DOES INTERMEDIATE COPYING OF COMPUTER SOFTWARE FOR THE PURPOSE OF REVERSE ENGINEERING A NON-INFRINGING PRODUCT INFRINGE THE COPYRIGHT IN THE SOFTWARE?

Robert V. Donahoe

INTRODUCTION

Researchers and engineers have long increased the speed of scientific and technological development by borrowing ideas from others. [1] The ideas contained in a work are often learned when the work is publicly sold or used. [2] Legitimate use of other's ideas for the benefit of broader society has its underpinnings in the U.S. Constitution and a long history of support in the judicial interpretation of the Patent and Copyright Acts. [3] This knowledge borrowing is legal so long as the property rights of the original innovator or artist are not violated. However, courts face a continuing challenge as technological developments create new claims of property rights violations. [4] The challenge is increased because technology frequently pushes beyond the direct reach of existing statutes. [5] Thus, courts ultimately find solutions for these new situations by applying the policy on which the law was based. [6]

The intent of the Copyright Act is to encourage the development of creative works to benefit the general public. [7] Copyright protection extends to the creative and expressive portion of the work. [8] However, copyright does not protect the facts, ideas or functional aspects contained in the work. [9] The grant of the copyright monopoly encourages development of additional creative works by encouraging the author of a work to share it with the public without the risk of the work's creative content being stolen. The creation of additional works is further promoted when the original work is published because others may use the ideas and factual information contained in the work to develop their own creative works. The publication of these new works provides a similar benefit to the general public. [10]

The tension surrounding copyrights on computer programs stems from their nature. Computer codes in their final form are designed to be read by computers. Thus, they are in a digital machine language, “object code”, that is not easily understood by humans, even software designers. [11] This is unlike literary works that are presented in alphanumeric characters that make up the language of the human reader. The copyright granted these works will act to protect the facts and ideas contained therein unless others are allowed to decompile the programs into an intelligible programming language, “source code.” The process of decompiling the object code and using the source code to develop a compatible product is referred to as reverse engineering and may involve repeated copying of the copyrighted computer program. [12] These copies are referred to as intermediate copies because the protected expression contained therein does not appear in the finished product. [13]

Software developers have successfully applied the affirmative defense of fair use to defeat claims that this intermediate copying violated the owner's copyright in the computer programmer. [14] Courts have found the equitable nature of the fair use defense useful in analyzing claims of computer program copyright infringement because the functional elements are unintelligible unless the object code is decompiled. [15] In *Sony v. Connectix*, 203 F.3d 596 (9th Cir. 2000), the Ninth Circuit applied the fair use doctrine to further expand the ability of computer software developers to legally make intermediate copies of computer software for the purpose of developing non-infringing products. The issue is vitally important to software developers because the reverse engineering of copyrighted material is used extensively in the industry. [16] It is just as important to companies who own copyrights in popular software and systems and view the intermediate copying as theft of their copyrighted work. The result of the copier's efforts adds insult to injury because it generally leads to the development of a product that takes market share from the owner of the original work. [17]

This memo analyzes the recent *Connectix* decision and the *Sega* decision to try and determine the boundaries of the fair use doctrine when applied to the intermediate copying of computer software for the purposes of reverse
engineering. The memo concludes that Ninth Circuit's reasoning supports intermediate copying as fair use where it is necessary and results in a final product that is transformative.

DISCUSSION

A. Fair Use

The inaccessible nature of copyrighted object code creates a unique challenge for courts dealing with copyright protection on these programs. The fair use doctrine has provided courts a flexible means of solving these problems in a manner supported by policy even where such solutions are not expressly provided for in the Copyright Act. [18]

Fair use is an affirmative defense to charges of copyright infringement. The fair use defense is a codified equitable rule of reason that was judicially created. [19] Thus, the doctrine allows the court flexibility in reaching a decision based on the facts of each case. Fair use allows reproduction of copyrighted works without permission where rigid application of the copyright statute would stifle the creativity it is designed to create. [20] Fair use claims are evaluated under multi-factor test described in section 107 of the copyright Act. [21] The statutory test includes the following factors: 1) purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes; 2) the nature of the copyrighted work; 3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and 4) the effect of the use upon the potential market for or value of the copyrighted work. [22] No factor is dispositive but all the statutory factors must be considered in the analysis. [23] The statute allows judicial consideration of other factors. [24] It is an attempt to balance the interest of the public in the production of creative works with the interest of the copyright owner in securing a fair return on their creative efforts. [25]

