Fall 1996


Mark L. Gordon

Follow this and additional works at: https://repository.law.uic.edu/jitpl

Part of the Antitrust and Trade Regulation Commons, Computer Law Commons, Intellectual Property Law Commons, Internet Law Commons, Privacy Law Commons, and the Science and Technology Law Commons

Recommended Citation

https://repository.law.uic.edu/jitpl/vol15/iss1/8

This Article is brought to you for free and open access by UIC Law Open Access Repository. It has been accepted for inclusion in The John Marshall Journal of Information Technology & Privacy Law by an authorized administrator of UIC Law Open Access Repository. For more information, please contact repository@jmls.edu.
COPYING TO COMPETE: THE TENSION BETWEEN COPYRIGHT PROTECTION AND ANTITRUST POLICY IN RECENT NON-LITERAL COMPUTER PROGRAM COPYRIGHT INFRINGEMENT CASES

by Mark L. Gordon†

I. INTRODUCTION

A. Copyright

The relationship between copyright law and antitrust policy is one of tension. Copyright law, as codified in the Copyright Act of 1976 (hereinafter "the Act"), provides the creator of an original work with a host of exclusive rights. The copyright owner can enjoy a virtual monopoly on the work's circulation throughout the market by fully exploiting the Act's potential. The Act provides the copyright owner with the following: 1) the right to reproduce the copyrighted work in copies; 2) the right to prepare derivative works based upon the copyrighted work; 3) the right to distribute copies of the work to the public; and, 4) the right to perform and display the work publicly. The backbone of the Act's potential grant of monopoly power lies in its copyright infringement provision. This provision allows the copyright owner to file suit against infringers and enjoin them from interfering with the owner's exclusive rights.

† Mark L. Gordon is a partner at Gordon & Glickson P.C., Chicago, Illinois. The author gratefully acknowledges the assistance of Timothy J. Harris and Michele M. Rivkin, who contributed to the preparation of this article.

2. Id.
3. Id.
4. Id.
5. Id.
7. Id.
Constitutional mandate justifies the Act's broad grant of power. Article 1, section 8 of the Constitution provides that Congress has the power:

To promote the progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries.\(^8\)

Therefore, copyright protection serves a dual purpose by rewarding the creator of the work and increasing the public's access to creative works. First, the exclusivity of the copyright rewards innovation and encourages research by allowing the copyright owner to reap the benefits of his labor and creativity. Second, the proliferation of original works of expression injected into mainstream society serves the public good.

\section*{B. Antitrust}

The antitrust laws, according to the United States Justice Department and the Federal Trade Commission, have the same goals in mind: to "promote innovation and consumer welfare by prohibiting certain actions that may harm competition with respect to either existing or new ways of serving consumers."\(^9\) The potential monopoly on a work, created by the Copyright Act's broad grant of exclusive rights, is antithetical to the antitrust concept of open competition. The Justice Department, which is the primary enforcer of United States antitrust regulations, does not wish to stifle creativity by denying copyright owners their exclusive rights. Yet, the monopoly power inherent in the Act is difficult to reconcile with a policy of disallowing market dominance. In other words, antitrust policy in this context focuses on "the balance between protecting intellectual property to reward innovation and maintaining competition in markets where innovation occurs."\(^10\) That balance is complicated and often instigates conflict.

\section*{C. Conflict in the Context of Computer Programs}

The conflict comes to a head with the protection of copyrighted computer programs where innovation is key. As confirmed by the legislative history of the Copyright Act, the 1980 amendments to the Act, and numerous court opinions, "computer programs are literary works entitled

---

\(^8\) U.S. Const. art. I, § 8, cl. 8.


\(^10\) Anne K. Bingaman, Assistant Attorney General, Antitrust Division, Antitrust and Innovation in a High Technology Society, Address at the Anniversary of the Founding of the Antitrust Division (Jan. 10, 1994), reprinted in 7 Trade Reg. Rep. (CCH) ¶ 50,128, at 48,998.
to copyright protection under federal law.” Computer programs are a comundrum for the courts because of their functional nature. This functional nature triggers antitrust concerns that were never an issue with other “literary works.” The Act states that “expression” is copyrightable, but that “ideas” are not. Computer programs incorporate protectable expression with “unprotectable” ideas and processes. Accordingly, courts often encounter difficulty in establishing the applicable level of protection which individual elements of a program should receive.

II. LITERAL COPYING OF COMPUTER PROGRAMS

A. PROTECTION OF LITERAL ELEMENTS

Copyright law protects the literal elements of a computer program, such as its source and object codes, because they contain protectable expression apart from ideas and processes. Accordingly, the literal copying of a computer program’s copyrighted expression (which is equivalent to word-for-word copying of a copyrighted newspaper article) is a clear violation of the copyright owner’s exclusive rights.

