightable menu command hierarchy into copyrightable subject matter." Id.

[n.145]. See H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.146]. Two commentators stated that "[i]t is evident that the court was heavily influenced by arguments made by Borland's amici. Borland's amici had filed briefs based on four concepts which, if taken to their logical extreme, would deprive software of effective copyright protection." Anthony L. Clapes and Jennifer M. Daniels, Lotus v. Borland: Nightmare on Milk Street?, Computer Law., May 1995, at 1, 19.

[n.147]. Stahl wrote that if Lotus were to prevail "the user would have to rewrite his or her macro using that other program's menu command hierarchy. This is despite the fact that the macro is clearly the user's own work product. We think that forcing the user to cause the computer to perform the same operation in a different way ignores Congress's direction in § 102(b) that 'methods of operation' are not copyrightable." At another point, he refers to this possibility as "absurd." Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 818, 34 U.S.P.Q.2d (BNA) 1014, 1023 (1st Cir. 1995).

[n.148]. Id. at 821, 34 U.S.P.Q.2d (BNA) at 1026 (Boudin, J., concurring).

[n.149]. Judge Boudin stated that in creating the Lotus Emulation Interface Borland is "merely trying to give former Lotus users an option to exploit their own prior investment in learning or in macros." Id.

[n.150]. Boudin wrote "Lotus has already reaped a substantial reward for being first; assuming that the Borland program is now better, good reasons exist for freeing it to attract old Lotus customers: to enable the old customers to take advantage of a new advance, and to reward Borland in turn for making a better product. If Borland has not made a better product, then customers will remain with Lotus anyway." Id.

[n.151]. Boudin wrote that "if a better spreadsheet comes along, it is hard to see why customers who have learned the Lotus menu and devised macros for it should remain captives of Lotus because of an investment in learning made by the users and not by Lotus." Id.

[n.152]. This ends-based view allows Boudin to conclude, rather shockingly, that "for me the question is not whether Borland should prevail but on what basis." Id.

[n.153]. During these years the Lotus menu command hierarchy enjoyed copyright protection via Judge Keeton's rulings at the district court level.

[n.154]. Anthony L. Clapes and Jennifer M. Daniels, Lotus v. Borland: Nightmare on Milk Street?, Computer Law., May 1995, at 1, 20.

[n.155]. Id.

[n.156]. In other words, if one assumes that we have reached the pinnacle in terms of menu command hierarchy development, then there is no reason to worry about the decision of the First Circuit in terms of potential devastation to incentives to innovate, since our assumption has obviated the need to innovate. For obvious reasons, however, one ought to be careful before making such a statement regarding future developments in the personal computing environment.

[n.157]. Lotus Dev. Corp. v. Paperback Software Int'l., 740 F.Supp. 37, 57, 15 U.S.P.Q.2d (BNA) 1577, 1591 (D.Mass. 1990). Keeton continued "[r]ather than promoting and encouraging both the development and disclosure of the best, such a rule would offer incentives to market only the second, or third, or tenth best, and hold back the best for fear that it is too good for copyrightability. Copyrightability is not a synonym for imperfection." Id. at 58-59, 15 U.S.P.Q.2d (BNA) at 1591.

[n.158]. See Autoskill, Inc. v. National Educ. Support Sys., Inc., 994 F.2d 1476, 1495 n.23, 26 U.S.P.Q.2d (BNA) 1828, 1842 n.23 (10th Cir. 1993).

[n.159]. See, e.g., Ashton-Tate v. Ross, 916 F.2d 516, 16 U.S.P.Q.2d (BNA) 1541 (9th Cir. 1990). See also Plains Cotton Coop. Ass'n v. Goodpasture Computer Serv., Inc., 807 F.2d 1256, 1 U.S.P.Q.2d (BNA) 1635 (5th Cir. 1987).

[n.160]. 807 F.2d at 1262, 1 U.S.P.Q.2d (BNA) at 1640 (5th Cir. 1987).

[n.161]. Id.

[n.162]. Id. The court wrote that the "record supports the inference that market factors play a significant role in determining the sequence and organization of cotton marketing software, and we decline to hold that those patterns cannot constitute 'ideas' in a computer context." Id.

[n.163]. The literal aspects of the program are source and object code, and here Lotus does not allege that Borland copied its source code in creating its Lotus Emulation Interface.

[n.164]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 815, 34 U.S.P.Q.2d (BNA) 1014, 1020 (1st Cir. 1995).

[n.165]. Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 216- 19 (D.Mass. 1992).

[n.166]. Lotus Dev. Corp. v. Borland Int'l, Inc., 831 F.Supp. 202, 211, 30 U.S.P.Q.2d (BNA) 1081, 1085 (D.Mass 1993).

[n.167]. Id. Keeton wrote that "[e]ven though the executable operations are not copyrightable, the menu tree is copyrightable because the (hierarchical) arrangement of the definition and identification of executable operations contains expression." Id.

[n.168]. Id. at 214, 30 U.S.P.Q.2d (BNA) at 1088.

[n.169]. Id. Judge Keeton explained that when differences in menu trees appear between competing spreadsheets, they "are due to different choices about how to express to the user the available user cho ices about all the particular operations that the program can perform." Id. at 215, 30 U.S.P.Q.2d (BNA) at 1089.

[n.170]. One must remember that the lawsuit began when Borland constructed a Lotus Emulation Interface, which allowed users to operate the Quattro spreadsheet using either the Lotus or Borland menu command hierarchy.

