New York Law Journal
SERVING THE BENCH AND BAR SINCE 1888
Web address: http://www.nylj.com
VOLUME 235—NO. 89 TUESDAY, MAY 9, 2006 ALM

COMPUTER LAW

BY RICHARD RAYSMAN AND PETER BROWN

Copyright Infringement of Computer Software and the 'Altai' Test

Computer programs are protectable under copyright law, with such protections extending to the source code and object code,[1] as well as applications and operating system programs.[2]

In the 20 years since the law became settled concerning the copyrightability of software, varying approaches and tests have been developed by the federal judiciary to determine when infringement of computer software has occurred. Often, to determine whether copying is actionable, the critical issue is whether there is so-called "substantial similarity" between the defendant's work and the protectable elements of the plaintiff's work, a question that is not always easily answered in the computer software realm.

Despite fairly extensive litigation in this area, courts have yet to adopt a universal test for copyright infringement of "non-literal" aspects of computer software.

Nonetheless, many seemingly have begun to embrace the U.S. Court of Appeals for the Second Circuit's *Altai* test[3] or variations of it when determining when there is a substantial similarity between two computer programs.

'Literal' and 'Non-Literal' Copying

Generally speaking, "literal" copying is the verbatim copying of original expression, while "non-literal" copying is that which is paraphrased, or loosely paraphrased. See *Ilog Inc. and Ilog S.A. v Bell Logic, LLC*, 181 FSupp2d 3 (D. Mass. 2002), citing *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 49 F3d 807, 814 (1st Cir. 1995). It is one of the basic tenets of copyright law that protection of a literary property (which includes computer programs) cannot be limited to the text (or software code), or else a plagiarist would sidestep the law by simply making slight variations from the original work. Literal copying is fairly simple to detect; accordingly, the more difficult cases that reach the courts are typically for non-literal copying of software.

Richard Raysman *Peter Brown*

'Literal' and 'Non-Literal' Elements

Courts also classify the elements of computer programs into literal and non-literal categories. Generally, source and object codes constitute the "literal" elements. *Computer Associates International, Inc. v. Altai, Inc.*, 982 F2d 693, 702 (2d Cir. 1992). The "non-literal" elements of software consist of elements other than the code and include the program's "structure, sequence, and organization," as well as elements of the program that are the products generated by the code's interaction with the computer hardware and operating program, and "the various steps a programmer employs prior to actually writing the instructions or source code." See *O.P. Solutions, Inc. v. Intellectual Prop. Network, Ltd.*, 1999 USDistLEXIS 979 at *18 (SDNY Feb. 2, 1999); *Cognotec Services, Ltd. v. Morgan Guar. Trust Co. of New York*, 862 FSupp 45, 49 n.3 (SDNY 1994).[4] Thus, a defendant may commit copyright infringement of literal elements if he copies the plaintiff's source or object code, or infringe the non-literal elements if he copies the program's structure, including general flow charts and the more specific organization of the program, including "inter-modular relationships, parameter lists, and macros." *Altai*, 982 F2d at 702. Often, however, courts do not always distinguish between the two, or simply mischaracterize what is at issue in a certain case. See *MiTek Holdings, Inc.* 89 F3d at 1556 n.16. Indeed, as the U.S. Court of Appeals for the Second Circuit has noted: "To be frank, the exact contours of copyright protection for non-literal program structure are not completely clear." *Altai* at 712.

Early Approaches

One of the earliest approaches applied to infringement of computer software was the "iterative approach," which was highlighted in 1985 by the district court in *E.F. Johnson Co., v. Uniden Corp. of America*, 623 FSupp 1485 (D. Minn.

Richard Raysman *and* **Peter Brown** *are partners at Brown Raysman Millstein Felder & Steiner. They are co-authors of "Computer Law: Drafting and Negotiating Forms and Agreements" (Law Journal Press).*

1985). Noting that several other courts had recently employed the test in form, if not name, the court explained the approach's formulation. To establish a prima facie case of infringement using this approach, it must be shown that (1) the defendant "used" the copyrighted work in preparing the alleged copy, which may be established by proof of access and similarity sufficient to reasonably infer use of the copyrighted work; and (2) that the defendant's work is an iterative reproduction, that is, one produced by iterative or exact duplication of substantial portions of the copyrighted work. Id at 1493. Furthermore, under this method, expert testimony is essential to the infringement analysis because the fact finder's focus shifts to an analysis of the "quantitative and qualitative evidence of similarities" as gauged by the court's evaluation of expert testimony. Id. at 1493. This approach seemingly has not seen widespread use, even in its own district, as the types of computer programs have became more sophisticated since the case was decided and other tests have been developed.[5]

Shortly after the *E.F. Johnson* decision, the U.S. Court of Appeals for the Third Circuit adopted its own test for infringement in *Whelan Associates, Inc. v. Jaslow Dental Laboratories, Inc.*, 797 F2d 1222 (3d Cir. 1986). In that case and subsequent cases, the critical question was to what extent a non-literal aspect of a program could be considered copyrightable "expression," as opposed to an uncopyrightable "idea" under the Copyright Act. In *Whelan*, the court held that a program's structure, sequence, and organization does not necessarily represent an uncopyrightable idea, but may in fact constitute copyrightable expression. Id. at 1248. Furthermore, *Whelan* defined the idea of a computer program in terms of its function or purpose and stated that anything not necessary for that purpose or function should be considered the expression of that idea. The court found that a showing of substantial similarity between the literal elements of the original and the allegedly infringing copy is not necessary for a finding of copyright infringement, as the appropriation of a work's "total concept and feel" is sufficient for such a finding.