However, these exclusive rights are subject to a series of exceptions. The most relevant exception in the context of recent computer software cases is the “fair use” exception. This exception permits “an individual

---


12. 17 U.S.C. § 102 (1996). “Copyright protection subsists . . . in original works of authorship fixed in any tangible medium of expression . . . . In no case does copyright protection for an original work of authorship extend to any idea . . . regardless of the form in which it is described, explained, illustrated, or embodied in such work.” Id.

13. A program’s “source code” is the version of the program which is written in a human-readable programming language (such as BASIC, FORTRAN, C, or COBAL). The source code can then be “compiled” (or translated) into “object code,” which is written in machine language.


16. Another potentially active area of exception to literal copyright infringement, which also involves antitrust concerns, is that of “copyright misuse.” Although the copyright misuse defense, unlike fair use, is not codified, it has received some support in the
in rightful possession of a copy of a work to undertake necessary efforts to understand the work's ideas, processes, and methods of operation."\textsuperscript{17}

\textbf{B. Intermediate Copying}

The fair use exception has recently been applied in the area of "intermediate copying," thus serving to narrow the scope of copyright protection for computer programs. Intermediate copying involves the literal copying of computer codes as "an initial step in developing a competing product."\textsuperscript{18} In other words, the copying occurs merely as a means to an end—the final product is not similar to the copyrighted work, but a literal copy is made in the process of reaching that final product.

Pursuant to the plain language of the Copyright Act, intermediate copying constitutes copyright infringement.\textsuperscript{19} However, the courts, with apparent anti-monopoly concerns in mind, have endorsed some instances of intermediate copying as justifiable fair use.

1. \textit{Atari: Excessive Intermediate Copying}

In \textit{Atari Games Corp. v. Nintendo of America Inc.} (hereinafter "\textit{Atari}")\textsuperscript{20}, the Federal Circuit, applying Ninth Circuit law, recognized that intermediate copying may constitute fair use when such copying is necessary to understand the ideas and processes in a copyrighted

\begin{itemize}
  \item \textsuperscript{17} Atari Games Corp. v. Nintendo of Am., Inc., 975 F.2d 832, 842 (Fed. Cir. 1992). The fair use exception is codified in 17 U.S.C.A. § 107 (1996). For other cases involving fair use as an exception to literal-copy infringement, see Advanced Computer Servs., Inc. v. MAI Syss. Corp., 845 F. Supp. 356 (E.D. Va. 1994) (loading of copyrighted software into RAM did not qualify as fair use); Triad Syss. Corp. v. Southeastern Express Co., 31 U.S.P.Q.2d (BNA) 1239 (N.D. Cal. 1994) (where defendant copied a copyrighted program in order to service computers, insufficient evidence was presented on the adverse commercial impact prong of fair use determination to grant plaintiff summary judgment); Lewis Galoob Toys, Inc. v. Nintendo of Am., Inc., 964 F.2d 965 (9th Cir. 1992) (assuming that audiovisual displays created by "Game Genie" device, which allowed player to alter features of Nintendo's copyrighted video games, were derivative works, they did not constitute actionable infringement under the fair use doctrine).
  \item \textsuperscript{18} Sega Enterprises Ltd. v. Accolade, Inc., 977 F.2d 1510, 1518 (9th Cir. 1992).
  \item \textsuperscript{19} 17 U.S.C.A. § 106 (1)-(2) (1996). For a related proposition, albeit not in the area of computer law, see also Walker v. University Books, 602 F.2d 859, 864 (9th Cir. 1979) (stating that "the fact that an allegedly infringing copy of a protected work may itself be only an inchoate representation of some final product does not in itself negate the possibility of infringement").
  \item \textsuperscript{20} Atari, 975 F.2d 832 (Fed. Cir. 1992).
\end{itemize}
work. However, the Atari court found that the defendant's actions exceeded the bounds of fair use.

In this case, Atari attempted to unlock a security code (called the 10NES program) that prevented Nintendo’s video game consoles from accepting unauthorized game cartridges. Atari's end goal was to manufacture game cartridges which would be compatible with the Nintendo system. After Atari's original failed attempts to unlock the code by analyzing the 10NES chips, the corporation fraudulently obtained a Copyright Office copy of the program. Atari finally succeeded in “reverse engineering” the chips and developing a compatible program by using the unauthorized copy.

The court noted that Atari might lawfully reverse engineer the chips to understand the program's ideas and processes, without the help of an unauthorized program copy. However, any copying beyond that which was necessary to understand the program constituted an unfair use. Thus, Atari’s use of the illegally obtained copy of the 10NES program barred the application of the fair use exception.

The court's reasoning rested on a clear delineation between the Copyright Act's mandate and the concerns of antitrust policy—monopoly control is disfavored, but Atari's actions overstepped the bounds of competitive inquiry. The opinion suggests a narrowing of the scope of copyright protection in light of competition principles: intermediate copying, within limits, falls into the category of permitted activities under the Act's fair use exception.

