Chips, Locks, and Video Games

Courts Rule on the Scope of Protection in Computer Copyright Cases

Three leading federal appellate courts have signalled a shift narrowing the scope of copyright protection in computer programs

by David R. Ellis

In the past year, three leading federal appeals courts have announced important rulings on the scope of protection for computer programs in copyright infringement cases. One of the cases, decided by the Second Circuit Court of Appeals in New York, was a landmark opinion on standards for determining infringement in computer software cases. The other two cases, decided by the Federal Circuit in Washington, D.C., and the Ninth Circuit in California, considered whether the copying of copyrighted computer programs in the course of reverse engineering video games constitutes copyright infringement or is protected under the doctrine of "fair use."

Computer Associates International v. Altai

This case, *Computer Associates International, Inc. v. Altai, Inc.*, 982 F.2d 693 (2d. Cir. 1992), was decided in New York City in June 1992. Computer Associates International (CAI), a software developer and distributor, sued a competitor, Altai, for copying substantial portions of CAI's job scheduling program for IBM mainframe computers. CAI claimed that Altai infringed CAI's copyrights and misappropriated CAI's trade secrets in the software.

Altai conceded that one of its employees had copied from a portion of CAI's program, but said that it was done without anyone's knowledge at Altai. Altai further claimed that once it was alerted to the copying and had confirmed it, Altai rewrote the program to eliminate any of the infringing material. One of the principal points at issue was whether Altai had been successful in its attempted "cleanup" of the software.

According to Altai, as soon as it learned that one of its programmers had copied some of CAI's code, Altai immediately set out to rewrite the program so as to eliminate the copied code. It did this by analyzing the way the program functioned and then having eight programmers who had not seen the original code write new code based on descriptions of how various portions of the program functioned. The rewrite took about six work months.

In determining whether Altai's new code infringed CAI's program, the trial judge tried to decide whether there was "substantial similarity" of protected copyrightable expression. Copyright does not protect ideas—the way a program functions—but only the *expression* of those ideas in a particular way. The original copying of code by Altai's programmer was clearly an infringement of CAI's expression, but was the rewritten code, which was based on an analysis of how CAI's program worked, also an infringement?

In deciding the question whether Altai's rewritten code infringed CAI's copyrights, the court considered an important previous case, *Whelan Associates v. Jaslow Dental Lab*, 797 F.2d 1222 (3d Cir. 1986), decided by a federal appeals court in Philadelphia in 1986. The *Whelan* court adopted a very broad view of copyrightability for computer software, saying that the purpose or function of a useful work, like a computer program, is its idea and "everything that is not necessary to that purpose or function would be part of the expression of the idea." On that basis, the *Whelan* court found that protectable expression included not only the program code, but also extended to the "structure, sequence and organization" of the program. This expansive reading has led some courts and commentators to say that the "look and feel" of a computer program is protected by copyright.

In *CAI*, the Second Circuit noted that the *Whelan* rule had received a mixed reception in the courts and had been roundly criticized in the academic community as inadequately distinguishing ideas from expression. The court agreed, saying that the *Whelan* approach relied too heavily on metaphysical distinctions and not enough on practical considerations in computer programming. It thus declined to follow its rationale and instead adopted a three-step procedure, adapted from an earlier "abstraction" test first proposed by Judge Learned Hand in 1930 in a case involving dramatic works. *Nichols v. Universal Picture Corp.*, 45 F.2d 119 (2d Cir. 1930). This test distinguishes between protectable ex-

pression and unprotectable ideas by looking at an abstract continuum of increasing generality. In a play or movie, this increasing generality would be from dialogue, to incident, to plot, to theme, to ideas. In a computer program, it would be from object code, to source code, to parameter lists, to services required, to general outline.

