UIC John Marshall Journal of Information Technology & Privacy Law

Volume 12
Issue 1 *Computer/Law Journal - Fall 1993*

Article 1

Fall 1993

The Legality of Disassembly of Computer Programs, 12 Computer L.J. 1 (1993)

David L. Hayes

THE LEGALITY OF DISASSEMBLY OF COMPUTER PROGRAMS[†]

by DAVID L. HAYES*

I. INTRODUCTION

This article analyzes two very important decisions from federal Court of Appeal holding that disassembly of a computer program in the course of reverse engineering that program constitutes a fair use under the copyright laws, at least under many circumstances.

The second of these decisions also holds that when there is no other way to gain access to a computer than to copy an "initialization" code into a program designed to run on that computer, then it is permissible to copy such initialization code, even if the result of the code is to cause a false trademark message to appear on the screen of the computer stating that the program is licensed by the manufacturer of the computer.

These two decisions, *Sega Enterprises Ltd. v. Accolade, Inc.*[1] and *Atari Games Corp. v. Nintendo of America, Inc.*[2] constitute part of what appears to be a growing trend in the courts to limit the scope of copyright protection for computer programs, particularly as it relates to "functional" aspects of programs. Both decisions recognize that computer programs, being inherently utilitarian works, are of a unique nature among copyrighted works and therefore require great care in drawing the line between protectable and unprotectable elements to ensure that underlying ideas and functions of the program are not protected.

Although these decisions leave much latitude for disassembly of computer programs under the fair use doctrine, both decisions require that such disassembly be *necessary* to gain access to unprotected elements of the program. The decisions leave open uncertainty as to what circumstances render disassembly "necessary" and therefore lawful, and the *Sega v. Accolade* decision notes some specific examples of cir-

[†] Copyright © 1992 Fenwick & West. All rights reserved.

* Mr. Hayes is with the Palo Alto, California office of Fenwick & West.

[1] 1992 U.S. App. LEXIS 26645 (October 20, 1992).

[2] 24 U.S.P.Q.2d (BNA) 1015 (1992). These two opinions were issued within about five weeks of each other.

cumstances in which disassembly would not be "necessary" to understand the functions of a program. Thus, persons contemplating disassembly of computer programs—such as in the course of a clean room development—should still consult counsel in advance to determine whether the proposed disassembly would qualify as a fair use under the copyright laws.

II. THE *SEGA V. ACCOLADE* DECISION

A. Background of the Case

Sega markets a home video entertainment system known as the "Genesis," which contains a microprocessor capable of executing video game computer programs contained in game cartridges that are inserted into the Genesis. Although Sega licenses developers to create games for the Genesis, Accolade desired to develop cartridges that would be Genesis-compatible without taking a license and paying royalties to Sega.

In order to develop cartridges that would execute on the Sega Genesis, Accolade disassembled several commercially available copies of Sega's game cartridges and produced printouts of the resulting disassembled code. Accolade engineers studied the printouts and created a technical specification describing the functioning of, and the interface to, the Genesis console. Accolade programmers then used the technical specification to develop Genesis-compatible video games that Accolade marketed and sold.

In 1991 Accolade learned of an impending release by Sega of a new version of the Genesis called the "Genesis III." At a consumer electronics show, a demonstration of a prototype Genesis III revealed that Accolade's game cartridges would not execute on it. After learning of this fact, Accolade performed further disassembly of Sega game cartridges and discovered a small segment of twenty to twenty-five bytes of code that was included in the "power up" sequence of every Sega game. Accolade suspected that this segment of code was required to enable a game cartridge to execute on the forthcoming Genesis III and inserted this code into all of its games.

As Accolade soon discovered, the code was indeed required to make a game cartridge execute on the Genesis III. Sega had added a feature called the "Trademark Security System" (TMSS) to the Genesis III that caused it to look for this code (the "TMSS initialization code"). If the Genesis III found the TMSS initialization code in an inserted game cartridge, it would execute the game and cause a trademark message to appear on the user's monitor reading "PRODUCED BY OR UNDER LICENSE FROM SEGA ENTERPRISES LTD" (the "Sega Message"). If the initialization code was absent, the game would not run. Accolade did not discover until after the Genesis III release that the TMSS initialization code Accolade had inserted into all of its games caused the Sega Message—which was false as applied to Accolade's games—to appear on the screen.