2. Sega: Permissible Intermediate Copying

In Sega Enterprises Ltd. v. Accolade, Inc. (hereinafter “Sega”), the Ninth Circuit also considered the question of fair use with regard to intermediate copying, but concluded in favor of the defendant. The Sega court’s opinion rested on the unique nature of a computer program: computer programs being hybrid creations which combine creativity with functionality.

21. Id. at 843.
22. Id.
23. Id. at 836. "Reverse engineering" involves replicating and disassembling an object code which translates it back into source code, in order to understand a program's functions. Id.
25. Id. at 841-42.
26. Id. at 843.
27. Id.
28. 977 F.2d 1510 (9th Cir. 1992).
29. Id. at 1514,1527.
The facts in Sega were very similar to those in Atari, with two key exceptions. First, the Sega code, subsequently copied into the defendant's program, was primarily functional. Second, and more importantly, the Sega defendant managed to successfully reverse engineer Sega's lockout code without resorting to an unauthorized copy of the program. In holding that such reverse engineering constituted a fair use, the court noted that the defendant had no other means to access the unprotected ideas and functions of the program. Accordingly, the defendant had a "legitimate purpose" for its intermediate copying, despite the fact that its ultimate goal of creating things such as compatible games had a financial motive.

The Sega court stressed the fairness of its judgment in light of antitrust principles, noting that if such reverse engineering automatically constituted an unfair use, the copyright owner would hold a virtual monopoly on the functional aspects of a work. The court further reasoned that because section 102(b) of the Copyright Act expressly denies such protection, the protection is only available if the defendants meet the more stringent requirements of the patent laws. By allowing the defendant to disassemble Sega's code, the court believed it was promoting the production of creative works through increased competition which would ultimately benefit the public.

Both Sega and Atari indicate that copyright protection in the area of intermediate copying gives way to antitrust policy concerns and, as a result, the two become less oppositional. By incorporating and resolving antitrust concerns in copyright infringement cases, the courts reduce the monopoly possibilities of the Copyright Act in the area of computer technology.

III. NON-LITERAL COPYING OF COMPUTER PROGRAMS

The ongoing resolution of tension between copyright and antitrust has similarly begun in the "non-literal" copying cases. Unlike "literal copying," which refers to verbatim copying of the literal elements of a program, non-literal copying only involves the copying of the "essence

30. Id.
31. Id. at 1522.
32. Id.
33. Sega, 977 F.2d at 1522.
34. Id. at 1523.
35. Id. at 1523-24. Significantly, these courts are not engaging in comprehensive antitrust analysis. Instead, competition concerns surface merely as a subtext to a court's willingness to restrict copyright scope.
36. Examples of the literal elements of a program would include its source or object code.
or structure of a work."\(^{37}\)

Because the question of copying is quite hazy when it comes to non-literal elements of a computer program, the courts have great latitude in determining whether a copyright owner's rights have been infringed. Perhaps because of this latitude, antitrust concerns often appear as an important subtext in the courts' decisions. A brief outlay of the problem is illustrative.

In a standard copyright infringement claim, the copyright owner must prove: (1) ownership of a valid copyright; and, (2) copying by the defendant of protected elements of the copyrighted material.\(^{38}\) A certificate of copyright registration, which is prima facie evidence of validity, usually satisfies the first prong.\(^{39}\) However, defining which elements of the copyrighted work are actually protectable is often a more difficult task for the courts. Essentially, the scope of the grant of exclusivity for the copyright is determined once the court defines which elements are protectable.

With respect to the second prong of the test, "illegal copying," direct proof of such copying is rarely available. Rather, the plaintiff may prove copying by showing: (1) the defendant's access to the copyrighted work; and, (2) substantial similarity between the copyrighted work and the copy.\(^{40}\) Access is not hard to prove, but establishing substantial similarity can be significantly more difficult. Courts have noted that on a continuum between "no similarity" and "complete and literal similarity," the location of "substantial similarity" is very difficult to pinpoint.\(^{41}\)

### A. Pre-\textit{Altai} Broad Copyright Protection

In \textit{Computer Assocs. Int'l, Inc. v. Altai, Inc.} (hereinafter "\textit{Altai}"),\(^{42}\) the Second Circuit redefined the problem, paved the way for numerous future decisions, and placed previous decisions in perspective. Taking competition implications into consideration, the \textit{Altai} court constructed a three-part test to narrow the scope of copyright protection for the non-

---

39. Gates Rubber, 9 F.3d at 831.
40. Id. at 832-33.
41. Bateman, 79 F.3d at 1544 n.25 (quoting 3 \textsc{Melville B. Nimmer & David Nimmer, Nimmer on Copyright} § 13.03[A] (1995)).
42. 982 F.2d 693 (2d Cir. 1992).
literal elements of computer programs.\textsuperscript{43} The trend toward narrowed copyright protection is subtle. This trend becomes apparent only when viewed in the context of previously employed solutions to the problem.

