January 2005

Copyright Law and Subject Matter Specificity: The Case of Computer Software

Joseph P. Liu
Boston College Law School, joseph.liu@bc.edu

Stacey L. Dogan
Northeastern University School of Law, s.dogan@neu.edu

Recommended Citation
Joseph P. Liu and Stacey L. Dogan. "Copyright Law and Subject Matter Specificity: The Case of Computer Software." *New York University Annual Survey of American Law* 61, no.2 (2005): 203-236.

COPYRIGHT LAW AND SUBJECT MATTER SPECIFICITY: THE CASE OF COMPUTER SOFTWARE

BY STACEY L. DOGAN AND JOSEPH P. LIU***

ABSTRACT

Drawing on recent work by Dan Burk and Mark Lemley in the patent context, this paper explores the extent to which courts have adapted pre-existing copyright doctrines to the special case of computer software. We argue that a number of courts have, as has been widely recognized, significantly adapted copyright doctrines to deal with special features of the computer software market. We further argue that these adaptations have, by and large, positively sought to strike a balance between the copyright act's dual goals of incentive and access. Despite this general trend toward adaptation, however, we point to a handful of instances in which courts and legislatures have adopted a more wooden approach to software copyright questions. Given the nuanced nature of copyright law's underlying goals, we contend that some level of flexibility and adaptation is critical in the software context, where network effects, interoperability, and functionality play a prominent role. We suggest that copyright law should—and indeed must—have some vehicle for considering these unique features of software markets, and we recommend a number of changes to maintain the more flexible, policy-lever approach to software copyright law.

CONTENTS

* Associate Professor, Northeastern University School of Law.

** Associate Professor, Boston College Law School. Thanks to Michael Carroll, Mark Lemley, Michael Meurer, Shubha Ghosh, and the participants at the Works-in-Progress Intellectual Property Colloquium at Boston University School of Law and the IP Meets IT Conference at the University of Pittsburgh School of Law for helpful comments and suggestions.

I. INTRODUCTION

In a recent series of articles, Dan Burk and Mark Lemley have examined the extent to which patent law is technology specific.[1] Although the patent statute[2] and patent doctrines are facially technology neutral, Burk and Lemley argue that the courts have in fact applied these doctrines in a way that differs significantly depending on the underlying technology. Although Burk and Lemley take issue with the specific way in which the courts have adapted patent law to particular technologies, they ultimately conclude that this kind of flexible adaptation, when done consciously and with an eye toward broader policy considerations, can play an important role in adjusting the law to the different ways in which innovation occurs in different industries.

Inspired by this recent work, this paper takes a preliminary look at the same question in the context of copyright law. Is copyright law subject-matter specific? Does it vary according to the type of copyrighted work? At one level, the answer to these questions is a straightforward "yes." The copyright act, unlike the patent act, expressly distinguishes between different types of copyrighted works.[3] In many cases, the rights and limitations vary significantly according to the type of work at issue. Thus, for example, musical works are expressly subject to a very different set of rights and limitations as compared to books, movies, or software.[4] Indeed, Burk and Lemley

1. Dan L. Burk & Mark A. Lemley, *Policy Levers In Patent Law*, 89 VA. L. REV. 1575 (2003) [hereinafter Burk & Lemley, *Policy Levers*]; Dan L. Burk & Mark A. Lemley, *Is Patent Law Technology-Specific?*, 17 BERKELEY TECH. L.J. 1155, 1156–57 (2002).

2. 35 U.S.C. §§ 1–376 (2000).

3. *See* 17 U.S.C. § 102(a) (2000).

4. The publishing and movie industries are generally subject to the same copyright provisions that govern all copyrighted works more generally. *See, e.g., id.* § 106 (defining exclusive rights). The music industry, by contrast, is governed by a complex overlay of additional provisions, including complex compulsory licenses. *See, e.g., id.* § 114 (compulsory license for sound recordings); Lydia Pallas Loren,

cite this kind of statutory subject-matter specificity in copyright as an example of a path that patent law should not take.[5]

In this paper, we are not interested—at least not directly—in this kind of express, statutory subject-matter specificity. Instead, we focus on examining whether, in areas where the copyright act makes no such express distinctions, courts nevertheless apply copyright law in a differential manner. For example, the idea-expression dichotomy[6] is a general copyright doctrine that applies to all categories of copyrightable subject matter. Do courts apply this doctrine differently based on the underlying subject matter? Similarly, do courts adjust their infringement or fair use analysis to the specifics of different copyright markets? Or in Burk and Lemley's terms, do courts in copyright cases also use "policy levers"[7] in copyright law to adjust and adapt copyright law in this manner?[8]

This paper begins the process of considering copyright's subject matter specificity. We address the question in the narrow context of computer software, as it presents the clearest example of this kind of adaptation. In Part II of this paper, we argue that, as has been widely recognized, courts have significantly adapted pre-existing copyright doctrines to fit the peculiarities of the computer software market in cases involving competing and interoperable software.[9] Thus, for example, courts in software copyright cases have adapted the doctrines of infringement, idea-expression, and fair use in ways that depart significantly from other areas of copyright law.[10]

Untangling The Web Of Music Copyrights, 53 CASE W. RES. L. REV. 673 (2003) (analyzing copyright law's interaction with the music industry).

5. *See* Burk & Lemley, *Policy Levers*, *supra* note 1, at 1637–38.

6. The idea-expression dichotomy refers to the distinction drawn between copyrightable expression and uncopyrightable ideas. *See, e.g.*, Baker v. Selden, 101 U.S. 99, 102–04 (1879) (distinguishing between copyrightable books and the uncopyrightable content described therein).

7. According to Burk and Lemley, "policy levers" are doctrines, such as the "person having ordinary skill in the art" in patent law, that give courts flexibility to adapt patent law to the specifics of particular technologies. Burk & Lemley, *Policy Levers*, *supra* note 1, at 1578, 1648–51.

8. *See* Pamela Samuelson & Suzanne Scotchmer, *The Law and Economics of Reverse Engineering*, 111 YALE L. J. 1575, 1649–50 (2002) (discussing right to reverse engineer as a policy lever used by courts in software copyright cases).

9. *See, e.g.*, Peter S. Menell, *Envisioning Copyright Law's Digital Future*, 46 N.Y.L. SCH. L. REV. 63, 65 (2002-03) (observing that courts, over time, settled on an approach to software copyright that "finessed the metaphysical dilemmas and avoided the creation of undue economic power in computer markets.").

10. *See infra, Part* II.B.

We further argue that this judicial adaptation has, by and large, been a positive development. Courts have, on the whole, successfully adapted copyright doctrines in a way that respects the underlying copyright policies, as applied to the unique aspects of computer software. In particular, judges in copyright cases have effectively mediated between two core goals of copyright law: protecting against the exploitative use of copyrighted expression, while allowing the dissemination and use of the ideas and functions that may be embedded in such expression. In advancing these goals, courts have consciously considered the functional nature of software, its inherent non-transparency, and its unique level of interactivity with both users and complementary software products.

In Part III of this paper, we argue that, despite this encouraging trend, a separate line of recent cases and legislative developments runs in quite a different direction. We identify three areas in which the law has evolved in a more formalistic manner, without the type of careful consideration to underlying copyright policies or the peculiarities of the software market that characterized the seminal software copyright decisions. In each of these three areas—which we refer to as the service and repair cases, the Digital Millennium Copyright Act (DMCA)[11] developments, and the (over)extension of contract law—we fear that the move away from an adaptive approach will undermine the careful balance achieved by courts in the earlier cases. We argue that these departures from the trend toward adaptation are unwarranted, and we offer suggestions for returning to the adaptive approach. Indeed, several recent decisions suggest that courts share our concern about the decline of copyright's policy levers, and are finding creative ways to re-import them.

More generally, this case study of subject matter specificity in the context of computer software sheds light on the broader question of whether and how copyright should be subject-matter specific. Copyright law's shifting approach to computer software highlights the relative strengths and weaknesses of the two different approaches to subject-matter adaptation discussed in Burk and Lemley's work. We ultimately conclude, in agreement with Burk and Lemley, that a judicial approach to adaptation may offer advantages over a legislative approach, particularly where an industry is subject to rapid technological change.

11. 17 U.S.C. §§ 1201–1205 (2000).

II.
THE SPECIAL TREATMENT OF COMPUTER SOFTWARE

A. The Lead-up

As many others have observed before us, copyright and software make strange bedfellows.[12] The classic articulations of copyright law describe it as protecting expression rather than function,[13] and yet computer programs are, by nature, functional.[14] At the same time, programs are usually written in a "language" and contain the kinds of symbols, letters, and numbers that we traditionally associate with literary works. This fact, together with an instinct that copyright was the best available alternative, persuaded the National Commission on New Technological Uses of Copyrighted Works (CONTU) to recommend, and Congress to endorse, copyright protection for software more than two decades ago.[15] CONTU recognized that the pairing might present challenges for the law, but suggested that the flexibility of copyright doctrine

12. *See* Computer Assocs. Int'l v. Altai, Inc., 982 F.2d 693, 712 (2d Cir. 1992) ("Thus far, many of the decisions in this area reflect the courts' attempt to fit the proverbial square peg in a round hole."). The literature on the suitability of copyright to software is voluminous, and we mention only a few noteworthy examples. *See, e.g.*, Dennis S. Karjala, *A Coherent Theory for the Copyright Protection of Computer Software and Recent Judicial Interpretations*, 66 U. CIN. L. REV. 53 (1997); Mark A. Lemley & David W. O'Brien, *Encouraging Software Reuse*, 49 STAN. L. REV. 255 (1997); Peter S. Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 STAN. L. REV. 1045 (1989); Pamela Samuelson et al., *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 COLUM. L. REV 2308 (1994); *cf.* Jane C. Ginsburg, *Four Reasons and a Paradox: The Manifest Superiority of Copyright Over Sui Generis Protection of Computer Software*, 94 COLUM. L. REV. 2559 (1994); Arthur R. Miller, *Copyright Protection for Computer Programs, Databases, and Computer-Generated Works: Is Anything New Since CONTU?*, 106 HARV. L. REV. 977 (1993).