Sega filed suit against Accolade, alleging that Accolade's intermediate copying of disassembled Sega game code constituted copyright infringement, and that the display of the false Sega Message by Accolade's games constituted trademark infringement.

The district court ruled for Sega on both theories. The court found that Accolade's intermediate copying of disassembled Sega game code was a copyright infringement and that, because of such intermediate infringement, all of Accolade's games developed using the results of the disassembly were infringing based implicitly on a "fruit of the poisonous tree" notion.

The district court also ruled for Sega on its trademark claim. At the hearing on Sega's motion for a preliminary injunction, Sega demonstrated two game cartridges that allegedly contained modifications enabling those cartridges to run on the Genesis III but not produce the Sega Message. Sega refused to allow Accolade's engineers to examine the modified cartridges or to reveal the manner in which the cartridges had been modified. Based on this evidence, the district court concluded that the TMSS initialization code was not necessary to make the Genesis III execute game cartridges, and that Accolade could therefore not assert a "functionality" defense[3] to Sega's claim of trademark infringement.

The district court entered a preliminary injunction against Accolade enjoining it from distributing its game cartridges, and Accolade appealed to the Ninth Circuit.

B. Copyright Arguments of Accolade Rejected by the Court

On appeal, Accolade made four principle arguments as to why its disassembly should not be considered a copyright infringement. The court rejected three of these arguments, but accepted the fair use defense. The court's analysis of the three arguments it rejected contains several important points of copyright law, and that analysis is therefore summarized first before turning to the fair use analysis.

1. *Intermediate Copying*

Accolade first argued that intermediate copying does not constitute copyright infringement unless the final end product of the copying is substantially similar to the copyrighted work. The court rejected this argument, concluding that the literal language of the copyright statute

[3] The "functionality" doctrine of trademark law is discussed further below.

does not distinguish between unauthorized copies of a copyrighted work on the basis of what stage of the alleged infringer's work the unauthorized copies represent.

Before the *Sega v. Accolade* decision, there were a number of decisions in various jurisdictions which, although not squarely raising the issue of whether intermediate copying is an infringement if the final product is not substantially similar, suggested contrary results. Some decisions, including some from the Ninth Circuit, contained language suggesting that intermediate copying is *not* an infringement if the final product is not substantially similar.[4] Other decisions, including at least one district court decision within the Ninth Circuit's jurisdiction, contained language suggesting the opposite.[5]

The court in *Sega v. Accolade* held, however, that in none of these cases was the question of the lawfulness of intermediate copying squarely raised, and therefore concluded that the question was one of first impression. The court held that intermediate copying of computer program object code is a literal copyright infringement regardless of whether the end product of the copying also infringes.

This is an important ruling for its implication for cases in which one has set out to create a work that starts in whole or in part from the copyrighted work of another. The *Sega* decision suggests that one may not avoid infringement simply by making enough changes to the copyrighted material of another so that the end product is no longer similar, if one has started out by copying in whole or in part. Thus, if a company (or a court) were to determine or discover that a part of its product were too similar to, or had been copied from, the copyrighted expression of another, the company could not necessarily avoid a claim of infringement simply by making sufficient changes to remove the similarities.

2. *The Idea/Expression Distinction*

Accolade's second argument on appeal was that disassembly of object code should be considered lawful per se, because it is necessary in order to gain access to the ideas and functional concepts embodied in object code, which ideas and functions are not protected by copyright under § 102(b) of the copyright statute.

The court rejected this argument, however, both because object code should be entitled to the full range of copyright protection, and because of the court's belief that the "ideas and functional concepts underlying many types of computer programs, including word processing

[4] *See, e.g., See v. Durang*, 711 F.2d 141 (9th Cir. 1983).