Traditionally, courts focused on the idea-expression dichotomy when determining infringement. Under section 102 of the Copyright Act, copyright protection only attaches to expressions of ideas, not to the ideas themselves.\textsuperscript{44} Based on this distinction, the court devised the "extrinsic/intrinsic" test, also known as the "objective/subjective" test.\textsuperscript{45}

The first part of the test, which is the extrinsic or objective portion, focuses on the similarity of ideas between the copyrighted and challenged works.\textsuperscript{46} To find similarity of ideas courts must rely on expert testimony and analytic dissection.\textsuperscript{47} The second part of the test, which is the intrinsic or subjective portion, focuses on whether the expression of the ideas in the two works is substantially similar.\textsuperscript{48} The basis for this second part is the response of the "ordinary reasonable person" because expert testimony is not admissible.\textsuperscript{49}

In \textit{Whelan Assocs. v. Jaslow Dental Lab.}\textsuperscript{50} (hereinafter "\textit{Whelan}")\textsuperscript{51}, the Third Circuit took a different approach by designing a test for cases involving computer programs. Notably, the court dropped the ordinary reasonable person test for similarity because computer programs are complex and unfamiliar to most members of the public.\textsuperscript{52} Instead, the \textit{Whelan} court adopted a single substantial similarity test, focusing on the initial filtration of ideas from expression.\textsuperscript{53} The court summarized the test as follows:

\begin{quote}
[The line between idea and expression may be drawn with reference to the end sought to be achieved by the work in question. In other words, the purpose or function of a utilitarian work would be the work's idea, and everything that is not necessary to that purpose or function would be part of the expression of the idea . . . . Where there are various means of
\end{quote}

\footnotesize
\begin{itemize}
  \item \textsuperscript{43} \textit{Id.} at 711-12.
  \item \textsuperscript{44} 17 U.S.C. § 102 (1996); see supra text accompanying note 11; see also \textit{Baker v. Selden}, 101 U.S. 99 (1879) (finding that copyright protection does not attach to a system of bookkeeping).
  \item \textsuperscript{45} The Second Circuit first set out this test in \textit{Arnstein v. Porter}, 154 F.2d 464, 468-69 (2d Cir. 1946), \textit{cert. denied}, 330 U.S. 851 (1947), and other Circuits subsequently adopted and reformulated \textit{Arnstein}'s test. \textit{See, e.g.,} \textit{Sid & Marty Krofft Tele. Prod., Inc. v. McDonald's Corp.}, 562 F.2d 1157, 1164-65 (9th Cir. 1977). Although these early cases did not concern computer programs, the intrinsic/extrinsic test was later applied in that context.
  \item \textsuperscript{46} \textit{Arnstein}, 154 F.2d at 468.
  \item \textsuperscript{47} "Analytic dissection" is the analysis of distinct aspects of the computer program utilizing objective criteria.
  \item \textsuperscript{48} \textit{Krofft}, 562 F.2d at 1164.
  \item \textsuperscript{49} \textit{Id.}
  \item \textsuperscript{50} 797 F.2d 1222 (3d Cir. 1986).
  \item \textsuperscript{51} \textit{Id.} at 1232.
  \item \textsuperscript{52} \textit{Id.} at 1236.
\end{itemize}
achieving the desired purpose, then the particular means chosen is not necessary to the purpose; hence, there is expression, not idea.\textsuperscript{53} The Whelan test’s focus on the \textit{singularity} of a work’s idea necessarily offers broad protection to computer program copyright holders.\textsuperscript{54} After distilling the program’s idea, the remaining multitude of expression affords protection to the full extent of the Act.\textsuperscript{55} However, because of serious negative repercussions in the technology market, courts since Whelan, as well as several commentators, have criticized Whelan as overbroad and ill-fitted to computer copyright law.\textsuperscript{56}

\textbf{B. \textit{Altai} Abstraction-Filtration-Comparison Test}

In \textit{Altai}, the Second Circuit criticized the Whelan test as being inapplicable to the very field in which the case broke ground—non-literal copying of computer programs.\textsuperscript{57} Because of a computer program’s reliance on subroutines, and since each subroutine has its own idea, the \textit{Altai} court found Whelan’s declaration that a program’s overall purpose constituted the program’s idea to be “descriptively inadequate.”\textsuperscript{58} Seeking to overcome these shortcomings, the Second Circuit Court, in \textit{Altai}, introduced the three-part \textit{Abstraction-Filtration-Comparison} test (“\textit{Altai test}”).\textsuperscript{59} The \textit{Altai} test reconstructs the standard copyright infringement test. The question of copyright ownership, with the attendant question of actual copyrightability of particular components, becomes juxtaposed with the issue of substantial similarity.