B. Ninth Circuit's Initial Approach to Intermediate Copying: Sega v. Accolade

The Connectix decision is the second decision where the Ninth Circuit held that fair use allowed the intermediate copying of copyrighted computer programs for purposes of reverse engineering a non-infringing product. [26] In the first case, Sega Enterprises Ltd. v. Accolade, Inc., the Ninth Circuit held that Accolade's copying of the BIOS from a Sega video game player was a fair use where it was used to develop Sega compatible games that did not contain copyrighted material. [27] The court concluded that the legitimate interest Accolade had in accessing the otherwise inaccessible ideas included in the BIOS was sufficient to constitute a fair use. [28]

The court acknowledged that protected expression was included in the copies and determined that intermediate copying may infringe the copyright in the computer code. [29] The court then evaluated the affirmative defense of fair use. [30] The court found, in applying the first statutory factor, that there was no direct commercial exploitation because the copying was intermediate. [31] Accolade's purpose, capturing the program's functional aspects, was important to the analysis of the first factor. [32] The court recognized that Accolade already had a creative work in its existing games. [33] It simply needed to include the functional aspects required for compatibility with the Sega console. [34] Additionally, the ultimate commercial use had to be evaluated against any public benefit that might result from the use. [35] The court determined that the increase in the number of independently designed video games offered for the Sega console did provide a public benefit. [36] Thus, the first factor favored Accolade. [37]

The court's analysis of the nature of the copyrighted work focused on two issues. [38] First, the court looked at the utilitarian nature of computer programs. [39] The court found that works such as computer programs that have a great proportion of unprotected functional or factual content receive a weaker form of copyright protection. [40] Second, the court found that the functional aspects of the computer program are not readily accessible. [41] The court looked to the policy behind the Copyright Act and held that statutory factor two favored Accolade because protecting the functional aspects of the copyrighted works is not the purposes of the copyright act and to prevent copying would grant the copyright holder a monopoly on the unprotected expression. [42]

The court did find that statutory factor three, the amount and substantiality of the portion used in relation to the copyrighted work as a whole, favored Sega because the entire work was copied. However, the court gave the factor little weight because the end product contained no protected expression. [43]
The court analyzed the effect on the potential market for the copyrighted work, statutory factor four, applying the standard established in *Harper & Rowe*. [44] The *Harper & Rowe* analysis finds that where the use effectively supplants the market for the copyrighted work the factor will favor the copyright holder. [45] Here the court determined that since consumers might purchase both Sega games and Accolade games the use of the copied work did not supplant the market for the copyrighted work. Thus, the fourth factor favored Accolade. [46]

The court's reasoning in *Sega* highlights ideas important to the Ninth Circuit's later analysis in *Connectix*. [47] The *Sega* court ultimately concluded that the policy rationale supporting copyright does not support a result that allows non-protected elements of a work to be kept from the public under the guise of copyright protection. Although a commercial product ultimately results, the court's approach favors the copier where the non-protected elements are inaccessible and the product does not supplant the market for the original.

C. Sony v. Connectix

The *Sony* court, unlike the *Sega* court, had the benefit of the Supreme Court's decision in *Campbell v. Acuff-Rose* when evaluating the fair use defense. [48] In *Acuff-Rose*, the Supreme Court applied the fair use analysis in holding that a commercial parody of a song could be fair use. [49] The *Acuff-Rose* court selected the term “transformative” to describe a work that doesn't supplant the original. [50] The *Acuff-Rose* court found that a transformative work had the characteristic of altering the original with something new, a further purpose or different character. The copying is more likely to be viewed as fair use where a work is transformative. [51]

In *Sony*, the court was faced with facts similar to those in *Sega*. [52] Sony sells a PlayStation console and PlayStation games sold in CD format. Players use the console in conjunction with a television to play the games. Sony owns a copyright on the system BIOS. Connectix Corporation developed a software program to allow the Sony PlayStation games to be played on a personal computer. Connectix developed the product by the process of reverse engineering. Connectix purchased a single Sony PlayStation, copied the system BIOS from a chip inside the Sony Play Station and used a decompiler to convert the object code to source code. Connectix then used the Sony BIOS with hardware emulation software in order to engineer a product that would play the Sony games on a personal computer. Connectix's final product did not contain any of Sony's protected expression. [53]