[5] *See, e.g., Atari Games Corp. v. Nintendo of America, Ind.*, 18 U.S.P.Q.2d 1935 (N.D. Cal. 1991).

programs, spreadsheets, and video game displays, are readily discernible without the need for disassembly, because the operation of such programs is visible on the computer screen. The need to disassemble object code arises, if at all, only in connection with operations systems, system interface procedures, and other programs that are not visible to the user when operating—and then only when no alternative means of gaining an understanding of those ideas and functional concepts exists."[6]

The court thus refused to establish a per se rule that disassembly of object code is always lawful, preferring instead to treat the analysis on a case by case basis under the fair use doctrine. The court's attempt to draw a distinction between situations in which disassembly of object code is necessary from those in which it is not is confusing, for the video game programs that Accolade disassembled certainly were "visible on the computer screen," yet the court found the disassembly lawful. The extent to which the court's observations on this point limit the scope of the right to disassemble is discussed in further detail below.

3. *Section 117*

Accolade's third argument was that disassembly is permitted by § 117 of the copyright statute, which allows the lawful owner of a copy of a computer program to copy or adapt the program if the new copy or adaptation "is created as an essential step in the utilization of the computer program in conjunction with a machine and . . . is used in no other manner." Section 117 was enacted in part to make clear that the making of a copy of a computer program in memory in order to execute it does not constitute an infringement. The court, with very little discussion, rejected this argument, noting simply that § 117 "does not purport to protect a user who disassembles object code, converts it from assembly into source code, and makes printouts and photocopies of the refined source code version."[7]

C. The Court's Analysis of the Fair Use Doctrine

Having rejected Accolade's first three arguments, the court turned to Accolade's fair use argument. Section 107 of the copyright statute lists four factors to be considered in determining whether a particular use is a fair one:

> (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

[6] 1992 U.S. App. LEXIS 26645 at 10.

[7] *Id.*

(4) the effect of the use upon the potential market for or value of the copyrighted work.

1. *The First Factor*

Turning to an analysis of the first factor—the purpose and character of the use—the court rejected Sega's argument that because Accolade had copied its object code to produce a competing product (a commercial purpose), there could not be a fair use. The court noted that, although the copying was for an ultimate commercial purpose to make a competing product, the *direct* purpose of the copying was to discover the functional elements of Sega's games necessary for compatibility with the Genesis console—aspect of Sega's games that are not protected by copyright. The court noted strong public policy reasons for permitting Accolade to create compatible games in order to increase the number of independently designed video game programs for use with the Genesis console. Accordingly, the court concluded that the first fair use factor favored Accolade.

The court's grounding of its holding on public policy considerations of competition is an important one. Previous decisions had held that if copying is otherwise infringing, the fact that it is done in the name of compatibility is not a defense.[8] The *Sega* court's recognition that creation of compatible products by competitors is something the copyright law in general should promote may tend to influence other courts to grant a narrower scope of copyright protection in future cases in which a fair use defense is asserted for disassembly to create compatible products.

2. *The Second Factor*

Under the second fair use factor—the nature of the copyrighted work—the court noted that computer programs pose unique problems because of their essentially utilitarian nature. The court observed that, unlike most types of works in which the unprotected ideas are readily apparent from reading or examination of the work itself, "humans

[8] *See, e.g., Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240, 1253 (1983) ("Franklin may wish to achieve total compatibility with independently developed application programs written for the Apple II, but that is a commercial and competitive objective which does not enter into the somewhat metaphysical issue of whether particular ideas and expressions have merged."), *cert. dismissed*, 464 U.S. 1033 (1984). The *Apple* case presented a more compelling case for a finding of infringement because the defendant had copied literal *code* of the operating system of the plaintiff and claimed that such copying was necessary to achieve compatibility. Nevertheless, the *Apple* court's comments about the irrelevance of compatibility seem to run contrary to the public policy notions informing the court's treatment of compatibility in *Sega v. Accolade.*

often cannot gain access to the unprotected ideas and functional concepts contained in object code without disassembling that code."[9]

The court rejected Sega's argument that disassembly was not the only available method for gaining access to the interface specifications of the Genesis console. Sega had argued that at least two alternative methods of reverse engineering other than disassembly were available to Accolade: "peeling" the microchips contained in the Genesis to study their structure and design, and "clean room" procedures. The court noted that, at most, peeling would enable Accolade to examine the transistors of the chip to determine the object code it contained, but that it would still be necessary to write that object code down on paper if it were to be useful—which would require the making of a copy of that code. Moreover, the court noted that disassembly would still have been necessary even in a clean room development.