Whelan and its progeny have been criticized as setting a vague and difficult standard with respect to the idea/expression dichotomy as it relates to computer programs and copyright infringement. See e.g., *Altai* 704. Since *Whelan*, however, the Second Circuit articulated a more specific standard, one that has been embraced by other circuit courts.

The 'Altai' Test

The Second Circuit, in *Computer Associates International, Inc. v. Altai, Inc.*, 982 F2d 693, 702 (2d Cir. 1992), found that *Whelan*'s "structure, sequence and organization" test did not provide a fair method for considering copyright infringement claims with respect to computer software. Id. at 706. Alternatively, the court proposed a new three-part method, commonly known as the *Altai* or "abstraction-filtration-comparison" test to determine whether elements of a computer program qualify for copyright protection and could therefore subject a defendant to liability for copying. Essentially, the "abstraction-filtration-comparison test" provides a structured approach to analyzing computer programs under traditional doctrines of copyright law developed to differentiate unprotectable ideas from protectable expression, as well as protectable forms of expression from unprotectable forms. *Maddog Software Inc v. Skadler*, 382 FSupp2d 268 (D.N.H. 2005).

The court proposed a three-part method, known as the 'Altai' or "abstraction-filtration-comparison" test to determine if elements of a computer program qualify for copyright protection and could subject a defendant to liability. The test has a structured approach to analyzing computer programs under traditional doctrines of copyright law developed to differentiate unprotectable ideas from protectable expression.

Although these various tests (or slight variations of them) are used in copyright infringement cases, depending upon the jurisdiction, the *Altai* test, in addition to binding district courts in the Second Circuit,[6] seems to be emerging as the preferred one with several other circuits having adopted its analysis or discussed it with approval.[7] The test operates as follows:

Abstraction

- Apply the *Nichols* abstractions test[8] to separate the uncopyrightable idea of a program from its copyrightable expression. The analysis begins with the code, and working backward through the various levels of abstraction, end with a general statement of the program's idea. The levels of abstractions would be (i) the main purpose, (ii) the program structure or architecture, (iii) modules, (iv) algorithms and data structures, (v) source code, and (vi) object code.

Filtration

- Filter the allegedly infringed program at each level of abstraction through various copyright doctrines, such as merger and scènes à faire[9] deny protection to certain types of materials. For example, the scènes à faire doctrine denies protection to program elements that are dictated by external factors such as "the mechanical specifications of the computer on which a particular program is intended to run," considerations of efficiency, or "widely accepted programming practices within the computer industry." *Altai* at 709-10.

Comparison

- "Compare" the remaining core to the allegedly infringing work, to determine whether the defendant copied any aspect of this remaining protected expression, "as well as an assessment of the copied portion's relative importance with respect to the plaintiff's overall program."[10]

In short, using the *Altai* test, the court will ultimately determine whether the

alleged infringing program is "substantially similar" to the plaintiff's original work.

Variations on 'Altai' Test

In *Ilog Inc. and Ilog S.A. v. Bell Logic, LLC*, 181 FSupp2d 3 (D. Mass. 2002), for example, the U.S. District Court for Massachusetts was presented not with allegations that an entire program had been copied, but rather with complaints that plaintiff ILOG copied particular elements of a program. *Ilog* is interesting because the court used the *Altai* test, in part, to reach its conclusion. Noting that the U.S. Court of Appeals for the First Circuit has not explicitly adopted the Second Circuit's "abstraction-filtration-comparison test," the court commented that "the *Altai* [abstraction-filtration-comparison] test may provide a useful framework for assessing the alleged non-literal copying of computer code," (quoting *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 49 F3d 807, 815 (1st Cir. 1995)).

Consequently, the *Ilog* court was not faced with the task of separating the composite elements of an entire program into levels of abstraction. Instead, the *Ilog* court needed to determine whether the elements complained of were copyrightable. *Ilog* at 22. Accordingly, under the *Altai* test, it was unnecessary for the court to perform the "abstraction" part of the test, but rather only "filter" the identified elements. Id. at 28. Having determined that none of the elements of the defendant's program allegedly copied were copyrightable, the court ultimately found that no infringement occurred.

Avoiding Infringement

By paying careful attention to the exclusive rights granted to copyright holders throughout the process of development and licensing of competitive software products, parties can attempt to limit the types of conduct that may lead to a claim of infringement.