The three-step test proposed by the *CAI* court is as follows: Step One: Abstraction. Dissect the allegedly copied program's structure and isolate each level of abstraction, beginning with the code itself and ending with the program's ultimate function and purpose. Step Two: Filtration. Examine the structural components at each level of abstraction to determine whether it is essentially an idea, dictated by efficiency or external factors, taken from the public domain (hence unprotectable), or an independently created expression (hence protectable by copyright). Step Three: Comparison. After a court has filtered out the unprotectable matter, there may remain a core of protectable expression. Infringement results when the defendant's program is substantially similar to the plaintiff's protected expression; that is, the defendant copied a relatively important amount of the protected expression in the plaintiff's program so that they are substantially similar.

On the basis of its application of the three-step test to the facts of this case, the court agreed with the trial judge that Altai had succeeded in cleaning up its program and eliminating the infringing similarities. There were virtually no lines of code that were identical and many, if not most, of the parameter lists and macros were dictated by the functionality of the program. Similarly, a great deal of the list of services required was attributable to the demands of functionality and was extensively determined by the demands of the operating system and the applications program to which it was to be linked. The court thus agreed that Altai's new version of the program was not "substantially similar" to CAI's program and affirmed the decision that Altai did not infringe CAI's copyrights.

This case is likely to prove important in computer software copyright infringement cases because it narrows the scope of copyright protection for nonliteral components of computer programs. Originally, only the literal code was thought to be protected, but *Whelan* and cases that followed it provided for greater protection, extending to the "structure, sequence and organization" of the program. The Second Circuit has now cut this back to elements traditionally considered protectable expression rather than allowing it to cover areas more commonly considered unprotectable ideas. Programmers thus will not be able to obtain exclusive rights over broad methods of operation of programs but only over their own particular original contributions, thus allowing and indeed encouraging future innovations by developers in the computer software industry.

Atari v. Nintendo

In *Atari Games Corp. v. Nintendo of America, Inc.*, 975 F.2d 832 (Fed. Cir. 1992), the Federal Circuit decided an appeal in a dispute between two rival manufacturers of home video games concerning the scope of the fair use doctrine in computer copyright cases. Specifically, the case raised the issue of whether Atari's copying of Nintendo's copyrighted computer programs in the course of reverse engineering a lockout microprocessor chip constituted copyrighted infringement or was protected as fair use. The court's decision in September 1992, together with the decision of the Ninth Circuit a month later in *Sega Enterprises, Ltd. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), is expected to give the software industry guidance in determining whether certain types of reverse engineering of software programs is permissible under the copyright laws.

Nintendo is the manufacturer of a popular home video game system, the Nintendo Entertainment System (NES). The NES includes a monitor, a console that accepts game cartridges, and controls. Nintendo designed a program—a lockout program—to prevent the NES from accepting unauthorized game cartridges. When a user inserts an authorized cartridge into the console, a microprocessor chip in the console detects a coded message in a chip in the cartridge and accepts the cartridge. When a user inserts an unauthorized cartridge, the console does not detect an unlocking message and refuses to operate the cartridge, thus denying access to the user.

Atari wanted to "unlock" Nintendo's lockout program in order to sell game cartridges that could be played on the NES. Therefore, Atari attempted to analyze and replicate the NES security system by using reverse engineering techniques. First, Atari tried to break the NES program code by monitoring the communication signals between the console and cartridge chips. Next, Atari tried to decipher the code by chemically peeling layers from the NES chips to allow microscopic examination of the machine-readable object code. Still, Atari could not read the code sufficiently to replicate the NES security system.

Atari then had its attorney obtain a copy of the human-readable source code of the NES chip program from the U.S. Copyright Office by falsely representing that Atari was a defendant in an infringement case in California involving the program and that Atari would use the copy of the program only in connection with that litigation. Using this source code, Atari was able to decipher the NES program and develop its own program—the Rabbit program—to unlock the NES. Because Atari chose a different microprocessor and programming language to implement its Rabbit program, the line-by-line instructions of the NES and Rabbit programs were different. Nonetheless, the Rabbit program generated signals functionally indistinguishable from the NES program, thereby allowing owners of Atari cartridges to play games on Nintendo's system.