The \textit{Altai} test consists of three stages: abstraction, filtration, and comparison. The “abstraction” stage involves breaking down the copyrighted program into its structural parts.\textsuperscript{60} The second stage, “filtration,” entails examining each part and separating protectable expression from ideas and non-protectable expression.\textsuperscript{61} Finally, in the “comparison” stage, the fact-finder compares whatever protected “kernels of creative expression” remain with the allegedly infringing program to determine substantial similarity.\textsuperscript{62}

The \textit{Altai} approach narrows the scope of copyright protection because of the limiting aspect of each step. Abstraction splices the program into varying levels of generality, thus allowing the court to make numer-

\textsuperscript{53} Id.
\textsuperscript{54} \textit{Altai}, 982 F.2d 693, 705 (2d Cir. 1992).
\textsuperscript{55} Id.
\textsuperscript{56} Id.
\textsuperscript{57} Id.
\textsuperscript{58} Id.
\textsuperscript{59} \textit{Altai}, 982 F.2d at 706-10.
\textsuperscript{60} Id. at 706.
\textsuperscript{61} Id. at 707.
\textsuperscript{62} Id. at 706.
ous micro-level assessments of protectability. The abstraction process is contradistinguished from the Whelan test, which considers the program as a whole. Filtration requires a court to exclude the components of a program that are not protectable, such as: 1) elements dictated by efficiency; 2) elements dictated by external factors; 3) uncreative or unoriginal elements; 4) ideas; 5) processes; 6) facts; 7) methods; 8) scientific discoveries; and, 9) public domain information. As courts add to the list of a computer program’s unprotectable elements, they will correspondingly reduce the expansiveness of the copyright holder’s monopoly.

Thorough application of abstraction and filtration could result in elimination of any kernel of creative expression that needs comparison. Assuming the court reaches the comparison step, the Altai test imposes one final limitation on copyright scope: the “de minimis” rule. Here, the court considers the copied expression in light of the expression’s significance with respect to the overall program at issue. Thus, even if substantial similarity exists, the court may refuse to find infringement if the copied expression is insignificant.

Competition considerations were a significant factor in the Altai court’s reasoning. The court recognized the copyright-narrowing implications of this new test, balanced them against anti-monopoly interests, and decided in favor of the latter. Acknowledging the poor fit of copyright protection and computer programs and suggesting that patent registration for computer software might be more suitable, the court noted that the Whelan rule could enable first-comers to “lock up basic programming techniques.” The crux of the Altai court’s policy concern is the deleterious effect of broad protection of a program’s utilitarian components. As a remedy, the Altai court offered the three-part test to reduce

---

63. Id. at 707.
64. See supra note 52-53 and accompanying text.
65. Altai, 982 F.2d at 708. When there are only a limited number of efficient and workable options for expressing an idea, those limited options are not protected by copyright law. Id.
66. Id. at 709. Examples of external factors would include the standard techniques, hardware, and scènes à faire. Under the scènes à faire doctrine, expressions that are indispensable, or at least standard, to expressing an idea are not copyrightable. Id. The scènes à faire doctrine includes stock characters and plots in dramatic works and “now encompasses stereotyped expression, standard or common features in a wide variety of works, including audiovisual works generated by computers.” Apple Computer Inc. v. Microsoft Corp., 799 F. Supp. 1006,1021 (N.D. Cal. 1992).
67. Altai, 982 F.2d at 707-10.
68. Id. at 710.
69. Id.
70. Id. at 715. The court, noting the lack of persuasive evidence on substantial similarity, found it unnecessary to reach the “de minimis” issue. Id.
71. Id. at 712 (quoting Peter S. Menell, An Analysis of the Scope of Copyright Protection for Application Programs, 41 Stan. L. Rev. 1045, 1085-86 (1989)).
monopoly control of useful morsels of technology.\textsuperscript{72}

C. \textit{POST-ALTAI NARROW COPYRIGHT PROTECTION}

Since \textit{Altai}, courts have used the same three-part test (or variations thereof) to subtly steer the law in the direction of narrower protection for a computer program’s non-literal elements. While the case law is limited, it appears that monopoly concerns are an important subtext in many of these decisions.

1. \textit{Ninth Circuit}

The Ninth Circuit, beginning with \textit{Brown Bag Software v. Symantec Corp.} (hereinafter “\textit{Brown Bag}”),\textsuperscript{73} and continuing with \textit{Capcom U.S.A., Inc. v. Data East Corp.} (hereinafter “\textit{Capcom}”),\textsuperscript{74} and \textit{Apple Computer, Inc. v. Microsoft Corp.} (hereinafter “\textit{Apple}”),\textsuperscript{75} has undertaken a filtering analysis, similar to the \textit{Altai} test, which yields few protectable elements in non-literal infringement actions.