Many software products are designed to perform substantially the same functions and compete for the same groups of users. When this fact is coupled with the reality that many such competing software products are programmed using similar strains of software code, serious concerns may arise about the possibility of infringement during product development. Case law has shown, however, that one of the ways[11] to avoid a claim that a new, competing product infringes upon an existing product is to implement a "clean room" procedure during development. See *DSC Communication Corp. v. DGI Technologies*, 898 FSupp 1183 (N.D. Tex. 1995), aff'd 81 F3d 57 (5th Cir. 1996); *NEC Corp. v. Intel Corp.*, 1989 U.S. Dist. LEXIS 1409 (D. Cal. Feb. 6, 1989).

In short, a clean room procedure separates those developing the specifications for the software (parties who may be exposed to other software products) from those actually programming the code. It further prevents both the specification developers and the code programmers from having contact with the competing product. In the end, a clean room procedure allows a software developer to assemble evidence of independent creation of software, despite any similarities that may exist between the new product and any competing products.[12]

Conclusion

Although the *Altai* test, and variations of it, appears to be the favored method for addressing copyright infringement of computer software, it is arguably not the only test (see, e.g., *Whelan* test, supra), or approach available to a court deciding whether there has been a copyright infringement of computer software.

Additionally, counsel should be acquainted with the *Altai* test and its variations when faced with the issue of computer software infringement and determine if the test or a variation of it predominates in the circuit in which the case is being litigated. Until the Supreme Court settles the issue or until appropriate legislation is passed, courts likely will continue to use these tests, or perhaps even develop other methods or variations to determine whether copyright infringement has occurred when it comes to non-literal infringement of computer software.

1. See e.g., *Midway Manufacturing Co. v. Strohon*, 564 F Supp. 741 (N.D. Ill. 1983); *Apple Computer, Inc. v. Formula International, Inc.*, 562 F. Supp. 775 (C.D. Cal. 1983), aff'd 725 F. 2d 521 (9th Cir. 1984).

2. See e.g., *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F. 2d 1240 (3d Cir. 1983); See also 17 USC §101 and §117. While computer programs are not specifically listed as part of the above statutory definition, the legislative history leaves no doubt that Congress intended them to be considered literary works. See H.R.Rep. No. 1476, 94th Cong., 2d Sess. 54, reprinted in 1976 U.S.C.C.A.N. 5659, 5667).

3. *Computer Associates International, Inc. v. Altai, Inc.*, 982 F.2d 693, 702 (2d Cir. 1992).

4. Courts also have extended the definition of non-literal elements to include its screen displays and the main menu and submenu command tree structure contained within the program's user interface. See *MiTek Holdings, Inc. v. Arce Engineering Co., Inc.*, 89 F.3d 1548, 1556 (11th Cir. 1996).

5. In fact, a recent decision from another district court in Minnesota appears to have chosen the "*Altai* test," the prevailing test used in many jurisdictions and discussed in greater detail later in this article. See *Systems, Inc. v. Softwares, Inc.*, 2004 U.S. Dist. LEXIS 6001 at *32-33 (D. Minn. March 29, 2004).

6. See e.g., *eScholar, LLC v. Otis Educ. Sys.*, 2005 U.S. Dist. LEXIS 40727 (SDNY Nov. 3, 2005).

7. See e.g., *MiTek Holdings, Inc. v. Arce Engineering Co.*, 89 F.3d 1548, 1555-56 nn.15-16 (11th Cir. 1996) (commenting, approvingly and adopting same test); *Engineering Dynamics, Inc. v. Structural Software, Inc.*, 26 F.3d 1335, 1343 (5th Cir. 1994) (adopting *Altai's* segmentation method); *Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, 9 F.3d 823, 834 (10th Cir. 1993) (adopting the abstraction-filtration-comparison test of *Altai*). See *eScholar, LLC v. Otis Educ. Sys.*, 2005 U.S. Dist. LEXIS 40727 (SDNY Nov. 3, 2005).

8. This "abstractions test" was formulated in *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930), cert. denied 282 U.S. 902 (1931) to evaluate which aspects of a play constitute copyrightable expression and which aspects constitute uncopyrightable idea.

9. This doctrine is applied primarily to fictional works such as novels or dramas, and withholds copyright protection from incidents or character traits preordained by the work's underlying idea. See e.g., *Walker v. Time-Life Films, Inc.*, 784 F.2d 44, 50 (2d Cir. 1985).

10. *Computer Associates International, Inc. v. Altai, Inc.*, 982 F.2d at 710, referencing Nimmer on Copyright §13.03[F][5].

11. There are other ways to avoid infringement claims that may be used independently or in conjunction with clean room procedures, which include, among others, written representation and warranties from the software developer and other indemnification agreements with other third parties.

12. See Phil Albert, "Clean-Room Development Avoids Copyright Battles," LinuxInsider.com (May 18, 2004), available at: http://www.technewsworld.com/story/33839.html.