In sum, the court held that disassembly of object code in Sega's video game cartridges "was necessary in order to understand the functional requirements for Genesis compatibility. The interface procedures for the Genesis console are distributed for public use only in object code form, and are not visible to the user during operation of the video game program."[10] The court, again borrowing from public policy principles, noted that, because disassembly is required for humans to understand object code, "[i]f disassembly of copyrighted object code is per se an unfair use, the owner of the copyright gains a de facto monopoly over the functional aspects of his work"—a monopoly that should be available only under the patent laws.[11] The court therefore concluded that the second fair use factor favored Accolade.

3. *The Third Factor*

The court held that the third factor—the amount copied—favored Sega because the entire Sega code had been disassembled by Accolade. The court held that this factor should carry little weight in this circumstance, however, because the ultimate use of the entire code was limited essentially to understanding its functionality.

4. *The Fourth Factor*

Finally, the court held that the fourth fair use factor—the effect on the potential market for the copyrighted work—favored Accolade because Accolade's games did not *supplant* the market for Sega's games in view of the fact that video game purchasers typically purchase more than one game. The court rejected Sega's argument that it is sufficient

[9] 1992 U.S. App. LEXIS 26445, at 16.

[10] *Id.* at 17.

[11] *Id.*

to rebut fair use if the use enables a copier to enter the market for works of the same *type* as the copied work. Rather, the new work must supplant the direct market for the particular copied work.

The court's ruling with respect to the fourth fair use factor is important. It will generally always be the case that a compatible or competitive product will potentially reduce the market for the original product merely by offering a competitive alternative. Thus, if Sega's argument were accepted that producing a competitive alternative is of itself sufficient to rebut a finding of fair use, then fair use would, as a practical matter, probably never be available in cases of disassembly. The *Sega* court rejected this approach to the fair use doctrine, however. Again invoking public policy, the court stated that "an attempt to monopolize the market by making it impossible for others to compete runs counter to the statutory purpose of promoting creative expression and cannot constitute a strong equitable basis for resisting the invocation of the fair use doctrine."[12] This language will be of assistance to competitors in other situations who disassemble in order to create compatible products and wish to invoke the fair use doctrine.

D. THE TRADEMARK ISSUE

The facts of the *Sega* case presented a uniquely interesting trademark issue. As the court characterized the issue, "both parties agree that there is a misuse of a trademark, both agree that there is unlawful mislabeling, and both agree that confusion may result. The issue, here, is—which party is primarily responsible? Which is the wrongdoer—the violator?"[13]

The court held that Sega should be held responsible for the false labeling created by the Sega TMSS when Accolade cartridges are executed in the Genesis because such false labeling was the result of a deliberate decision by Sega to include a TMSS device that would both limit general access to the Genesis and cause false labeling. The court held that Sega should have foreseen that a competitor might discover how to utilize the TMSS, and that when it did and included the TMSS initialization code in its cartridges, a false trademark message would be produced. Moreover, the court held that Sega had misused its trademarks by using them to serve to limit competition in the manufacture and sale of a product.

The court rejected Sega's argument that the TMSS is not a "functional" use of its trademarks. Under trademark law, "functional" features of products cannot be protected. The court noted that functional features are those "which constitute the actual benefit that the con-

[12] *Id.* at 15.

[13] *Id.* at 19.

sumer wishes to purchase, as distinguished from an assurance that a particular entity made, sponsored, or endorsed a product."[14]

Sega argued that the TMSS was not "functional" because Accolade could have "engineered around" the TMSS and caused the false Sega Message not to appear while still enabling its cartridges to execute on the Genesis III. As proof of this fact, Sega submitted the affidavit of one of its engineers stating that cartridges can be altered to execute on the Genesis III but not produce the Sega Message, although the engineer did not reveal how the alterations were to be made.