In \textit{Brown Bag}, which was decided before \textit{Altai}, the Ninth Circuit reformulated the old extrinsic/intrinsic test into one that bears a striking resemblance to the \textit{Altai} three-part test.\textsuperscript{76} The revised extrinsic portion of the test still uses analytic dissection,\textsuperscript{77} but this revision determines whether similarities between two programs result from protected or unprotected expression.\textsuperscript{78} In other words, the court breaks down the program and separates out the unprotectable elements in order to determine the scope of the owner’s copyright, much like \textit{Altai}’s abstraction and filtration steps.\textsuperscript{79}

\textit{Capcom} followed \textit{Brown Bag}’s modification of the extrinsic/intrinsic test, again using an \textit{Altai}-like filtration that removes many of a program’s components from the protection of copyright. In holding that the defendant, Data East, did not infringe on plaintiff Capcom’s copyright on a series of reality-based video games called \textit{Street Fighter II}, the court stressed the unprotectability of many of \textit{Street Fighter II}’s elements.\textsuperscript{80} In particular, the elements allegedly copied by Data East were unprotectable because of the idea/expression dichotomy, the “merger doc-

\begin{itemize}
\item \textsuperscript{72} See supra notes 63-70 and accompanying text.
\item \textsuperscript{73} 960 F.2d 1465 (9th Cir. 1992).
\item \textsuperscript{74} No. 93-C3259, 1994 U.S. Dist. LEXIS 5306 (N.D. Cal. Mar. 16, 1994).
\item \textsuperscript{75} 35 F.3d 1435 (9th Cir. 1994), cert. denied, 115 S. Ct. 1176 (1995).
\item \textsuperscript{76} See supra notes 60-70 and accompanying text.
\item \textsuperscript{77} See supra note 47.
\item \textsuperscript{78} \textit{Brown Bag}, 960 F.2d at 1475.
\item \textsuperscript{79} Id. at 1475-76.
\item \textsuperscript{80} \textit{Capcom}, 1994 U.S. Dist. LEXIS 5306 at *35.
\end{itemize}
trine,"81 functional and practical limitations, "inherent utility,"82 the "scènes à faire" doctrine,83 and lack of creativity or originality.

When the Capcom court finally applied the intrinsic test, which mimics the Altai comparison component, the court used a virtual identity standard rather than a substantial similarity standard.84 The court reasoned that where the alleged similarities to a copyrighted work consist primarily of unprotectable elements or are "capable of only a narrow range of expression," the relevant standard is "virtual identity."85 Since few items remained after filtration, and those that remained were not virtually or even substantially identical to Data East’s copies, the court denied Capcom’s motion for a preliminary injunction.86

Prior to the opinion’s conclusion, competition concerns entered the calculus, suggesting the policy underlying the limiting test.87 The court noted that copyright law did not entitle Capcom to a monopoly on either "a range of characters and moves that it did not create" or on the idea of "reality based fight games featuring human characters."88 This modified extrinsic/intrinsic test, like the Altai test, serves to narrow the scope of copyright to incorporate antitrust policy and significantly reduce the opposition between these two policies.89

In Apple, the Ninth Circuit again employed the extrinsic/intrinsic test, explicitly acknowledging the test’s counterpart in Altai.90 The court concluded that the defendant, Microsoft, was not liable for copyright infringement in its development of a graphical user interface similar to that used by the plaintiff’s Macintosh system.91 In doing so, the court engaged in a piece-meal analysis, finding that many elements of Apple’s interface were unprotectable because of the merger doctrine, the scènes à faire doctrine, lack of originality, and inherent utility.92 Since Apple li-

81. Id. at *16. Under the merger doctrine, if an idea and the expression of that idea cannot be separated, then the expression is only protected from identical copying. Id.
82. Id. at *21. The court explained that if an element performs a highly utilitarian function (for example, a useful purpose), then the element is protected if it contains "separate artistic features." Id.
83. See supra note 66 and accompanying text.
85. Id. at 36.
86. Id. at 43.
87. Id.
88. Id.
89. Again, in reaching a decision, the courts in computer program copyright cases are not entering into detailed antitrust analysis. Rather, antitrust concerns are taken into consideration, as a policy matter, without discussion of the intricacies of antitrust law.
90. Apple, 35 F.3d at 1445 (citing Computer Assocs. Int’l v. Altai, Inc., 982 F.2d 693, 706-11 (2nd. Cir. 1992)). The court noted that "other courts perform the same analysis, although articulated differently." Id.
91. Id. at 1447.
92. Id. at 1444-46.
Censored out many of the interface's features to the defendant, and because few features remained after analytic dissection, the work as a whole was entitled to only limited copyright protection. Accordingly, the Ninth Circuit applied the standard of virtual identity and affirmed the district court's verdict for Microsoft.\textsuperscript{93}

Once again, utilitarian concerns were paramount. By refusing to grant Apple patent-like protection for the graphical user interface idea, the court sought to prevent first-comers from obtaining monopoly control over useful technology.\textsuperscript{94} Again, a subtext of monopoly fear supported the court's application of a copyright-narrowing, bit-by-bit dissection.