[n.171]. Judge Keeton evaluated this question, concluding that "the order in which commands are listed in a menu has very limited functional value." Lotus Dev. Corp. v. Borland Int'l, Inc., 799 F.Supp. 203, 218 (D.Mass. 1992). This conclusion was later accepted by Judge Stahl. See Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 816, 34 U.S.P.Q.2d (BNA) 1014, 1021- 22 (1st Cir. 1995).

[n.172]. One court phrased the question as "whether the utilitarian function of the input formats ... outweigh their expressive purpose so as to preclude copyright protection." Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335, 1346, 31 U.S.P.Q.2d 1641, 1649 (5th Cir. 1994).

[n.173]. Judge Stahl, in considering the applicability of the Altai test, wrote that Altai "may actually be misleading because, in instructing courts to abstract the various levels, it seems to encourage them to find a base level that includes copyrightable subject matter that, if literally copied, would make the copier liable for copyright infringement. While that base (or literal) level would not be at issue in a nonliteral-copying case like Altai, it is precisely what is at issue inthis appeal." Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 815, 34 U.S.P.Q.2d (BNA) 1014, 1020 (1st Cir. 1995) (footnote omitted).

[n.174]. See Broderbund Software, Inc. v. Unison World, Inc., 648 F.Supp. 1127, 231 U.S.P.Q. (BNA) 700 (N.D. Cal. 1986); Autoskill, Inc. v. National Educ. Support Sys., Inc., 994 F.2d 1476, 1495 n.23, 26 U.S.P.Q.2d (BNA) 1828, 1848 n.23 (10th Cir. 1993); Engineering Dynamics, Inc. v. Structural Software, Inc., 26 F.3d 1335, 1346-47, 31 U.S.P.Q.2d (BNA) 1641, 1649 (5th Cir. 1994).

[n.175]. See Synercom Technology, Inc. v. University Computing Co., 462 F.Supp. 1003, 1012, 199 U.S.P.Q. (BNA) 537, 545 (N.D. Tex. 1978). The court stated that "it follows that the formats are copyrightable if the ideas they express are separable from their expression." Id.

[n.176]. See H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.177]. See Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 821, 34 U.S.P.Q.2d (BNA) 1014, 1026 (1st Cir. 1995) (Boudin, J., concurring).

[n.178]. The four factors are: 1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes; 2) the nature of the copyrighted work; 3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and 4) the effect of the use upon the potential market for or the value of the copyrighted work. 17 U.S.C. § 107 (1995).

[n.179]. "Although the courts have considered and ruled upon the fair use doctrine over and over again, no real definition of the concept has ever emerged. Indeed, since the doctrine is an equitable rule of reason, no generally applicable definition is possible, and each case raising the question must be decided on its own facts. On the other hand, the courts have evolved a set of criteria which, though in no case definitive or determinative, provide some gauge for balancing the equities." H.R. 1476, 94th Cong., 2d Sess. (1976).

[n.180]. Harper & Row, Publishers, Inc. v. Nation Enters., 471 U.S. 539, 562, 24 U.S.P.Q.2d (BNA) 1073, 1081 (1985).

[n.181]. See Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1522, 24 U.S.P.Q.2d (BNA) 1561, 1569 (9th Cir. 1992).

[n.182]. Id. at 1524, 24 U.S.P.Q.2d (BNA) at 1571. See also Harper & Row, Publishers, Inc. v. Nation Enters., 471 U.S. 539, 563-64, 24 U.S.P.Q.2d (BNA) 1073, 1082 (1985).

[n.183]. "To the extent that a work is functional or factual, it may be copied, as may those expressive elements of the work that 'must necessarily be use as incident to' expression of the underlying ideas, functional concepts, or facts." Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1524, 24 U.S.P.Q.2d (BNA) 1561, 1571 (9th Cir. 1992) (citing Baker v. Selden, 101 U.S. 99, 102-04 (1879)). Of course, in the context of the Lotus menu command hierarchy, this question has to a large extent already been answered by the above discussion of the idea/expression dichotomy. It also would be covered through application of the Altai test, where the copyrightable material is separated from the uncopyrightable.

[n.184]. See Harper & Row, Publishers, Inc. v. Nation Enters., 471 U.S. 539, 564-65, 24 U.S.P.Q.2d (BNA) 1073, 1082-83 (1985).

[n.185]. See Id. at 565, 24 U.S.P.Q.2d (BNA) at 1083, where Justice O'Connor wrote that "the fact that a substantial portion of the infringing work was copied verbatim is evidence

of the qualitative value of the copied material, both to the originator and to the plagiarist who seeks to profit from marketing someone else's copyrighted expression."

[n.186]. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 821, 34 U.S.P.Q. (BNA) 1014, 1026 (1st Cir. 1995) (Boudin, J., concurring).

[n.187]. Id.

[n.188]. Anthony L. Clapes and Jennifer M. Daniels, Lotus v. Borland: Nightmare on Milk Street?, Computer Law., May 1995, at 1, 19-20.

[n.189]. See Harper & Row, Publishers, Inc. v. Nation Enters., 471 U.S. 539, 560, 24 U.S.P.Q.2d (BNA) 1073, 1081 (1985) ("The factors enumerated in the section are not meant to be exclusive"); see also H.R. 1476, 94th Cong., 2d Sess. (1976) (doctrine is equitable rule of reason).

[n.190]. 49 F.3d 807, 34 U.S.P.Q.2d (BNA) 1014 (1st Cir. 1995).

[n.191]. Lotus Dev. Corp. v. Borland Int'l, Inc., No. 94-2003, 1996 U.S. LEXIS 470, at * 1 (Jan. 16, 1996).