The court rejected Sega's position, noting that it was indisputable that part of "the actual benefit that the customer wishes to purchase" is compatibility with the Genesis III console. Because the TMSS initialization code provides that compatibility, the court held it to be "functional" for trademark purposes. Moreover, the court held that at most the affidavit of Sega's engineer established that an individual familiar with the operation of the TMSS can discover a way to engineer around it, but did not establish that a competitor unfamiliar with the operation of the TMSS could do so. The court held that Sega must prove that knowledge of an alternate method for gaining access to the Genesis III exists or is *readily available* to knowledgeable persons in the industry. Sega had submitted no such proof.

In sum, the court ruled, "Because the TMSS serves the function of regulating access to the Genesis III, and because a means of access to the Genesis III without using the TMSS initialization code is not known to manufacturers of competing video game cartridges, there is an insufficient basis for a finding of non-functionality."[15] The court noted that Sega might be able to produce sufficient evidence of alternative methods at trial, but it had failed to do so at the preliminary injunction stage.

The court's ruling on the trademark issue establishes an important precedent with respect to the use of elements of a computer program that might otherwise be protectable intellectual property primarily or solely to limit access to the program or to hardware on which the program runs. The logic of the decision might apply beyond the use of a trademark pattern as a "lock and key" mechanism (as in the TMSS system) to the use, for example, of a copyrighted header file or initialization code segment as a "lock and key" mechanism by requiring the presence of such file or segment, or the execution thereof, before a program can be executed on particular hardware. Because the *Sega* decision suggests that such uses may constitute misuse of the intellectual property, and others may be free to copy such property in any event,

[14] *Id.* at 23 (quoting *Vuitton et Fils S.A. v. J. Young Enterprises, Inc.*, 644 F.2d 769, 774 (9th Cir. 1981).

[15] 1992 U.S. App. LEXIS 26645, at 24.

companies considering the use of such techniques should consult counsel in advance.

II. THE *ATARI V. NINTENDO* DECISION

A. Background of the Case

Less than six weeks before the *Sega v. Accolade* opinion was issued, the U.S. Court of Appeals for the Federal Circuit issued a decision in *Atari v. Nintendo* that reached a very similar result. In fact, the court in the *Sega v. Accolade* decision acknowledged that its decision was consistent both with the analysis and the result of *Atari v. Nintendo*.

The facts of the *Atari v. Nintendo* case are very similar to those of *Sega v. Accolade*. Nintendo manufactures a home video game system known as the Nintendo Entertainment System (NES). The NES contains a computer program known as the "10NES" to prevent the NES from accepting unauthorized game cartridges. Atari is the manufacturer of NES-compatible game cartridges.

Atari attempted to decipher the 10NES program by monitoring the communication signals passed back and forth between the NES console and game cartridges, and by peeling the NES chips down layer by layer to allow microscopic examination of the object code of the 10NES.

After Atari had been unsuccessful using these two methods in deciphering the 10NES program, Atari's attorney obtained a copy of the source code of the 10NES program from the Copyright Office by falsely representing to the Copyright Office that Atari was presently a defendant in litigation in California involving the 10NES 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 10NES program and developed 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 10NES and Rabbit programs varied.

Nintendo filed suit against Atari, alleging, among other things, that Atari infringed Nintendo's copyrights in the 10NES program by making intermediate copies of the program during the course of its reverse engineering of the 10NES. The district court entered a preliminary injunction against Atari, and Atari appealed.

B. The Court's Analysis of the Fair Use Doctrine

Atari argued that its intermediate copying of the 10NES program in the course of its reverse engineering of the NES was permitted under the fair use doctrine. The court began its analysis of the issue by noting that the fair use doctrine generally "permits an individual in rightful possession of a copy of a work to undertake necessary efforts to understand the work's ideas, processes, and methods of operation."[16] With respect to computer programs in object code form, the court stated, "An author cannot acquire patent-like protection by putting an idea, process, or method of operation in an unintelligible format and asserting copyright infringement against those who try to understand that idea, process, or method of operation."[17]

The court did not engage in an extensive individual analysis of each of the four factors of the fair use doctrine, as did the *Sega* court. Instead, the court placed heavy emphasis on the second factor, the nature of the copyrighted work. "When the nature of a work requires intermediate copying to understand the ideas and processes in a copyrighted work, that nature supports a fair use for intermediate copying. Thus, reverse engineering object code to discern the unprotectable ideas in a computer program is a fair use."[18]

1. *Limits on the Scope of the Fair Use Doctrine*

The *Atari* court's language seems to suggest that reverse engineering of object code is *always* a fair use, because object code is of such a natnre that intermediate copying (in the form of disassembly) is required to understand the ideas and processes embodied therein.[19] A couple of paragraphs later in the opinion, however, the court places three limits on the fair use privilege as applied to object code.[20]

(1) Fair use does not extend to commercial exploitation of "protected expression." Thus, if a substantial amount of the disassembled code is copied into a product of the disassembler, there is probably infringement.