13. *See* 17 U.S.C. § 102(a), (b) (separating decisively a "work of . . . expression" from "any idea, procedure, process, system, method of operation, concept, principle, or discovery"); *see* Baker v. Selden, 101 U.S. 99, 102-04 (1879); *cf.* Sheldon v. Metro-Goldwyn Pictures Corp., 81 F.2d 49 (2d Cir. 1936) (rejecting a film studio's argument that in modeling a motion picture on a copyrighted play based on actual events, the studio had merely utilized "general themes, motives, or ideas in which there could be no copyright").

14. *See* Samuelson et al., *supra* note 12, at 2314 ("Although programs are texts and their texts can be valuable, the most important property of programs is their behavior (i.e., the set of results brought about when program instructions are executed).").

15. *See* NAT'L COMM'N ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT 1, 15–18 (1978) (hereinafter, "CONTU REPORT"). Congress commissioned CONTU in 1974 to examine, among other issues, the question of whether copyright law should protect software and, if so, what changes were required to the Copyright Act in order to accommodate the medium. *See* Act of Dec. 31, 1974, Pub. L. No. 93-573, § 201, 88 Stat. 1873, 1873–74.

would allow for accommodation to software's unique character. In particular, the report suggested that, "[s]hould a line need to be drawn to exclude certain manifestations of programs from copyright, that line should be drawn on a case-by-case basis by the institution designed to make fine distinctions—the federal judiciary."[16]

Following this cue, Congress amended the Copyright Act in only minor ways to accommodate the new medium, adding a definition of "computer program" and providing that "owner[s]" of computer programs have the right to make copies for archival reasons or in the course of using them for their intended purpose.[17] The more fundamental questions—what aspects of computer programs deserved protection, and against what kinds of copying—awaited resolution by the courts. It remained to be seen whether the courts would address these questions mechanically, or more flexibly with an eye toward the broader implications for the software industry and others affected by software copyrights.

Initially, the courts took a rather wooden approach. In *Apple Computer, Inc. v. Franklin Computer Corp.*,[18] for example, the defendant had argued that compatibility considerations counseled in favor of either limiting or denying protection for individual operating system programs.[19] The Third Circuit, however, refused to consider such industry-oriented concerns, holding instead that "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

16. CONTU REPORT, *supra* note 15, at 22-23. As a normative matter, CONTU emphasized that the law of software copyright should develop in a way that balanced the interests of software developers and those of the public. The Commission identified four goals of software copyright:

1. Copyright should proscribe the unauthorized copying of these works.
2. Copyright should in no way inhibit the rightful use of these works.
3. Copyright should not block the development and dissemination of these works.
4. Copyright should not grant anyone more economic power than is necessary to achieve the incentive to create.

Id. at 12.

17. *See* 17 U.S.C. §§ 101, 117(a) (2000).

18. 714 F.2d 1240 (3d Cir. 1983).

19. *Id.* at 1253 (noting Franklin's claim "that whether or not the programs can be rewritten, there are a limited 'number of ways to arrange operating systems to enable a computer to run the vast body of Apple-compatible software'"). *Apple v. Franklin* itself involved literal copying of entire programs, rather than program interfaces, which may have affected the court's analysis. The court's language, however, suggested a general hostility to the notion of interoperability as a justification for copying.

enter into the somewhat metaphysical issue of whether particular ideas and expressions have merged."[20] Commercial and competitive objectives, in other words, had no relevance to copyright doctrine; protectability turned instead on whether the program's "idea"—i.e., its ultimate function—could have been accomplished through alternative versions of the code. This doctrinal approach was not nonsensical; it found roots in the longstanding common law approach to the copyright doctrine of merger.[21] But in the software context, the court's analysis overlooked the fact that a program's "function" can signify many different things, including the ability to work with other programs.[22] More generally, *Apple v. Franklin* and decisions like it reflected an unwillingness to adapt copyright doctrines to address concerns about the unique features of software—including its functionality and the interdependence of certain software programs[23]—that might have justified a more flexible approach.

B. Policy Levers: The Altai-Lotus-Sega *Trilogy*

Beginning in the early 1990s, courts in many software copyright cases began to show a new concern toward the needs of consumers and competitors in software markets. Judicial decisions moved from a formulaic application of pre-software doctrines toward a view of such doctrines as flexible tools to achieve copyright law's normative goals. In particular, courts began to consider issues such as lower-level functionality, interoperability, and use in analyzing questions of copyrightability, infringement, and fair use.

In considering the protectability of program features, the early cases had looked at the copyrighted program in isolation, and asked only whether the programmer had other design options to

---

20. *Id.*

21. *See* Morrissey v. Procter & Gamble Co., 379 F.2d 675, 678–79 (1st Cir. 1967) ("When the uncopyrightable subject matter is very narrow, so that 'the topic necessarily requires,' if not one form of expression, at best only a limited number, to permit copyrighting would mean that a party or parties, by copyrighting a mere handful of forms, could exhaust all possibilities of future use of the substance.") (quoting Sampson & Murdock Co. v. Seaver-Radford Co., 140 F. 539, 541 (1st Cir. 1905)) (other citation omitted).

22. The Third Circuit adopted a similarly restrictive approach to merger in *Whelan Assoc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1236–40 (3d Cir. 1986) (defining "idea" of computer program as the ultimate function to be performed by the program, with all implementations of that idea potentially copyrightable).

23. *See, e.g.*, Samuelson & Scotchmer, *supra* note 8, at 1615. ("In the software industry, platforms and applications are not just complementary products; they are complementary parts of a system by virtue of their conformity to interfaces necessary for achieving interoperability.").

achieve the program's overall function.[24] Questions of efficiency, interoperability, and programming context were deliberately set aside. The Second Circuit's 1992 decision in *Computer Associates International v. Altai, Inc.*[25] marked a turning point in software copyright analysis. The court spurned the detached approach of the early software cases, and held that software copyright decisions *had* to consider context:

> We think that *Whelan's* approach to separating idea from expression in computer programs relies too heavily on metaphysical distinctions and does not place enough emphasis on practical considerations. . . . As the cases that we shall discuss demonstrate, a satisfactory answer to this problem cannot be reached by resorting, *a priori*, to philosophical first principles.[26]

Rather than considering code and program structure in the abstract, the Second Circuit adopted a fact-specific "[a]bstraction–[f]iltration– [c]omparison" approach for identifying the protected aspects of software and determining whether they have been infringed.[27] While the specifics of the test have attracted the attention of countless scholars and courts,[28] we limit ourselves to two general observations. First, the court self-consciously used copyright doctrines as policy levers to accommodate the unique nature of software. In discussing why merger analysis in software cases should consider the "structural economy" of a program,[29] for example, the court pointed to "a program's essentially utilitarian nature and the competitive forces that exist in the software marketplace."[30] Second, the goals of compatibility and interoperability were presumed without discussion to be functional objectives that the policy levers could be adjusted to achieve. The court held, for example, that the *scenes a faire* doctrine requires the filtering out of program

---

24. *E.g.*, *Whelan*, 797 F.2d at 1236–40.

25. 982 F.2d 693 (2d Cir. 1992).

26. *Id.* at 706.

27. *Id.* at 706–12.

28. *See, e.g.*, Peter S. Menell, *Envisioning Copyright Law's Digital Future*, 46 N.Y.L. SCH. L. REV. 63, 84 (2002-03); Peter S. Menell, *The Challenges of Reforming Intellectual Property Protection for Computer Software*, 94 COLUM. L. REV. 2644, 2652 (1994); Mark A. Lemley, *Convergence in the Law of Software Copyright?*, 10 HIGH TECH. L.J. 1 (1995).

29. By "structural economy," the court referred to the efficiency of a particular choice in program structure. If a particular set of modules was found necessary to efficiently implement a particular program sub-function, for example, the court indicated that the merger doctrine would bar protection. *Computer Assocs.*, 982 F.2d at 708.

30. *Id.* The court also commented that, "[e]fficiency is an industry-wide goal." *Id.*

elements dictated by external factors, including "compatibility requirements of other programs with which a program is designed to operate in conjunction."[31]

Three years later, in *Lotus Development Corp. v. Borland International, Inc.*,[32] the First Circuit showed similar concerns about use and interoperability in analyzing copyrightable subject matter. On its face, the decision purported to offer a straightforward doctrinal analysis, finding a spreadsheet program's menu command hierarchy uncopyrightable as a "method of operation."[33] The court supported its decision, however, with reference to practical considerations—and particularly to concerns about compatibility—this time from the perspective of end users:

> That the Lotus menu command hierarchy is a "method of operation" becomes clearer when one considers program compatibility. Under Lotus's theory, if a user uses several different programs, he or she must learn how to perform the same operation in a different way for each program used. For example, if the user wanted the computer to print material, then the user would have to learn not just one method of operating the computer such that it prints, but many different methods. We find this absurd.[34]

The court's interpretation of "method of operation," in other words, took account of practical considerations, including program functionality in a pragmatic rather than a metaphysical sense.[35] Like the *Altai* court, the First Circuit in *Lotus* used the subject matter inquiry as a policy lever that enabled a textured approach to

---

31. *Id.* at 710. The court also tailored its approach to infringement analysis, finding that the "ordinary observer" test may require some expert input, given the technical nature of software. *See id.* at 713-14.

32. Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807 (1st Cir. 1995) *aff'd per curiam by an equally divided Court*, 516 U.S. 233 (1996).

33. *Id.* at 815–17 (method of operation includes "the means by which a person operates something"). *See also* 17 U.S.C. § 102(b) (2000) (excluding methods of operation from copyright protection).

34. *Id.* at 817–18.

35. Of course, on some level, one could view every aspect of a computer program as a "method of operation," because every line of code comprises part of the means through which the program is operated. *Lotus* did not address this issue because the case involved structure of interface rather than code, but its analysis suggested that the "method of operation" issue should be resolved pragmatically, with reference to the needs of users of the software. The *Lotus* court found the menu commands to be methods of operation because they served as points of interaction between the user and the program; this approach would presumably find programming interfaces—i.e., points of interaction between programs—similarly unprotectable. *See id.* at 815–18.

software analysis, one that considered the impact of copyright protection on users and (at least implicitly) on competitors in the marketplace.

Judge Boudin's concurrence in *Lotus* was even more frank in advocating a policy-levers approach. In Boudin's estimation, "the heart of copyright doctrine—what may be protected and with what limitations and exceptions—has been developed by the courts through experience with individual cases."[36] Rather than a "cookie cutter" approach to questions of subject matter and scope, he observed that "case law development is *adaptive*: it allows new problems to be solved with help of earlier doctrine, but it does not preclude new doctrines to meet new situations."[37] The new situation in *Lotus*, he suggested, arose not so much from the inherently functional nature of the menu commands,[38] but from the fact that Lotus's dominant position in the spreadsheet market had effectively locked most users into programs that were compatible with its commands.[39] Boudin would have allowed Borland's copying for uniquely fact-specific reasons: because it incorporated the menu command structure into a better product, solely to overcome the barriers to entry that existed because of Lotus's dominance.[40] His preferred policy lever—the fair use doctrine—admits to a more nuanced approach that expressly balances the user's need for access against this access's possible impact on the incentive structure of copyright law.[41] Boudin's choice of policy lever differed from the majority's, but his motivating goal—to allow access to program functions that mattered to consumers—reflected the same pragmatic concerns as the majority opinion.

While Judge Boudin did not persuade a majority of the *Lotus* panel to adopt a privileged use approach, the Ninth Circuit in *Sega*

---

36. 49 F.3d at 820.

37. *Id.*

38. Judge Boudin viewed as "defensible" the decision to call the menu command hierarchies unprotectable methods of operation, but suggested that a more nuanced approach focused on the *need* to copy might be preferable to a categorical exclusion of such features from copyright law. *See id.* at 821 ("The difference is that such a privileged use approach would not automatically protect Borland if it had simply copied the Lotus menu (using different codes), contributed nothing of its own, and resold Lotus under the Borland label.").

39. *See id.* at 821 (discussing users being "locked into Lotus" as a result of its "sway" in the market, which made it the "de facto standard for electronic spreadsheet commands").

40. *See id.*

41. *See* Stacey L. Dogan, *Infringement Once Removed: The Perils of Hyperlinking to Infringing Content*, 87 IOWA L. REV. 829, 836 (2002) (noting copyright law's balance between incentive and access).

Enterprise v. Accolade, Inc.[42] opted for fair use's inherent flexibility in considering whether intermediate copying constitutes infringement. Accolade, the defendant, had reverse engineered Sega's game cartridges to discover the requirements for compatibility with the Sega console. In that process, Accolade made a complete copy of Sega's source code, but solely to learn how to make a compatible product. The Sega-compatible cartridges that Accolade later sold had only a few lines of code in common with the Sega game. Sega sued for copyright infringement based on the intermediate copies made in the reverse engineering process.

Although Accolade tried to persuade the court to declare intermediate copies of computer programs presumptively legal, the court found no room in the Copyright Act for such a categorical limitation on copyright owners' rights. To read the definition of "copy" to exclude intermediate copies, the court found, would go beyond importing flexibility into broad doctrines; it would defy the clear language of the Copyright Act.[43] Even the more flexible idea/expression distinction could not be stretched to justify a per se rule in favor of software copying, given Congress's unambiguous intent to give computer programs "the full range of copyright protection" under the Act.[44]

Despite its refusal to exempt all intermediate copies, however, the *Sega* court was clearly sympathetic to Accolade's argument that copyright law should allow space for parties who copy solely to discover the unprotected, functional characteristics of a computer program. The court found the needed flexibility to address these concerns in the fair use doctrine. Like the *Altai* and *Lotus* courts, the Ninth Circuit in *Sega* relied heavily on pragmatic considerations—and particularly the need for interoperability—in finding Accolade's copying permissible.

In considering the nature of Accolade's use, for example, the court emphasized that Accolade's purpose in making the copies "was simply to study the functional requirements for Genesis compatibility,"[45] and that "no other method of studying those require-

42. 977 F.2d 1510, 1514 (9th Cir. 1992).

43. *See* 977 F.2d at 1518–19 ("In light of the unambiguous language of the Act, we decline to depart from the rule . . . for copyrighted works generally.").

44. *Id.* at 1519–20. Because Accolade had copied Sega's code in full, including both expressive and non-expressive components, this result comported with the analysis of idea/expression in *Altai* and *Lotus*.

The court also rejected Accolade's reliance on § 117, finding that Accolade's multiple copies "went far beyond that contemplated by CONTU and authorized by section 117." *Id.* at 1520.

45. *Id.* at 1522.

ments was available."[46] The market, moreover, benefited from the introduction of new Sega-compatible products.[47] Because Accolade's product worked with Sega's consoles but did not exploit the expression in Sega's own games, the intermediate copying ultimately promoted the underlying goals of copyright law.[48] This allowed the court to discount the commercial nature of the use, which ordinarily would have counted heavily against Accolade. The court's analysis of market harm similarly departed from traditional fair use analysis. Accolade's games competed with Sega's games and would thus ordinarily have given rise to a strong presumption of harm to the market. However, the court suggested that the games might not be complete substitutes for each other.[49] The court went on to note, in any event, that the importance of enabling competing compatible products outweighed any economic loss suffered by Sega.[50]

The Ninth Circuit quite consciously viewed fair use as a policy lever. The court acknowledged that its opinion stretched fair use doctrine beyond its traditional bounds, but found the result necessitated by the nature of software and the Copyright Act's expression-promoting goals:

> As discussed above, the fact that computer programs are distributed for public use in object code form often precludes public access to the ideas and functional concepts contained in those programs, and thus confers on the copyright owner a *de facto* monopoly over those ideas and functional concepts. That result defeats the fundamental purpose of the Copyright Act—to encourage the production of original works by protecting the expressive elements of those works while leaving the ideas,

---

46. *Id.*

47. The court stated:

> In the case before us, Accolade's identification of the functional requirements for Genesis compatibility has led to an increase in the number of independently designed video game programs offered for use with the Genesis console. It is precisely this growth in creative expression, based on the dissemination of other creative works and the unprotected ideas contained in those works, that the Copyright Act was intended to promote.

Id. at 1523.

48. *See also* Lewis Galoob Toys, Inc. v. Nintendo of Am., Inc., 964 F.2d 965, 971 (9th Cir. 1992) ("a party who distributes a copyrighted work cannot dictate how that work is to be enjoyed."). *See generally* Samuelson & Scotchmer, *supra* note 8.

49. Sega, 977 F.2d, at 1523.

50. *Id.*

facts, and functional concepts in the public domain for others to build on.[51]

The opinion shares much in common with *Altai* and *Lotus*. Like those cases, the *Sega* decision used a general copyright doctrine as a policy lever to take account of the interests of competitors and consumers in the software context. And while the particular policy lever differed from those cases, its object closely resembled that of *Altai* and *Lotus*. Yet again, the court balanced the incentive goals of copyright against the public's interest in understanding software products and copying their unprotected features; and once again, the court found consumers' interest in accessing compatible products a legitimate basis for employing policy levers in a way that cabined copyright holders' rights.[52]

The *Altai-Lotus-Sega* trilogy triggered a general shift in courts' thinking about computer copyright cases. Although the details of their doctrinal analysis frequently diverged, most subsequent decisions recognized interoperability and user-oriented functionality as critical to maintaining copyright's balance in software cases, and viewed copyright's traditional doctrines as flexible enough to achieve that balance. Whether through resort to the merger doctrine,[53] the idea-expression dichotomy, the "process-expression dichotomy,"[54] the exclusion of processes,[55] the *scenes a faire* doctrine,[56] the derivative work right,[57] or fair use,[58] the courts in-

51. *Sega*, 977 F.2d at 1527.

52. *See also* Sony Computer Entertainment v. Connectix, 203 F.3d 596 (9th Cir. 2000) (finding intermediate copying to constitute fair use in a case involving reverse engineering to create a video game emulator that would run Sony Playstation games).