Nintendo filed suit against Atari, alleging among other things that Atari infringed Nintendo's copyrights in the NES program by making "intermediate" copies of the program code during the course of its reverse engineering of the NES. The district court entered a preliminary injunction against Atari, 18 U.S.P.Q. 2d 1935 (N.D. Cal. 1991). Atari appealed to the Court of Appeals for the Federal Circuit, which had jurisdiction because the case also included patent infringement claims.

On appeal, the Federal Circuit first determined whether Atari had impermissibly copied protectable expression from Nintendo's program. The U.S. Supreme Court recently reaffirmed a core concept of copyright law that copyright protects only original *expression*, not ideas, facts, processes, or methods of operation. *Feist Publications v. Rural Telephone*, 111 S. Ct. 1282 (1991); 17 U.S.C. §102(b). In addition, the

court cited with approval the decision of the Second Circuit in *Computer Associates International v. Altai, Inc.*, and its three-step "abstraction" test for distinguishing unprotectable ideas from protectable expression.

On the basis of this test, the Federal Circuit found that Nintendo's program contained original protectable expression that had been designed to implement a lock and key function for the NES console. The court found original creative elements that were not dictated by external factors or the security function itself and that therefore were protectable by copyright. The court then considered whether Atari's intermediate copying of the code in the course of reverse engineering the program could be considered a fair use.

The fair use doctrine is an exception to the copyright owner's exclusivity in an original work. Section 107 of the Copyright Act, 17 U.S.C. §107, provides that copying which might otherwise be an infringement of the owner's exclusive rights may be permitted for certain purposes such as criticism, comment, news reporting, teaching, scholarship, and research. Since copyright protects only expression, not underlying ideas, the Copyright Act permits those in rightful possession of a copy of a work to undertake necessary efforts to understand the work's ideas, processes, and methods of operation. Moreover, Congress indicated in enacting the Copyright Act that the fair use doctrine should be adapted to accommodate technological innovations. H.R. Rep. No. 1476, 94th Cong., 2d Sess. 66 (1976).

Atari had made intermediate copies of the Nintendo program in two different settings. Before obtaining the Copyright Office's copy of the program, Atari chemically stripped some chips and copied portions of the object code from the chips. After obtaining the source code from the Copyright Office, Atari made other intermediate copies of the program such as photocopies of the Copyright Office copy. The district court found this latter copying to be an infringement.

Atari, however, maintained that copying for the purpose of reverse engineering Nintendo's microprocessor program was a fair use. The appeals court agreed that intermediate copying for the purpose of understanding the ideas and processes of a copyrighted program can be a fair use, depending on the nature of the work. According to the court, "[a]n individual cannot even observe, let alone understand, the object code on Nintendo's chip without reverse engineering." Thus, reverse engineering of the object code to discern the underlying unprotectable ideas in the program can be a fair use so long as it is limited to an amount necessary to understand the unprotected elements of the work: "Any reproduction of protectable expression must be strictly necessary to ascertain the bounds of protected information within the work. . . . Any copying beyond that necessary to understand the [NES] program [would be] infringement. Atari could not use reverse engineering as an excuse to exploit commercially protected expression."

The court thus ruled that Atari's

intermediate copying in the course of reverse engineering could have qualified as a fair use. However, fair use is an equitable doctrine, an "equitable rule of reason" according to the U.S. Supreme Court, *Sony Corp. of America v. Universal City Studios*, 464 U.S. 417, 448 (1984) (the "Betamax" case), to which courts apply traditional principles of "good faith" and "fair dealing." *Harper & Row v. Nation Enterprises*, 471 U.S. 539, 562-63 (1985). Here, Atari had not come into court with clean hands, because its attorney had purloined a copy of the NES program from the Copyright Office by falsely representing that Atari was already involved in litigation when in fact it was not.