(2) "Any reproduction of protectable expression must be strictly necessary to ascertain the bounds of protected information within the work." The court did not elaborate further on what it meant

[16] 24 U.S.P.Q.2d at 1023.

[17] *Id.*

[18] *Id.*

[19] Moreover, language that appears later in the court's opinion suggests that the Atari court does not share the *Sega* court's conclusion that intermediate copying in the form of disassembly is a literal copyright infringement. The *Atari* court observed, "The district court assumed that reverse engineering (intermediate copying) was copyright infringement. [cite omitted] This court disagrees." 24 U.S.P.Q.2d at 1024. The court's language is confusing, however, for if intermediate copying by disassembly were per se lawful, the court would not have needed to engage in its fair use analysis. Moreover, the court's requirement, discussed below, that disassembly be "strictly necessary" to glean unprotected information from a computer program suggests that the court does not believe that disassembly of object code is *always* lawful, even when the disassembled code is not copied into the disassembler's product.

[20] *Id.* at 1023-24.

by this statement. The requirement that disassembly be "strictly necessary" may be akin to the *Sega* court's requirement that there be "no other means of access to the unprotected elements" of the program.[21] Unlike the *Sega* court, however, the *Atari* court did not suggest any specific examples of situations in which disassembly might not be "strictly necessary."

(3) "To invoke the fair use exception, an individual must possess an authorized copy of a literary work." The court concluded that, because Atari was not authorized under Copyright Office regulations to have possession of the copy of the 10NES source code that it got from the Copyright Office, Atari's dishonest representations to the Copyright Office in order to get the 10NES source code tainted *both* its use of the source code *and* its disassembly of Nintendo's object code, so that Atari could not invoke the fair use defense that would have otherwise been available for its disassembly.

2. *The Importance of "Unclean Hands"*

The court's holding that Atari's possession of the 10NES source code tainted all subsequent reverse engineering efforts is somewhat akin to the "fruit of the poisonous tree" notion invoked by the district court in its original decision in the *Sega v. Accolade* case. This suggests that courts will often be strongly influenced by "unclean hands" notions in deciding on the validity of a fair use defense in any particular case.

Thus, it is important that any disassembly be done only on a copy of the "targeted" computer program that is lawfully and rightfully in the possession of the disassembler. Companies engaging in reverse engineering should acquire copies of the targeted computer program on the open market, where possible, and should *not* engage in disassembly in circumstances that suggest "unclean hands," such as in violation of a confidentiality agreement with the supplier of the targeted program or where the copy of the targeted program has been gained in some surreptitious manner.

Many shrinkwrap and signed license agreements prohibit disassembly and reverse engineering of the licensed software. It is unclear whether such prohibition would be sufficient to cause one who disassembles in violation of the prohibition to have "unclean hands," for the legality or enforceability of such a prohibition under federal law is uncertain. At least one case suggests that such a prohibition is unenforceable under federal law.[22]

[21] 1992 U.S. App. LEXIS 26645, at 3.

[22] *See* Vault Corp. v. Quaid Software, Ltd., 847 F.2d 255, 270 (5th Cir. 1988).

III. CONCLUSION

Both the *Sega* and the *Atari* decisions provide a strong foundation for a fair use defense on the part of those who disassemble another's computer program object code in order to ascertain the ideas and processes contained therein. Each opinion is strongly grounded on notions of public policy. Each decision notes that, if disassembly is needed to understand the unprotected elements of a computer program, then if one were to hold that such disassembly constitutes copyright infringement, the copyright law would, in effect, provide the equivalent of a patent monopoly.