2. Tenth Circuit

The Tenth Circuit's position on the scope of non-literal copyright protection is somewhat less clear. In \textit{Gates Rubber Co. v. Bando Chem. Indus.} (hereinafter "\textit{Gates}").\textsuperscript{95} the Tenth Circuit adopted the \textit{Altai} test for determining the scope of copyright protection of computer programs. However, the court added an additional component to the abstraction test which has broader implications on copyright scope. The court recommended an "initial holistic comparison," reasoning that by first viewing the two programs as a whole, the court will be better able to determine the factual issue of copying.\textsuperscript{96}

While this step does not replace the analytical breakdown which is fundamental to \textit{Altai}'s test, the new component does create another opportunity for finding infringement. Accordingly, although the \textit{Gates} court rejected the lower court's broad view of copyright protection and reversed the finding of infringement,\textsuperscript{97} the state of the law in the Tenth Circuit remains somewhat hazy. The \textit{Altai} test is the test of choice, but how strictly the courts will apply that test remains to be seen.

3. First Circuit

The First Circuit's highly publicized decision in \textit{Lotus Dev. Corp. v. Borland Int'l} (hereinafter "\textit{Lotus}")\textsuperscript{98} also follows the trend of narrowed protection in computer program copyright cases, but for a different reason. \textit{Lotus} involved the defendant Borland's deliberate and literal copy-

\textsuperscript{93} \textit{Id.} at 1446.
\textsuperscript{94} \textit{Apple}, 35 F.3d at 1446.
\textsuperscript{95} 9 F.3d 823 (10th Cir. 1993).
\textsuperscript{96} \textit{Id.} at 841. The \textit{Gates} court noted that by first comparing the two programs in their entirety, a pattern of copying may be detected that would not be readily apparent when comparing only the individual components. \textit{Id.}
\textsuperscript{97} \textit{Id.}
\textsuperscript{98} 49 F.3d 807 (1st Cir. 1995), \textit{aff'd by an equally divided Court}, 116 S.Ct. 804 (1996).
ing of the “menu tree” in the plaintiff's popular Lotus spreadsheet program. Borland copied the menu tree in order to provide users of its own spreadsheet program, Quattro, with an alternate user interface.

The First Circuit refused to apply the Altai test to this case, holding that the test only applied to non-literal copying cases, as opposed to the literal copying of the menu tree at issue. Nevertheless, the court found for Borland, reasoning that the menu command hierarchy was a “method of operation” under section 102(b) of the Act and, thus, was unprotectable. Therefore, even though Borland copied verbatim from Lotus, the copy did not infringe on Lotus's copyright because the menu tree was uncopyrightable.

Although the competition inquiry was absent from the court’s opinion, the concurrence stressed the underlying anti-monopoly concerns. The concurring opinion noted that if the court granted Lotus a monopoly on the menu tree, then users who had learned or created macros for the program would be locked into that program, “just as a typist who has learned the QWERTY keyboard would be the captive of anyone who had a monopoly on the production of such a keyboard." As in Apple, the utilitarian aspect was significant: even if a work is creative, granting broad copyright protection could limit one's ability to efficiently perform a task. The First Circuit's decision to limit Lotus's hold on the market does not bode well for software manufacturers concerned about the scope of their copyrights.

After the Supreme Court granted certiorari in the fall of 1995, the software industry eagerly awaited a decision in hopes of receiving further clarification regarding the scope of copyright protection for computer software. Instead, an equally divided Supreme Court issued an unsigned, one-sentence opinion affirming the First Circuit's judgment. Although the decision settled the matter between Lotus and Borland, it did nothing to clarify this murky area of copyright law.

4. Fifth Circuit

The Fifth Circuit, like the Tenth Circuit, is in a state of flux with respect to the non-literal copying question. This court first considered

99. Lotus, 49 F.3d at 810. Lotus’ “menu tree” contained nearly 500 commands arranged into menus and sub-menus in a clearly defined hierarchy. Id.
100. The literal copying of the menu tree also made it easier for Lotus users to switch to Quattro. Id.
101. Id. at 814.
102. Id. at 818 (citing 17 U.S.C. §102(b) (1976)).
103. Id. at 821 (Boudin, J., concurring in the judgment).
105. Id. (Justice Stevens took no part in the holding). Id.
106. Id. The Supreme Court denied Lotus’ Petition for Rehearing on March 4, 1996. Id.
the issue in *Kepner-Tregoe, Inc. v. Leadership Software, Inc.* (hereinafter "*Kepner-Tregoe*"), which is notable only for the court's recognition that, consistent with Supreme Court precedent, "non-literal aspects of copyrighted works, such as structure, sequence, and organization, may receive protection under copyright law." The court went no further in defining the extent of such protection until rendering an opinion in *Engineering Dynamics, Inc. v. Structural Software, Inc.* (hereinafter "ED/"). The issue presented was whether input and output formats of a computer user interface were protectable. Although the Fifth Circuit embraced the *Altai* test, the Circuit actually expanded the scope of copyright protection to potentially include input formats.