53. *See* Liberty Am. Ins. Grp. v. WestPoint Underwriters, 199 F. Supp. 2d 1271, 1301 (M.D. Fla. 2001) (finding no protection when the "parties' computer experts . . . agreed that certain arithmetic or design functions (e.g., drawing a box) limit how those ideas can be expressed.").

54. *E.g.*, Gates Rubber Co. v. Bando Chem. Indus., 9 F.3d 823, 836–37 (10th Cir. 1993).

55. *E.g.*, MiTek Holdings, Inc. v. Arce Eng'g Co., 89 F.3d 1548, 1557 (11th Cir. 1996) (refusing to protect software implementation of drafting function that mirrored the process as performed by human draftsperson).

56. *See Gates Rubber*, 9 F.3d at 838 (requiring filtration of features dictated by external factors including "hardware standards and mechanical specifications, software standards and compatibility requirements, computer manufacturer design standards, target industry practices and demands, and computer industry programming practices") (internal citations omitted).

57. *See* Micro Star v. Formgen, 154 F.3d 1107 (9th Cir. 1998); Lewis Galoob Toys, Inc. v. Nintendo of Am., Inc., 964 F.2d 965, 969–72 (9th Cir. 1992).

58. *See* Sony Computer Entm't, Inc. v. Connectix Corp., 203 F.3d 596 (9th Cir. 2000) (extending *Sega* to protect reverse engineering aimed toward develop-

creasingly used copyright's doctrines as policy levers to moderate the effects of copyright protection on users and makers of software.

By the mid-1990s, then, courts were showing a promising level of flexibility in considering infringement claims against software developers who copied features to develop competing or compatible software products. The normative goals of copyright law informed both the inquiry into what was protected, and the analysis of whether the defendant's use had infringed. Copyright law, in other words, provided vehicles for considering context in infringement suits, and courts were beginning to utilize those vehicles to adapt the law to the unique attributes of software.

C. The Policies Behind the Policy Levers

Beyond a general move toward contextual analysis, the *Altai-Sega-Lotus* trilogy and its progeny reflected a growing consensus over the type of interests that can justify wielding copyright's policy levers in the software context. Of course, the recognition of these interests as legitimate does not mean that they will prevail over the copyright holder's concerns in any particular case, but it suggests that they should at least be considered. We group these concerns into three general categories: edification, compatibility, and use.

1. Edification

Among its other distinctive features, software stands alone among copyrighted works in its non-transparency. To understand the workings of a system described in a book, one need merely pick up the book and read it. But as the Ninth Circuit recognized in *Sega*, understanding a computer program frequently requires the making of copies of that program, an act that would ordinarily constitute infringement.

Sega and other similar cases suggest that copying software in order to understand it can often promote copyright's core goals. When, for example, the copying reveals functional specifications that enable the creation of compatible but non-infringing products, it promotes the multiplicity of expressive products and viewpoints in our economy. Copying may not always achieve this result, however,[59] and for this reason the Ninth Circuit refused to issue a blan-

---

ing a competing (rather than compatible) product that uses similar functional specifications). *See also* Lewis Galoob Toys, Inc. v. Nintendo of Am., Inc., 964 F.2d 965, 969–72 (9th Cir. 1992).

59. *See* Samuelson & Scotchmer, *supra* note 8 at 1651 (suggesting context-specific approach to the reverse engineering "policy lever," given the uncertain impact that reverse engineering generally has on software innovation).

ket endorsement of edification-oriented copying, opting for the flexibility of fair use. But the fact that software does not immediately reveal its meaning to users—and, in particular, the fact that code, as distributed, conceals many elements that are unprotected by copyright—makes the goal of edification—i.e., copying programs to understand how they work—an appropriate policy goal to weigh in the balance.

2. Compatibility

In stark contrast to the early *Apple v. Franklin* approach, the later cases show a significant interest in software developers' ability to make products that work with other programs. *Altai*, by filtering interoperability requirements out of copyright analysis, treats the goal of interoperability as almost a legal right of software developers. *Sega* shows similar leanings, finding interoperability a valid excuse for intermediate copying. And *Lotus* offers its own support to interoperability as a legitimate goal by recognizing users' interest in buying programs that work with their existing macros.[60]

3. Use

The third concern reflected in these cases is the interest of users in accessing the functionality of programs that they buy.[61] This concern is admittedly more ambiguous than edification and compatibility, but it nonetheless resonates in the cases. The interest of users is implicit in the rule favoring compatibility, which supports the ability of users to purchase products that will work with their existing software. This concern emerges forcefully in *Lotus*, which views user commands as entirely beyond the scope of copyright protection.[62] It also surfaces in *Altai*, which instructs courts to

60. Lotus Dev. Corp. v. Borland Int'l Inc., 49 F.3d 807, 818 (1st Cir. 1995) *aff'd per curiam by an equally divided Court*, 516 U.S. 233 (1996). Many users of the Lotus 1-2-3 spreadsheet had written "macros," or short programs, to shorten the time required to perform certain tasks using the program. The court viewed the existence of these macros as further evidence that the Lotus menu command hierarchy constituted a method of operation. *Id.*

61. We use "buy" here mindful of the distinction frequently drawn between sale and license in the software context. None of the cases discussed above distinguished between license and sale, and each involved application of copyright, rather than contract law. *See infra* III.C (discussing distinction between license and sale in contract and copyright law).

62. *Lotus*, 49 F.3d at 817 (finding user commands to be "methods of operation" within the meaning of the Copyright Act).

filter out user specifications from the scope of copyright.[63] At the very least, these decisions appear mindful of the fact that programs are designed to be used, and the nature of the use of particular program features should play a role in evaluating whether those features deserve protection against copying.

III.
COUNTERVAILING TRENDS

The move toward judicial adaptation in software copyright cases has been far from uniform. While the early 1990s experienced a promising trend in adapting copyright law to accommodate software, the trend was limited to suits against those who designed competing or compatible software. Others—including independent service organizations who "copied" software in the course of computer repair—faced a less sympathetic reception in the courts. More recent years have witnessed a retrenchment even in the product development context. Congress's passage of the Digital Millennium Copyright Act of 1998 ("DMCA") created a series of new rights and restricted courts' ability to shape those rights in response to changes in markets or technology.[64] In addition, a number of courts have shown an increasing willingness to defer to software developers' licensing terms, which has led to a partial supplanting of the nuanced approach to software copyright discussed above. In this section, we introduce these developments, advance a few potential explanations for why they have occurred, and describe why they are troubling.

A. The Service and Repair Cases

Even as courts became increasingly comfortable with the use of copyright's policy levers to protect the design of competing or compatible products,[65] they showed a less adaptive approach to a different class of competitors: those in the business of repairing computers and software. A series of cases in the 1990s found infringement based on the use of software by employees of indepen-

63. *See* Computer Assoc. Intern., Inc. v. Altai, Inc. 982 F.2d 693, 709–10 (2d Cir. 1992) (discussing elements of the "[f]iltration" stage of analysis guided by "[e]xternal [f]actors").

64. 17 U.S.C. §§ 1201–1205 (2000).

65. Courts have also relied upon copyright misuse for this purpose. *See, e.g.*, DSC Communications Corp. v. DGI Technologies, Inc., 81 F.3d 597, 601 (5th Cir. 1996) (suggesting that copyright misuse may occur when copyright holder seeks to prevent copies made as a necessary step in developing compatible, non-infringing product).

dent service organizations, who loaded the software in order to repair the program or the machine in which it resided. In these cases, the courts applied copyright law in a far more formal manner, without careful attention to the unique features of software and the software markets.

The trend began with *MAI Systems Corp. v. Peak Computing, Inc.*[66] The plaintiff in *MAI* sold computer systems and licensed certain software running on those systems to end users under restrictive terms. The defendant Peak maintained and repaired computer systems, including systems sold by MAI. In maintaining and repairing MAI's computers, Peak often ran MAI's operating software, which was loaded on MAI's computers. MAI's suit against Peak included, among other things, a novel copyright claim: MAI argued that, by simply turning on MAI's computers, Peak loaded a copy of MAI's operating system software into the random access memory (RAM) of the computer without authorization and thereby infringed on MAI's exclusive right to reproduce the work.[67]

The Ninth Circuit accepted this reasoning and held that Peak, by loading the software into the RAM of the computer in the course of running it, had made an unauthorized, infringing copy.[68] In reaching that decision, the panel examined the statutory definition of a "cop[y]" as a material object in which the copyrighted work is "fixed."[69] A work is "fixed," in turn, if it is "sufficiently permanent or stable to . . . be perceived . . . for a period of more than transitory duration."[70] The panel reasoned that, once Peak turned on the computer, the copyrighted software was copied into the RAM and was accessible so long as the computer was turned on and the program was running.

The decision in MAI has been extensively criticized for being overly formalistic and disregarding unique features of computer software and digital technology.[71] While most of the criticism has

66. 991 F.2d 511 (9th Cir. 1993).

67. *Id.* at 517–19.

68. *Id.*

69. 17 U.S.C. § 101 (2000).