Each decision at various points couches the right under the fair use doctrine to perform disassembly in fairly strong, even absolute, terms. For example, the *Sega* court concluded its entire fair use analysis with the following statement: "We conclude 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 ones, disassembly is a fair use of the copyrighted work, as a matter of law."[23]

A careful reading of both decisions, however, raises several caveats to those who would perform disassembly. Each decision conditions the right to disassemble on some notion of necessity—the *Sega* decision requires that there be "no other means of access to the unprotected elements"[24] and the *Atari* decision requires that disassembly be "strictly necessary" to ascertain the bounds of protected information."[25]

It is unclear what the boundaries of this "necessity" doctrine will be, and it will undoubtedly give rise to much additional litigation in the future. Each of the courts seems bothered by the fact that disassembly gives one access to the creative expression of the programmer, and may therefore be a convenient vehicle for copying of all or a part of that expression. Accordingly, disassemblers should be extra cautious to ensure that no use of disassembled code beyond purely functional study is made. No disassembled code should be copied into a product without consulting counsel first, and the "default" assumption should always be that such copying is not permissible.

The *Sega* court suggests several specific instances in which disassembly is *not* necessary where the operation of the program is "visible." These examples are confusing, however, for the game programs disassembled in the case certainly produced much that was "visible," but the disassembly was nevertheless ruled a fair use, perhaps because the "security" aspects of the game cartridges that were the focus of Accolade's

[23] 1992 U.S. App. LEXIS 26645, at 19.

[24] *Id.* at 3.

[25] 24 U.S.P.Q.2d at 1024.

disassembly were an "invisible" (internal) *part* of the program's functioning.[26]

The *Atari* court offered no examples of any kind of the types of disassembly that it might consider not "strictly necessary." One may speculate, however, from the facts of the two cases that disassembly may be permissible in order to ascertain the processes and functions of a program that are purely internal to the program (such as a security lockout mechanism) and do not produce a visible effect to the user from which one could witness or infer their functioning. This conclusion also seems consistent with the *Sega* court's examples of the types of programs for which disassembly is necessary, such as "operations systems" and "system interface procedures."[27] "System interface procedures" in particular often constitute an "invisible" *portion* of the internal functioning of a program.

Conversely, where a company has published the details of an interface to its product, or the interface is ascertainable through some other means such as observation of the outputs of the program in reaction to various inputs, the fair use doctrine may not be available for further disassembly. A problem arises, however, from the fact that the would-be disassembler may not know what other information relating to the interface is available, or even necessary, until the disassembly has been performed. Companies that do publish interface information relating to their products may not publish *all* the details that a third party feels are necessary, or may publish them in a way that is confusing or otherwise incomplete.

One may speculate that the reference to "interface procedures" suggests that a reverse engineering exception will develop in the United States that is at least as broad as the reverse engineering exception contained in the European Software Directive recently adopted by the European Economic Community. That exception allows for reverse engineering to the extent necessary to establish "interoperability."[28]

How far beyond this the fair use exception may go will require further judicial development. Those desiring to do disassembly should consult carefully with counsel in advance.

[26] The *Sega* court noted that the "interface procedures for the Genesis console are distributed for public use only in object code form, and are not visible to the user during operation of the video game program." 1992 U.S. App. LEXIS 26645, at 17.

[27] *Id.* at 10.

[28] The preamble of the Council Directive defines "interoperability" as "the ability to exchange information and mutually to use the information which has been exchanged." Article 6(1) of the Directive provides that intermediate copying is permissible where it is "indispensable to obtain the information necessary to achieve the interoperability of an

independently created computer program with other programs, provided that the following conditions are met:

(a) these acts are performed by the licensee or by another person having a right to use a copy of a program, or on their behalf by a person authorized to do so;

(b) the information necessary to achieve interoperability has not previously been readily available to the persons referred to in subparagraph (a); and

(c) these acts are confined to the parts of the original program which are necessary to achieve interoperability."

Article 5(3) further provides: "The person having a right to use a copy of a computer program shall be entitled, without the authorization of the rightholder, to observe, study or test the functioning of the program in order to determine the ideas and principles which underlie any element of the program if he does so while performing any of the acts of loading, displaying, running, transmitting or storing the program which he is entitled to do.