The Ninth Circuit held that intermediate copying of the copyrighted BIOS was a fair use where the end product was transformative and non-infringing. [54] The functional and inaccessible nature of the computer program, the intermediate nature of the copying and the transformative nature of Connectix's end product were critical to the court's analysis. [55] The court relied on the reasoning found in both *Sega* and *Acuff-Rose* in reaching its decision. [56]

Connectix argued that if their use was found not to be a fair use the result would be to allow monopoly protection greater than patent protection based on Sony's copyright. Thus extending Sony's control well beyond the intended scope of copyright. [57] The computer developers trade group supported Connectix by arguing that the inaccessibility of the functional elements supported a finding of fair use. [58] Sony's argument focused on the commercial nature and impact on Sony's market of the resulting product. [59] Sony attempted to distinguish this case from *Sega* by explaining that the copies were not just studied but actually used by the Connectix developers. [60] Sony also argued that Connectix made unnecessary copies because it did not use alternative methods of reverse engineering that would have required fewer copies of the BIOS. [61] A manufacturer's amicus brief in support of Sony argued that allowing this type of copying would speed the proliferation of video game piracy. [62]

The court rejected Sony's attempt to distinguish the case from *Sega* before turning to the fair use analysis. Relying on its 1993 decision in *Sega*, the Ninth Circuit found that the ideas contained in the computer program could not be accessed without intermediate copying. [63] The court allowed the “use” of copied material in reverse engineering and expressly rejected Sony's argument that intermediate use is limited to “studying” the copies. [64] The court also rejected the argument that Connectix made unnecessary copies. [65] The court characterized the necessity to copy as the necessity of the general method of disassembly, and held that the decision should not be based on the number of times the method was used because of the ease with which such a rule could be manipulated.
The court analyzed the four statutory fair use factors and found only the third factor was in Sony's favor. The court found that statutory factor one, purpose and character of the use, favored Connectix. Using the rationale of the Acuff-Rose decision, the court evaluated the degree to which the work was transformative and weighed that against the commercial nature of the use. The court found that Connectix's work was “modestly” transformative because it provided a new platform on which the Sony games could be played and because it contained entirely new object code. The court concluded that the first factor favored Connectix because the transformative nature outweighed the commercial use which was both intermediate and for a legitimate purpose (i.e. creating a product that would be compatible with games made by Sony).

The court followed the lead of both the Sega and Acuff-Rose courts in acknowledging that some categories of works receive a lower form of protection. The functionality of the computer program in this case afforded it a low degree of protection. The court also focused on the necessity of gaining access to these unprotected functional aspects of the program. Thus, the second factor favored Connectix.

The court determined that the third factor favored Sony because Connectix did copy the entire work. However, they indicated that it had little weight because the infringement was intermediate and the final product was non-infringing.

The court relied on the transformative nature of the Connectix product in finding that the effect of the use upon the potential market favored Connectix, the fourth factor. The court acknowledged the potential for decreased sales of the Play Station but found that it was outweighed by the transformative nature of the Connectix product. The court turned to the policy reasons supporting copyright and found that it was not designed as a tool to eliminate competition.

**D. The Impact of the Connectix and Sega Decisions**

The Ninth Circuit's analysis supports the efforts of software developers who use intermediate copying and reverse engineering so long as their efforts meet certain minimum standards. First, the copying must be “necessary” in order to be secure in the shelter of the fair use defense. This point was set out expressly in the Sega decision. Second, the end product must not contain any of the original work's protected elements. Where the end product contains protected expression a developer has moved from the safety of the intermediate copying category into “direct” copying. This will likely result in the fair use factors shifting towards a finding of infringement. The work must also be transformative. The Connectix decision makes it clear that the transformative quality is critical to the analysis of statutory factors one and four. Transformativeness is especially important given the commercial nature of the products involved in these cases because it can be used to offset the commercial harm that results from the copying. The fair use defense will most likely fail where the product is found to be non-transformative. For example, a company that develops a clone of a competitor's product via reverse engineering will likely be determined to be infringing even where the copyrighted software was only used in an intermediate fashion.

The Connectix case has removed the majority of copyright protection for object code, at least in the Ninth Circuit. The court's reasoning appears to allow software developers complete latitude in copying and manipulating these programs as required to develop a non-infringing transformative product. The copyright owner has a chance to prevail in an infringement claim of intermediate copying only where the copier's resulting product dictate the result. The manner of intermediate use is not dispositive where intermediate copying is claimed. Thus, object code developers should consider patent protection for new programs. Patent protection unlike copyright does protect the specific functional aspects that are claimed. Additionally, copyright owners should consider the pros and cons of licensing agreements with competitors in lieu of having them reverse engineer a competing product. Some income from Connectix in exchange for program information may have left Sony in a more favorable position than they ultimately found themselves in.