However, the foundation for this result is shaky, because of the court's substantial reliance on *Lotus Development Corp. v. Borland Int'l, Inc.* (hereinafter "*Lotus*"), the district court opinion that was overturned by the First Circuit. Therefore, the answer remains unclear as to whether the Fifth Circuit would come to the same result today or whether the court would instead adopt the reasoning of the latest *Lotus* opinion, which was affirmed by a divided Supreme Court.

5. **Eleventh Circuit**

In *Bateman v. Mnemonics, Inc.* (hereinafter "*Bateman*"), the Eleventh Circuit adopted the *Altai* test, but held that a jury instruction limiting the test's application to non-literal elements was misleading. The case involved allegations of both literal and non-literal copying of a copyrighted program. The court held that, although *Altai* pertained to non-literal copying cases, the same filtering analysis should be applied in cases of literal copying. In other words, the fact-finder should be required, even in a literal copying case, to consider the defendant's potential challenges to copyright scope. The extension of *Altai* outside the non-literal arena further demonstrates the erosion of copyright protection available for computer software.

---

107. 12 F.3d 527 (5th Cir.), cert. denied, 115 S. Ct. 82 (1994).
109. 26 F.3d 1335 (5th Cir. 1994).
111. See supra note 98 and accompanying text.
112. 79 F.3d 1532 (11th Cir. 1996).
113. *Bateman*, 79 F.3d at 1543.
114. Id. at 1545-46. Once again examples of copyright scope would include the idea/expression dichotomy, merger doctrine, scènes à faire, etc. Id.
6. District Courts

A few recent district court opinions also demonstrate the narrowing scope of copyright protection in non-literal computer cases. *Productivity Software Intl, Inc. v. Healthcare Tech., Inc.* (hereinafter "PSI") involved a rigorous application of the *Altai* test, which resulted in summary judgment in favor of the alleged infringer. The court found that the plaintiff's word-processing program for automatically expanding abbreviations into full words contained many unprotectable elements such as the general idea, factors dictated by efficiency, and common features of computer software. After the court applied abstraction and filtration, only one element of the plaintiff's program, the screen displays, remained for consideration. Noting that the similarities between the two screen displays were generally dictated by industry standards, the court found that the programs were not substantially similar. As this strict application of the *Altai* test shows, much of the non-literal nature of a computer software product may be appropriated without incurring copyright infringement liability.

In *MiTek Holdings, Inc. v. Arce Engineering Co.* (hereinafter "MiTek"), the District Court adopted the *Altai* test as the most widely accepted approach, citing *EDI* and Gates with approval. After applying *Altai*’s abstraction and filtration steps, the court had only five protectable elements of the plaintiffs’ program left. Despite the fact that four of these elements were substantially similar to elements in the defendant’s program, the court denied plaintiffs’ motion for a preliminary injunction. Pursuant to *Altai*’s “de minimis” rule, the court found that these elements were too insignificant in the context of the plaintiffs’ programs as a whole to constitute infringement. This holding further demonstrates *Altai*’s restrictive trend.

*CMAX/Cleveland, Inc. v. UCR, Inc.* (hereinafter “CMAX”) also involved the rejection of Whelan and the adoption of the *Altai* test. Although the *CMAX* court found infringement, the finding does not suggest a broadening of copyright scope. Rather, because *CMAX* was a

---

116. Id. at *20-21.
117. Id. at *12-15.
118. Id. at *16-20.
119. Id. The court also noted that, because the screen displays contained little “original expression,” they were only entitled to narrow protection. Id. at *18-19 (citing *EDI*, 26 F.3d at 1344-46).
121. Id. at 1577.
122. Id. at 1584.
124. Id. at 359-60.
clear case of software cloning, this finding merely demonstrates a limitation in the courts' willingness to restrict copyright monopolies.

IV. CONCLUSION

In recent cases, the courts have narrowed the scope of copyright protection for the non-literal elements of computer programs. The application of the fair use doctrine to include intermediate copying, the employment of a dissection test (whether the extrinsic/intrinsic or the Altai test) to reduce the quantity of protectable elements, and the recent Lotus decision have all contributed to this subtle trend. The result of this trend is a lessening of the tension between copyright exclusivity and antitrust policy in the computer software context. In other words, based on a copyright holder's monopoly potential—and on the courts' corresponding limitation of the scope of copyright protection—copyright law and antitrust policy appear less at odds in recent non-literal copying cases. The courts are addressing monopoly concerns by scaling back copyright protection in an attempt to promote growth in the computer technology field.

Since case law in this area is extremely limited, the conclusions drawn in this article are nothing more than informed speculation. The few recently-decided cases on non-literal copyright infringement appear to suggest a trend, but it is too soon to tell precisely where the courts are headed. Further, whether this trend continues remains in large part dependent upon the next significant case decided by the Supreme Court in this area. Unfortunately, the Supreme Court's failure to clarify the landscape to date leaves the scope of copyright protection for non-literal elements of computer software uncertain.