70. *Id.*

71. R. Anthony Reese, *The Public Display Right: The Copyright Act's Neglected Solution to the Controversy Over RAM "Copies"*, 2001 U. ILL. L. REV. 83, 138–48 (2001); Joseph P. Liu, *Owning Digital Copies: Copyright Law and the Incidents of Copy Ownership*, 42 WM. & MARY L. REV. 1245, 1255, 1258 (2001); *see* James Boyle, *Intellectual Property Policy Online: A Young Person's Guide*, 10 HARV. J.L. & TECH. 47, 85–90 (1996); Jessica Litman, *Reforming Information Law in Copyright's Image*, 22 U. DAYTON L. REV. 587, 603 (1997); Mark A. Lemley, Dealing with Overlapping Copyrights on the Internet, 22 U. DAYTON L. REV. 547, 551-52 (1997).

focused on the court's interpretation of the term "copy," the literal terms of the Copyright Act may have nonetheless required that interpretation.[72] The real problem with the decision lay in its failure to consider that the "copies" at issue in that case arguably presented little or no threat to copyright incentives, because they did not serve as substitutes for either the original operating system or any potential derivative.[73] To the contrary, the RAM copies at issue in *MAI* were created as an incidental and necessary step in using an existing copy of software that customers had paid for and expected to be able to use.[74] The "copies" at issue had no impact on the market for MAI's software, but had a critical bearing on the related but distinct market in computer servicing. By finding these "copies" infringing, the ruling effectively gave MAI control over the service market, not based on any reasoned analysis, but based on an apparent perception that the copyright act required such a result.

Even if the *MAI* court was correct in its interpretation of "copy," the case law discussed above suggests that the court had other options to adapt copyright doctrines to the special facts of that case. The court could, for example, have adopted a fair use or "privileged use" approach that considered whether an injunction against the repair-related copying would serve or hinder copyright's underlying goals.[75] It could, in other words, have balanced the interest of the user in accessing the functionality of MAI's programs against MAI's interest in protecting against exploitation of its ex-

72. Indeed, CONTU assumed that RAM copies would satisfy the statutory definition of "copy," and for that reason incorporated the § 117 exception that allowed software owners to make copies as necessary to operate their machines. *CONTU Report*, *supra* note 15, at 12–13 (advocating exceptions for copies that are "essential step[s] in the utilization of . . . computer program[s]"). Virtually all of the courts considering the issue have agreed that RAM copies fit the statutory definition. *See, e.g.*, Stenograph L.L.C. v. Bossard Assocs., 144 F.3d 96, 102–03 (D.C. Cir. 1998); Triad Sys. Corp. v. Southeastern Express Co., 64 F.3d 1330, 1334–35 (9th Cir. 1995); Advanced Computer Services of Mich., Inc. v. MAI Sys. Corp., 845 F. Supp. 356, 362–63 (E.D. Va. 1994); Marobie-FL, Inc. v. Nat'l Ass'n of Fire and Equip. Distribs., 983 F. Supp. 1167, 1176–78 (N.D. Ill. 1997); *In re* Indep. Serv. Orgs. Antitrust Litig., 910 F. Supp. 1537, 1541 (D. Kan. 1995).

73. This may have resulted from the fact that the defendant never presented a fair use argument to the court, instead resting its defense on the definition of "copy" and on the narrow statutory exemption provided in § 117. *See MAI Systems*, 991 F.2d at 517–19.

74. *Cf.* Lewis Galoob Toys, Inc. v. Nintendo of Am., Inc., 964 F.2d 965, 971–72 (9th Cir. 1992) (holding that fair use protected users' ability to enjoy work that they paid for).

75. *See* Chad G. Asarch, Note, *Is Turn About Fair Play? Copyright Law and the Fair Use of Computer Software Loaded into RAM*, 95 MICH. L. REV. 654, 661 (1996) (arguing that use of software by ISOs should qualify as fair use).

pression. The fair use argument was apparently not raised in *MAI*. Just a few months later, however, in *Triad Systems Corp. v. Southeastern Exp. Co.*,[76] the Ninth Circuit rejected a fair use defense on nearly identical facts.

In *Triad* the Ninth Circuit again found an independent service organization liable for loading RAM copies into machines in the course of repair. The *Triad* defendants were charged with loading two kinds of software: operating system software and separate diagnostic programs that were designed specifically for purposes of computer repair. The defendants in *Triad* relied upon *Sega* for the proposition that their non-exploitative copies should be protected as fair use. The court disagreed, for two reasons. First, it found the service-related copies "wholly unlike the reverse-engineering in *Sega*," because unlike Accolade, the *Triad* defendants did not make the copies for the purpose of making their own transformative, creative works.[77] Second, the court found that the defendants were usurping Triad's legitimate market for service and licensing revenues.[78] The court assumed, in this analysis, that the protection of servicing revenues was a legitimate market protected by the Copyright Act. And yet the servicing of computers has much in common with the acts that courts in *Computer Associates*, *Lotus*, and *Sega* had found important to protect: acts that access the functional aspects of computer programs in order to make them work for the user's purpose.

The formalism of *MAI* and *Triad* differs markedly from the more adaptive and broad-ranging analysis presented by the software cases discussed in the previous section. In those cases, the courts were willing to interpret existing copyright doctrines in a flexible manner, to take into account the unique features of computer software and the need of users and competitors to access the functionality of the programs. In *MAI* and *Triad*, by contrast, the Ninth Circuit simply assumed that service-related copies lay within the copyright holder's exclusive rights and refused to consider the kinds of concerns about lock-in and access to functional features that had so concerned the courts in these earlier cases.

In 1998, Congress overruled the specific result in these cases by enacting a privilege for authorized repair agents to load operating

76. 64 F.3d 1330 (9th Cir. 1995).

77. *Id.* at 1336.

78. *See id.* at 1337 ("Southeastern is getting a free ride . . . . Triad is entitled to licensing fees from Southeastern and other ISOs that make use of Triad's software in servicing Triad computers.").

system software in the course of repairing users' machines.[79] By passing a specific and narrow statutory privilege, however, Congress left in place the presumption of infringement when the privilege does not apply. Computer companies have continued to use copyright law to restrict the ability of third-party service companies to service both software[80] and hardware.[81] The decisions have rested, not on a conclusion that the challenged use could threaten incentives in the copyright holder's primary market, but on a more mechanical assessment of the fact of copying and the inapplicability of the computer repair exemption.[82]

We do not argue for an absolute privilege for computer repair, as there may well be cases in which a repairer's use of software threatens incentives in the market for that software.[83] On the other hand, there is evidence to suggest that at least some of the repair cases raised copyright claims as a pretext to foreclose competition in the service market.[84] In these cases, we argue, it is appropriate to

---

79. 17 U.S.C. § 117(c) (2000) provides:

> Notwithstanding the provisions of section 106, it is not an infringement for the owner or lessee of a machine to make or authorize the making of a copy of a computer program if such copy is made solely by virtue of the activation of a machine that lawfully contains an authorized copy of the computer program, for purposes only of maintenance or repair of that machine, if – (1) such new copy is used in no other manner and is destroyed immediately after the maintenance or repair is completed; and (2) with respect to any computer program or part thereof that is not necessary for that machine to be activated, such program or part thereof is not accessed or used other than to make such new copy by virtue of the activation of the machine.

80. PracticeWorks, Inc. v. Prof'l Software Solutions of Illinois, Inc., 2004 WL 1429955 at *5–6 (D. Md. 2004) (holding that § 117(c) applies only to hardware repair and does not exempt copies made in the course of repairing software).

81. *See* Storage Tech. Corp. v. Custom Hardware Eng'g & Consulting, Inc., 2004 WL 1497688 at *3–5 (D. Mass. 2004) (mem.) (granting preliminary injunction against defendant's infringement when defendant performed repairs on plaintiff's computer data storage systems by copying the "Maintenance Code," which plaintiff had sought to protect as an exclusive diagnostic tool).

82. *Id.* at 3–4 ("Neither the statutory language nor its legislative history is expansive enough to safeguard [defendant's] use of plaintiff's program."). *See generally* David Nimmer, *Codifying Copyright Comprehensibly*, 51 UCLA L. REV. 1233 (2004).

83. Indeed, *Triad* may have been such a case, at least to the extent that the defendants copied special diagnostic software that was designed for use by the plaintiff in its servicing operations.

84. *See* DSC Communications Corp. v. DGI Technologies, Inc., 81 F.3d 597, 601 (5th Cir. 1996) (indicating that a court may find misuse where copyright holder's claim of infringement was pretext to impede competition in complementary market); Dan L. Burk, *Anticircumvention Misuse*, 50 UCLA L. REV. 1095,

turn to copyright law's policy levers, and to ask whether enjoining the use would promote or impede the law's underlying goals.

Both the fair use doctrine and the copyright misuse doctrine provide potential vehicles for accomplishing this result.[85] As *Sega* established, fair use ought to consider not just the defendant's act of copying in the abstract, but the nature of the defendant's use, including whether it was designed to educe some program functionality and whether it supplanted the market for the copyrighted expression. Misuse likewise bars the assertion of copyright rights by those who seek to foreclose competition in unrelated markets. In one sense, misuse and an exploitation-oriented fair use are flipsides of the same coin: they seek to limit the extension of copyright beyond what is necessary to preserve incentives in the market for expression.

B. The Digital Millennium Copyright Act

Judicial deployment of flexible policy levers in software copyright cases has also been limited in more recent years in the wake of Congress's passage of the DMCA. Enacted in response to the perceived threat to copyright holders posed by digital technology, the DMCA contains a number of provisions designed to help copyright owners use technology to prevent unauthorized copying. Specifically, the DMCA imposes liability for circumventing technologies that control access to copyrighted works.[86] It also bans the distribution of certain technologies or devices that enable circumvention.[87] The DMCA thus effectively creates a new entitlement, namely a right to prevent circumvention.