**CONCLUSION**

Copyright © 2001 Boston College Intellectual Property & Technology Forum, Robert V. Donahoe
The Ninth Circuit's conclusions in Sega and Connectix may not appear to be consistent with copyright protection granted to more traditional works. However, the Copyright Act was not created with digital technology in mind and does not have a provision that expressly deals with the circumstances of these two cases. Judicial solutions rely heavily on the policy behind the law in new and previously unaddressed situations. The court ultimately reached a decision that encourages technological development and is supported by the policy behind copyright protection. The Sega court concludes that this approach avoids trying to force a square peg into a round hole. One thing is certain - changing technology will continue to force the law to adapt to situations never contemplated by existing statutes. Furthermore, the court's response is important because it will dictate the manner in which future technology development occurs.

[2]. Id.
[5]. Id.
[6]. Id.
[9]. Id. See also 17 U.S.C. § 102(b) (copyright protection does not extend to, “any idea, procedure, process, system, method of operation, concept, principle or discovery”).
[10]. See Feist supra note 6.
[11]. See Sega supra note 4. Where this program is written into a read-only memory chip it is referred to as firmware or system BIOS. BIOS stands for basic-input-output-system.
[13]. See Sega supra note 4, at 1518.
[15]. See Sega supra note 4, at 1520.
[17]. *Id.*

[18]. See Connectix *supra* note 11, at 602.


[20]. *Id.* The preamble of §107 includes a non-exclusive list of examples of fair use; “criticism, comment, news reporting, teaching, scholarship, or research.”

[21]. *Id.*

[22]. *Id.*

[23]. *Id.*

[24]. *Id.*

[25]. See Connectix *supra* note 11, at 603.

[26]. See Sega *supra* note 4, at 1527.

[27]. *Id.*

[28]. *Id.*

[29]. *Id.* at 1519.

[30]. *Id.* at 1520.

[31]. *Id.* at 1522.

[32]. *Id.*

[33]. *Id.*

[34]. *Id.*

[35]. *Id.* at 1523.

[36]. *Id.*

[37]. *Id.*

[38]. *Id.* at 1524.

[39]. *Id.*

[40]. *Id.*

[41]. *Id.*
[42]. Id.

[43]. Id. at 1526-1527.

[44]. See supra note 1.

[45]. Id.

[46]. See Sega supra note 4, at 1523.

[47]. See Connectix supra note 11, at 602-603.


[49]. Id.

[50]. Id.

[51]. Id.

[52]. See Connectix supra note 11, at 602-603.

[53]. Id. at 599, 601.

[54]. Id. at 608.

[55]. Id.

[56]. Id.


[58]. Id.

[59]. They also argued that it violated certain of Sony's trademark rights. This point was also decided in Connectix's favor.

[60]. Id.

[61]. See Connectix supra note 11, at 605.


[63]. See Connectix supra note 11, at 602.

[64]. Id. at 604-605 (“Within the limited context of a claim of intermediate infringement, we find the semantic distinction between 'studying' and 'use' to be artificial, and decline to adopt it for purposes of determining fair use”).

[65]. Id.
[66]. Id.

[67]. Id. at 605-606.

[68]. Id. at 606.

[69]. Id. at 606-607.

[70]. Id.

[71]. Id.

[72]. Id. at 603.

[73]. Id.

[74]. Id. at 605.

[75]. Id. at 605-606.

[76]. Id.

[77]. Id. at 607-608.

[78]. Id.

[79]. Id.

[80]. Id.

[81]. Id. at 605.

[82]. See Sega supra note 4 (“... in many cases the operation of a program is directly reflected on the screen display and therefore visible to the human eye. In those cases, it is likely that a reverse engineer would not need to examine the code in order to understand what the program does).

[83]. Id. at 606-607.

[84]. Id.

[85]. Id.

[86]. Id.

[87]. Id.

[88]. Id.

[89]. Id.

[90]. Id.

[92]. See Connectix supra note 11, at 604-605.

[93]. Id.

[94]. See Sandburg, supra note 90.


[96]. See Sega supra note 4, at 1527.


[98]. See Sega supra note 4, at 1527.

[99]. Id.

[100]. Id.