Thus, the copy of the source code that Atari had used to reverse engineer Nintendo's program was tainted. To invoke the fair use defense, the accused infringer must possess an authorized copy of the copyrighted work. Because Atari was not an authorized possessor of the Copyright Office's copy of the program, the copying by Atari, even in aid of reverse engineering, did not qualify as a fair use.

The Federal Circuit thus affirmed the district court's finding of infringement against Atari, its rejection of the fair use defense, and the issuance of a preliminary injunction. As a result, the following conclusion can be drawn from this case: Intermediate copying of computer programs for the purpose of reverse engineering *can* be a fair use, but only if the use is truly *fair* in the light of traditional standards of equity and the principles of copyright law as embodied in the idea/expression dichotomy and the criteria established by §107 and the applicable case law.

Sega v. Accolade

In *Sega Enterprises, Inc. v. Accolade, Inc.*, the Ninth Circuit also considered whether reverse engineering of a video game console for the purpose of developing compatible games cartridges could be a fair use under the Copyright Act. Sega is a manufacturer of home video game systems, including the "Genesis" console and video game cartridges. Accolade is an independent manufacturer and marketer of computer games that are compatible with Genesis and other computer systems.

Although Sega licenses its copyrighted computer code to other developers to design Genesis-compatible games, Accolade elected not to take a license from Sega. Instead, Accolade reverse engineered Sega's video game programs to discover the requirements that would make its games compatible with Genesis. Accolade used a process called "disassembly" or "decompilation" to transform the machine-readable object code in Sega's commercially available game cartridges to human-readable source code. Accolade analyzed the resulting code to determine how it worked, and then created a functional description of the program's requirements that did not include any of Sega's code.

Accolade then developed its own games for Genesis by using the information in the functional description to create a new computer code. Accolade subsequently made modifications in the code to enable its games to play on the most recent version of Sega's system, the "Genesis III." Accolade's new code included a small segment copied from Sega's code that caused a trademark message to be displayed on the user's screen that read "PRODUCED BY OR UNDER LICENSE FROM SEGA ENTERPRISE LTD." This message, of course, was not true with regard to Accolade's games.

Sega sued Accolade for trademark infringement and also claimed that Accolade's disassembly of its computer program was copyright infringement. The district court ruled for Sega on both theories and issued a preliminary injunction against Accolade. The injunction barred Accolade from disassembling, using, or modifying Sega's copyrighted code or producing any Genesis-compatible games using that code, including in particular the code that prompted the trademark message. Accolade appealed the ruling to the Ninth Circuit Court of Appeals.

On appeal, Accolade contended that its disassembly of the object code was a fair use under §107 of the Copyright Act, 17 U.S.C. §107. Accolade argued, and the court agreed, that if disassembly is a necessary step for a user to take in order to examine and study the unprotected ideas and functional concepts of a program, then it can be a fair use. That is, if the only method of gaining access to the unprotected aspects of a program is to disassemble the code, and the user has a legitimate interest in gaining access, then it is a fair use to do so under §107. Here Accolade had a legitimate interest in learning how to make game cartridges compatible with the Genesis console.

In reaching its conclusion, the court examined the four statutory factors that are expressly set forth in §107 to assist courts in determining whether a particular use is a fair use:

1) The first statutory factor to con-

sider is "the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes." Here the court recognized that Accolade's use was for a commercial purpose—producing a competing product. However, the actual use was only intermediate (copying of the code through disassembly), so that the *ultimate* commercial use was indirect or derivative. Accolade copied Sega's code for the purpose of studying its unprotectable ideas and functions so it could modify existing games and make them compatible with the Genesis console. Since no other method of studying Sega's code was available to Accolade, the court concluded that Accolade's copying of the code was for a legitimate, essentially nonexploitative purpose, and that the commercial aspect of the use was therefore of minimal significance.