In creating this new right, Congress appears to have chosen to restrict the traditional role courts had played in crafting defenses to copyright liability.[88] Under conventional copyright analysis, courts used the fair use doctrine to adapt and adjust the scope of copy-

---

1124–31 (2003) (discussing development and application of copyright misuse doctrine).

85. The inquiry, of course, is far from straightforward, which may explain courts' reluctance to enter the fray. *Cf.* David McGowan, *Networks and Intention in Antitrust and Intellectual Property*, 24 J. CORP. L. 485, 494-96 (1999) (discussing the complexity of claims based on improper extension of intellectual property rights).

86. 17 U.S.C. §1201(a)(1) (2000).

87. *Id.* at §1201(a)(2).

88. *See* Universal City Studios, Inc. v. Corley, 273 F.3d 429, 443 (2d Cir. 2001) (holding that the DMCA does not allow circumvention for fair use purposes). *But see* Pamela Samuelson, *Intellectual Property and the Digital Economy: Why the Anti-Circumvention Regulations Need to Be Revised*, 14 BERKELEY TECH. L.J. 519, 539 (1999) (suggesting that DMCA allows courts to "distinguish between circumvention aimed

right protection in a case-by-case manner. Rather than relying on fair use, in creating the DMCA Congress substituted a number of statutory exemptions for specific uses. For example, the DMCA contains exemptions specifically permitting certain forms of encryption research, security testing, and reverse engineering for purposes of software interoperability.[89] It also gives the Librarian of Congress authority to promulgate further exemptions.[90]

The DMCA represents a departure from prior practice in two ways that are relevant to this paper. First, it includes provisions that directly implicate the software markets. More specifically, the statutory exemptions for encryption research and reverse engineering recognize some of the unique needs and features of the software markets. Second, the DMCA limits the role of the courts in adapting copyright doctrines to the specifics of computer software markets. It does so by replacing the flexible policy lever of fair use with narrower statutory and administrative exemptions. The DMCA is thus an example of a statutory, rather than judicial, strategy for tailoring copyright law to the specifics of computer software.

The judicial response to the DMCA in computer software cases nicely highlights some of the potential drawbacks of such a strategy. A number of initial district court cases applied the literal terms of the DMCA in a formalistic manner to situations that were clearly not contemplated by Congress.[91] In so doing, these courts did not carefully consider the unique circumstances surrounding computer software. Some of these cases were subsequently re-examined on appeal, and the appellate courts adopted a far more flexible and contextual approach, hearkening back to the earlier line of cases. We examine the district court opinions here, as an example of the trend away from flexible judicial adaptation. We then consider the appellate court decisions in the following section, as a partial response to this trend.

In *Lexmark Int'l v. Static Control Components*,[92] a laser printer manufacturer, Lexmark, attempted to use the DMCA to prevent a

---

at getting unauthorized access to a work and circumvention aimed at making non-infringing uses of a lawfully obtained copy.").

89. 17 U.S.C. § 1201(d) – (j) (2004).

90. *Id.* at §1201(a)(1)(B)–(C).

91. Davidson & Assocs. v. Internet Gateway, 334 F. Supp. 2d 1164 (E.D. Mo. 2004); Chamberlain Group v. Skylink Techs., Inc., 292 F. Supp. 2d 1040 (N.D. Ill. 2003), *aff'd* 381 F.3d 1178 (Fed. Cir. 2004); Lexmark Int'l, Inc. v. Static Control Components Inc., 253 F. Supp. 2d 943 (E.D. Ky. 2003), *rev'd*, 387 F.3d 522 (6th Cir. 2004).

92. 253 F. Supp. 2d at 943. *See generally* Daniel C. Higgs, *Lexmark International, Inc. v. Static Control Components, Inc. & Chamberlain Group, Inc. v. Skylink Technolo-*

third-party vendor from offering compatible toner cartridges. Lexmark used a technology to ensure that only authorized toner cartridges would work with its printers. Each authorized toner cartridge contained a computer chip with a small amount of software code on it (approximately 55 bytes). This code contained an authorization sequence, which told the printer that the cartridge was compatible. The defendant Static Control sold computer chips that mimicked the Lexmark chips.[93]

Lexmark sued, alleging both copyright infringement and violation of the DMCA. With respect to the copyright claim, Lexmark argued that Static Control's chips contained identical copies of the software code, including some elements that were not required for interoperability.[94] In its DMCA claim, Lexmark argued that each printer contained within it software that enabled operation of the printer.[95] The authentication process between the toner cartridge and the printer was therefore a "technological measure" that controlled access to a copyrighted work, namely the software residing in the printer. Therefore, by using the substitute chip to access the printer software, Static Control was circumventing a technological measure and therefore liable under the DMCA.[96]

Chamberlain v. Skylink[97] involved a similar use of the DMCA to prevent competition in a related hardware market, in this case, the market for compatible garage door opener remote controls. Plaintiff Chamberlain manufactured garage door openers and remote controls. On some of these models, Chamberlain employed a technology that increased the security of these garage door openers by varying the signal sent from the remote to the garage door opener. Defendant Skylink sold a universal remote control that was compatible with certain Chamberlain garage door openers. Chamberlain sued Skylink under the DMCA, arguing that its remote control contained a technological measure that controlled access to software residing on the garage door opener. That is, without the remote control and the technological measure embedded within it, one

gies, Inc.: The DMCA and Durable Goods Aftermarkets, 19 BERKELEY TECH. L.J. 59, 83 (2004).

93. *Lexmark*, 253 F. Supp. 2d at 955.

94. *See id.*

95. *See id.* at 955–56 (discussing Lexmark's "Toner Loading Program" and "authentication sequence," and concluding that defendant's toner cartridges "circumvent[ed]" these programs "[e]ach time a consumer install[ed]" one into a Lexmark printer).

96. *Id.*

97. Chamberlain Group, Inc. v. Skylink Techs., Inc., 292 F. Supp. 2d 1040 (N.D. Ill. 2003), *aff'd* 381 F.3d 1178 (Fed. Cir. 2004).

could not run the software residing on the garage door opener. Selling the universal remote control thus constituted distribution of a device that enabled circumvention.[98]

In both *Lexmark* and *Skylink*, the district courts interpreted the DMCA in a narrow and literal fashion, providing some support to the attempt to use copyright law to prevent competition in a separate market for compatible hardware. In *Lexmark*, the court accepted the plaintiff's theory of DMCA liability.[99] The court further rejected the fair use defense to copyright liability, reading the *Sega* defense in a narrow fashion.[100] In *Skylink*, the district court denied liability under the DMCA, but on grounds that lent implied support to the plaintiff's reading of the DMCA.[101]

While such a narrow interpretation of the DMCA may be supported by the literal text of the statute, the use of the DMCA for these purposes was clearly outside the DMCA's intended scope. The original purpose of the DMCA, whatever one's view of its merits, was to protect copyrighted content such as movies, music, and software.[102] In the two cases above, however, the DMCA was used for a completely unrelated purpose, i.e., to prevent a third party from selling compatible hardware. For example, Lexmark almost certainly did not care about the use or redistribution of its printer software; it was concerned about protecting its market for cartridges that worked with its printers. The company managed to in-

98. *Id.*, at 1043–44.

99. *Lexmark*, 253 F. Supp. 2d at 966–71. In so doing, the court rejected the argument that defendant's use fell within the exemption for reverse engineering. The court held that the exemption was only available to enable interoperability of "independently created computer program[s]." Since Static Control had copied all of the code on the chips, it had not created its own computer program and therefore could not take advantage of the exemption. *Lexmark*, 253 F. Supp. 2d at 970–71.

100. *See id.* at 960–62. The court had found that Static Control, in copying all of the software on the toner cartridges, had copied more than necessary to enable interoperability. The court found that 7 of the 55 bytes in the program were used as part of Lexmark's authentication sequence. *Id.* at 950. While the miniscule size of the program makes this holding appear peculiar, it finds some support in *Sega* and related cases, which had conditioned their fair use finding on the fact that the defendant had copied no more than necessary to achieve interoperability. *Id.* at 960–62.

101. The court ultimately held that owners of the compatible garage door remote controls were impliedly authorized to engage in circumvention, and therefore Skylink could not be liable under the DMCA. *Chamberlain*, 292 F.Supp.2d at 1043–45, The opinion suggested, however, that without such consent the use of the remote control might in fact lead to DMCA liability.

102. *See* S. REP. NO. 105–190, at 1–23 (1998), 1998 WL 239623; H.R. REP. NO. 105-551 (II), at 29–34 (1998), 1998 WL 414916.

voke the DMCA for this unrelated purpose because software happened to be involved in the interface between these compatible products. Thus, the literal application of the terms of the DMCA in *Lexmark* led to a result that was at odds with the broader purpose of the Act.

As noted below,[103] the courts of appeals in these two cases appeared to recognize the disjunction between these uses of the DMCA and the DMCA's underlying purpose. And in their interpretations of the DMCA, these two courts backed off from the narrow, more literal approach. For present purposes, however, it is important to note that the two district court cases initially signaled a very narrow approach to DMCA cases, one that did not take into consideration the wider context or the unique features of computer software.

Indeed, at least one other district court has similarly interpreted the DMCA narrowly to prevent the creation of compatible products, this time software rather than hardware. In *Davidson Associates v. Internet Gateway*,[104] the plaintiff Blizzard published popular PC game software and provided an online service, Battle.net, through which game owners could play each other on the Internet. When a game owner logged onto the game service, the service required an encrypted unique authorization key located on each game's CD-ROM, in order to prevent piracy of the game.[105] The defendants in the case created software for an alternative on-line service, which allowed individuals to play each other over the Internet without going through Battle.net. The alternative service did not require transmission of a unique authorization key from the CD.[106] Blizzard sued on various grounds, including copyright infringement, DMCA infringement, and breach of contract.

The district court granted summary judgment to the plaintiffs on each of the above claims. With respect to the DMCA claim, the court held that the authentication sequence was a technological measure that effectively controlled access to the portions of the gaming software that permitted multi-player play. Defendants, in creating their own software, which disregarded the authentication sequence, had circumvented the technological measure.[107] Moreo-

103. *See infra* Part IV.B.

104. 334 F. Supp. 2d 1164 (E.D. Mo. 2004). *Cf.* 321 Studios v. Metro Goldwyn Mayer Studios, Inc., 307 F. Supp. 2d 1085 (N.D. Cal. 2004) (holdindg that technology for making backup copies of DVDs violated DMCA).

105. *Id.* at 1169.

106. *Id.* at 1172–73.

107. *Id.* at 1184.

ver, the court held that the defendants' actions fell outside the statutory exemption for reverse engineering for two reasons. First, because the defendants' service mimicked the Battle.net service, the software was not "independently created" as required by the exemption. Second, the defendants' service facilitated piracy by permitting use of unauthorized copies of the software that lacked the unique key, and therefore was not designed for the "sole purpose" of interoperability.[108]

As with the earlier opinions, the district court opinion in *Davidson* reflects a narrow and literal application of the terms of the DMCA. Unlike the earlier cases, *Davidson* did not involve a completely unforeseen extension of the DMCA to hardware. Rather, this situation was arguably what Congress had in mind when it crafted the reverse engineering exemption. Moreover, the claim based on the failure to recognize the authorization key more directly implicates the copyright interest that the DMCA was intended to protect. So it is possible that a more contextual analysis could still result in a verdict for the plaintiffs in the case. Nevertheless, in applying the DMCA, the district court in fact gave scant consideration to the kinds of policies reflected in the earlier cases.

The three district court opinions discussed above thus illustrate an approach to software cases that is quite different from the prior, more adaptive approach. In each of these cases, the courts gave little consideration to the unique aspects of computer software or the concerns about edification, compatibility, and use that the *Altai*, *Lotus*, and *Sega* courts had found so important. In part, this reflects the relatively narrower room to maneuver provided to the courts by Congress in the DMCA, a statute that clearly opted for legislative rather than judicial solutions. This in turn suggests that such an approach, insofar as it leads to results at odds with underlying copyright policy, has significant drawbacks.[109]

C. The Extension of Contract Law

The DMCA is not the only tool that copyright holders have used to recalibrate the balance of rights among themselves, their users, and their competitors. With the near-universal shift to license rather than sale, software developers have turned to contract law to control the use that others make of their products. Courts

---

108. *Id.* at 1185. The court further held that the defendants had waived their fair use defense to copyright liability, insofar as they had agreed to the terms of the end-user license. *See infra* Part III.C.

109. *See* Burk & Lemley, *supra* note 1, at 1637–38 (arguing against statutory tailoring in the patent context).

have shown remarkable deference to these contracts, enforcing them even when their terms conflict directly with an established copyright norm. If this trend continues, the contextual, nuanced approach to software copyright reflected in *Altai*, *Sega*, and *Lotus* may be rendered irrelevant in the future.

In *Bowers v. Baystate Technologies*,[110] for example, the Federal Circuit enforced a contract provision that prohibited reverse engineering of software for any purpose whatsoever. The defendants in *Bowers* argued that the provision was preempted by copyright law, insofar as copyright's fair use doctrine, as interpreted in *Sega*, grants a privilege to reverse engineer. Thus, defendants argued, any attempt to eliminate that privilege via contract conflicted with underlying copyright policies and was preempted.[111]

The Federal Circuit rejected this argument and found the provision enforceable. The court analyzed the provision through the Copyright Act's express preemption clause,[112] finding that, because contract claims differed substantially from copyright claims, they were not expressly preempted.[113] Notably, the court did not address the possibility that the provision might be preempted under a theory of "conflict" preemption.[114] The court thus effectively held that contract terms are immune from copyright scrutiny.[115]

The court in *Davidson*,[116] discussed above, reached essentially the same result. In addition to the DMCA claim, the plaintiffs in that case sought to enforce a provision in the shrinkwrap license that barred reverse engineering. The district court in *Davidson* upheld the contract claim and found the provision enforceable, rejecting arguments based on preemption, unconscionability, and lack of enforceability.[117] The preemption analysis in *Davidson*

110. 320 F.3d 1317 (Fed. Cir. 2003).

111. *Id.* at 1323. *See also* Vault Corp. v. Quaid Software, Ltd., 847 F.2d 255, 270 (5th Cir. 1988) (finding a state provision preempted).

112. 17 U.S.C. § 301 (2004).

113. *Bowers* at 1323–28. *See also* ProCD, Inc. v. Zeidenberg, 86 F.3d 1447 (7th Cir. 1996) (finding no preemption). *Cf. Bowers*, 320 F.3d at 1335-38 (Dyk, J. dissenting in part) (contending that anti-reverse engineering provision was preempted under 17 U.S.C. § 301).

114. *See* Mark Lemley, *Beyond Preemption: The Law and Policy of Intellectual Property Licensing*, 87 CAL. L. REV. 111, 139–44 (1999) (describing two forms of preemption in copyright law).

115. *Bowers*, 320 F.3d at 1323–26 (following the rule developed in other circuits that "the Copyright Act does not preempt contractual constraints on copyrighted articles").

116. Davidson & Assocs. v. Internet Gateway, 334 F. Supp. 2d 1164 (E.D. Mo. 2004).

117. *Id.* at 1174–80.

tracked the discussion in *Bowers*. Finally, the *Davidson* court rejected a copyright misuse argument.[118]

The precise relationship between copyright law and contract law is a complicated one, and has been much discussed in the literature.[119] For the purposes of this paper, however, the trend toward easy enforcement of these provisions is relevant insofar as it has the potential to greatly reduce the ability of courts to flexibly adapt the rights of copyright owners in the context of computer software. In effect, the *Bowers* approach means that copyright holders can override any of the substantive limitations built into copyright simply by fiat.[120] As these provisions are inserted into mass-market shrinkwrap licenses, they effectively replace copyright law with what amounts to private legislation.[121] As others have observed, the shift to a contract regime in the software industry—particularly as supplemented by the DMCA—could well mean that the nuanced and balanced approaches to computer copyright discussed above will become obsolete and irrelevant. If courts cannot or will not change their absolutist approach to contract enforcement, the interests of users and competitors in accessing the functional aspects of software will suffer.

IV.
REINSTATING JUDICIAL ADAPTATION

Given the dynamic, functional and interoperative nature of software products, we believe that the earlier, more flexible and creative judicial attitude toward computer software resulted in sounder policies. The courts that grappled with and expressly considered the unique features of computer software were able to adapt copy-

---

118. *Id.* at 1182.

119. *See, e.g.*, Michael J. Madison, *Reconstructing the Software License*, 35 LOY. U. CHI. L.J. 275 (2003); Mark Lemley, *Beyond Preemption: The Law and Policy of Intellectual Property Licensing*, 87 CAL. L. REV. 111 (1999); Maureen O'Rourke, *Copyright Preemption After the ProCD Case: A Market-Based Approach*, 12 BERKELEY TECH. L.J. 53 (1997); Maureen A. O'Rourke, *Drawing the Boundary Between Copyright and Contract: Copyright Preemption of Software License Terms*, 45 DUKE L.J. 479 (1995).

120. In *Bowers*, Judge Dyk warned that:

> If by printing a few words on the outside of its product a party can eliminate the fair use defense, then it can also, by the same means, restrict a purchaser from asserting the "first sale" defense, embodied in 17 U.S.C. § 109(a), or any other of the protections Congress has afforded the public in the Copyright Act.

Bowers, 320 F.3d at 1337 (Dyk, J., concurring in part and dissenting in part).

121. *See* Mark Lemley, *Beyond Preemption: The Law and Policy of Intellectual Property Licensing*, 87 CAL. L. REV. 111, 147–50 (1999); Madison, *supra* note 119.

right doctrines so as to protect the underlying incentives for creating the software while ensuring that copyright protection did not have adverse effects, such as distorting competition. By contrast, the courts that applied copyright doctrines (including preemption) formalistically and mechanically often reached results that potentially hindered free competition and that fit poorly with copyright law's underlying goals. Attempts by Congress to restrict the room for judicial adaptation in software cases have met with similar results.

A. Signs of Promise

There are signs that the federal courts are beginning to recognize a need to restore flexibility and import into software cases more express consideration of the policy interests unique to computer software. As mentioned above, the district court opinions in both *Lexmark* and *Skylink* were subsequently appealed. In both of those cases, the courts of appeal issued opinions that differed markedly from the district court opinions and evinced a greater willingness to inject policy considerations into their interpretation of the DMCA. In fact, both of these opinions are notable for the extent to which they effectively create policy levers in the DMCA where arguably none existed before.