2) As to the second statutory factor, "the nature of the copyrighted work," the court pointed out that the work, consisting of computer programs, was essentially functional or utilitarian in nature. Copyright protects *expression* and not *ideas*, and Accolade copied Sega's protected expression in attempting to determine the software's underlying ideas. But the court found that the only practical way for Accolade to understand the functional requirements to make its games compatible with Sega's console was to disassemble the code in Sega's game cartridges.

If disassembly of copyrighted program code were per se an unfair use, said the court, the owner of the copyright would gain a monopoly over the functional aspects of the work—the unprotected idea, as well as the protected expression—which is contrary to the scheme of the Copyright Act. Therefore, since the only way Accolade could examine these unprotected ideas was by copying the protected expression in the code, the court found this factor as weighing in favor of Accolade's claim of fair use.

3) As to the third statutory factor, "the amount and substantiality of the portion used in relation to the copyrighted work as a whole," the court noted that Accolade had disassembled entire programs written by Sega, which would tend to weigh against a finding of fair use. However, the court said that copying an entire work does not necessarily preclude fair use (citing the decision in *Sony*, 464 U.S. 417, 448-50 (1984)), in which entire videocassettes were copied, but fair use was nonetheless found). In fact, said the court, where the ultimate use is a limited one, as it was here (to understand the unprotected ideas) this factor is of very little weight.

4) The fourth statutory factor is "the effect of the use upon the potential market for or value of the copyrighted work." Here the court felt that this factor bears a close relationship to the first factor, the "purpose and character" inquiry, in that there is a distinction between copying a work to facilitate the making of an independent creative work as opposed to simply exploiting another's creative efforts.

The court believed that Accolade was not usurping Sega's potential market in selling computer games but was legitimately competing in the market for Genesis-compatible games by making available additional games to those who might wish to buy a variety of games to play on their Genesis systems. Therefore, even if Sega's market was somewhat affected, this still would not give it the right to use the Copyright Act to foreclose competition in its market. Thus, this factor weighed in Accolade's favor, notwithstanding any minor economic loss Sega might thereby suffer.

Having weighed the statutory fair use factors, the court concluded that "where disassembly is the only way to gain access to the ideas and functional elements embodied in a copyrighted computer program and where there is a legitimate reason for seeking such access, disassembly is a fair use of the copyrighted work, as a matter of law." The court thus reversed the district court's grant of a preliminary injunction in favor of Sega on its claim of copyright infringement, thereby reaching a result similar to that of the Federal Circuit in the *Atari* case the previous month.

Conclusion

In these cases, the three leading federal appellate courts most concerned with copyright and intellectual property law have signalled a shift narrowing the scope of copyright protection in computer programs. In *Computer Associates International*, the Second Circuit receded from the broad protection given in such cases as *Whelan*, which some commentators said gave almost patent-like protection to ideas embodied in the structure, sequence, and organization of a program. In *Atari* and *Sega*, the Federal Circuit and Ninth Circuit not only gave their approval to the Second Circuit's formulation for separating protectable expression from unprotectable ideas, but also indicated that reverse engineering of a computer program in order to understand its underlying function and purpose could be viewed as a legitimate and fair use. Taken together, these cases indicate that courts will continue to apply copyright law to protect original expression in computer software, but that they will scrutinize the facts of each case carefully to make sure that software developers do not overreach by attempting to use the Copyright Act to monopolize competitive markets by obtaining protection for broad ideas rather than the particular original expressions of their programs. ❐

AUTHOR

David R. Ellis, Largo, practices computer law, corporations, contracts, copyrights, trademarks and trade secrets, entertainment and arts law, and franchise and licensing law. He is a graduate of M.I.T. and Harvard Law School, and the author of A Computer Law Primer. *Mr. Ellis is an adjunct professor at Stetson College of Law, teaching intellectual property and computer law.*

This column is submitted on behalf of the Computer Law Committee, Christopher K. Caswell, chair.