In *Lexmark*, for example, the Sixth Circuit panel overturned the district court's rulings on both DMCA and copyright liability.[122] With respect to the copyright claim, the court pointed out that the small, 55-byte toner-loading program found on the toner cartridges was functional and therefore not copyrightable.[123] Although the court noted that the unavailability of a copyright for the Toner Loading Program made it unnecessary to consider the fair use defense, the court did suggest that this defense would be available to the defendants. In this discussion, the court interpreted the *Sega* privilege in a far more expansive manner than the district court, in light of the policies served by copyright.[124]

122. Lexmark Int'l Inc. v. Static Control Components Inc., 387 F.3d 522 (6th Cir. 2004) (vacating the district court's preliminary injunction).

123. *See id.* at 539–44.

124. *See id.* at 544–45. The concurring opinion of Judge Merritt evinced an even broader understanding of the fair use privilege, urging the court to make clear that "in the future companies like Lexmark cannot use the DMCA in conjunction with copyright law to create monopolies of manufactured goods for themselves just by tweaking the facts of this case: by, for example, creating a Toner Loading Program that is more complex and 'creative' than the one here . . . ." *Id.* at 551.

With respect to the DMCA claim, the court held that the authentication sequence did not "effectively control access" to the software residing on the printer.[125] In reaching this conclusion, the court refused to construe the term "access" in a literal manner, as suggested by the plaintiffs. Instead, the court expressly interpreted the term in light of the copyright-based policies of the DMCA and found that, while the authentication sequence controlled whether the software would run or not, it did not prevent individuals from reading or downloading the software, as the software sat in unencrypted form on the printer. Thus, the authentication sequence did not prevent "access" or copying in any meaningful copyright-related sense.[126] The court further held that, even if the DMCA could be construed to reach the facts of this case, Static Control's actions may have fallen within the exemption for reverse engineering.[127]

The court of appeals in *Skylink*[128] injected flexibility into its consideration of the DMCA in an even more dramatic fashion. The Federal Circuit affirmed the lower court's decision in *Skylink* that the compatible remote control did not violate the DMCA. However, the court did not adopt the lower court's reasoning that purchasers of the compatible remote controls were impliedly authorized to engage in circumvention. Rather, the court held that, even if these purchasers were not authorized, the use of the compatible remote control did not constitute an act of circumvention. In reaching this result, the court rejected a literal application of the terms of the DMCA and expressly crafted a "rule of reason" for circumvention liability. Under this rule, an act of circumvention must be "reasonably related" to an underlying copyright interest before liability will attach.[129] In this case, the circumvention at issue was wholly unrelated to any copyright interest, in that the plaintiffs were not concerned with controlling access to, or preventing piracy of, the software residing on the garage door opener.

Both of these cases are notable to the extent that they rely upon, and even create, "policy levers" in the DMCA where none

125. *See id.* at 546–50.

126. *See id.*

127. In response to Lexmark's argument that "if independently created programs do exist . . . they must have existed prior to the 'reverse engineering' of Lexmark's Toner Loading Program," the court responded that "nothing in the [DMCA] precludes simultaneous creation of an interoperability device and another computer program; it just must be 'independently' created." *Id.* at 550–51.

128. Chamberlain Group, Inc. v. Skylink Techs., Inc. 381 F.3d 1178 (Fed. Cir. 2004).

129. *Id.* at 1202–03.

existed before. In each of these cases, no obvious doctrine in the DMCA gave the courts an easy way to interpret the statute flexibly in order to avoid what appeared to be an unanticipated extension of the DMCA. Indeed, the district court opinions were defensible under a literal reading of the DMCA. Yet the appeals courts in both of these cases reached out to inject the necessary flexibility into the statute, in one case by interpreting the term "access" in a particular way, in the other case by crafting a "rule of reason." The cases indicate that the need for flexibility in software cases still exists and that courts are still willing to respond to this need.

The creation of such flexibility is not without its costs. In particular, the rule of reason created by the *Skylink* court raises a number of questions. In some ways, the result appears inconsistent with prior case law.[130] The result also raises difficult questions about how this provision relates to other provisions of the DMCA.[131] Finally, such a rule may well create a non-trivial amount of uncertainty with respect to future cases, as the court itself acknowledged.[132] More generally, increasing judicial flexibility tends, all else being equal, to make the rights and duties of market participants more uncertain. One benefit of a clear rule is that it is certainly easier to administer.[133]

Yet as we have attempted to show above, the particular, inflexible rules that have been implemented in the software context have come at significant cost to important interests in edification, compatibility, and use of software. Nor should this be surprising, given

130. *See* 321 Studios v. Metro Goldwyn Mayer Studios, Inc., 307 F. Supp. 2d 1085, 1095 (N.D. Cal. 2004); Universal City Studios, Inc. v. Reimerdes, 111 F. Supp. 2d 294, 318 (S.D.N.Y. 2000), *aff'd sub nom.* Universal City Studios, Inc. v. Corley, 273 F.3d 429 (2d Cir. 2001); Sony Computer Ent. Am. Inc. v. Gamemasters, 87 F.Supp.2d 976, 987 (N.D. Cal. 1999); Pearl Investments, LLC v. Standard I/O, Inc., 257 F. Supp. 2d 326, 349–50 (D. Me. 2003); RealNetworks, Inc. v. Streambox, Inc., 2000 WL 127311, at *1–2, 6–7 (W.D. Wash. 2000) (No. 2:9CV02070).

131. In particular, the result in *Skylink* raises a question about how liability under § 1201(a)(1) for circumvention of access control technologies differs from liability under § 1201(b) for circumvention of technologies that protect an exclusive right provided under copyright law. *See generally* R. Anthony Reese, *Will Merging Access Controls And Rights Controls Undermine The Structure Of Anticircumvention Law?*, 18 BERKELEY TECH. L.J. 619, 622–46 (2003).

132. *Chamberlain*, 381 F.3d at 1202–03.

133. *See* Burk & Lemley, *supra* note 1, at 1638-39 & n.224 (noting "long-running debate over the comparative merits of rules versus standards" and citing references); Dan Burk, *Muddy Rules for Cyberspace*, 21 Cardozo L. Rev. 121 (1998); Edward Lee, Rules and Standards for Cyberspace, 77 Notre Dame L. Rev. 1275 (2002). *Accord* David Nimmer, *Codifying Copyright Comprehensibly*, 51 UCLA L. REV. 1233 (2004).

the complex and dynamic nature of the technology and software markets, which time and again generate new circumstances that far exceed what Congress anticipated. It may be that, at some point in the future, these markets may settle down and we may have enough information to more confidently provide clearer rules. The cases discussed in the previous section indicate, however, that we are not yet there, and that some form of judicial flexibility will remain necessary.

B. Other Avenues

While the signs described above are promising, more could be done, both legislatively and judicially, to increase the ability of courts to import policy concerns into software cases. First, copyright legislation implicating software should build in more discretion for courts to continue to adapt copyright doctrines to the computer software markets. We are nowhere near the point where we know enough about the future direction of the software markets and technology to be confident that we can address all future developments in a comprehensive statute.[134] Thus, in the case of the computer maintenance exemption, it may have been better to have crafted a flexible standard considering the effects of such competition on the primary software market, rather than the more specific approach that Congress took in the computer repair exemption.[135]

Similarly, in the context of the DMCA, the highly detailed exemptions have, time and again, proven far narrower than anticipated. It would have been better to give courts increased discretion to craft exemptions in light of changing circumstances. Thus, a broad fair use defense to circumvention would have permitted more judicial flexibility and would have been consistent with past judicial practice.[136] At the very least, broader, standards-based language in the specific exemptions would have given courts more room to interpret the statute to account for unintended cases like

134. Moreover, as Burk & Lemley point out, statutory tailoring is more subject to industry capture. Burk & Lemley, *supra* note 1, 1634–35. The DMCA may be a perfect example of such capture. *See, e.g.*, Pamela Samuelson, *Intellectual Property and the Digital Economy: Why the Anti-Circumvention Regulations Need to be Revised*, 14 BERKELEY TECH. L.J. 519 (1999).

135. *See* Nimmer, *supra* note 82.

136. Benefit Authors without Limiting Advancement or Net Consumer Expectations (BALANCE) Act of 2003, H.R. 1066, 108th Cong. §§ 2, 5 (2003) (proposing fair use defenses for circumvention where "such act is necessary to make a noninfringing use of the work under this title," and "the copyright owner fails to make publicly available the necessary means to make such noninfringing use without additional cost or burden to such person").

Lexmark and *Skylink*. More standards-based language might have obviated the strained approaches adopted by the appellate courts in those two cases.[137]

Even in the absence of legislative reform, courts could do more to consider and take express account of the unique aspects of computer software.[138] The *Skylink* and *Lexmark* opinions reflect an admirable attempt to make outcomes in DMCA cases bear some relationship to copyright law's underlying objectives. Other courts should follow the Federal Circuit's lead and interpret the relevant portions of the DMCA (whether the definition of "circumvention" or the reverse-engineering exemption) flexibly, with an eye to the underlying policy concerns.[139] If they do not, the legislature should step in to import some flexibility, or at least to address the most egregious examples of using the DMCA to achieve anticompetitive goals in a market only remotely related to copyright law.[140]

Finally, the importance of edification, interoperability, and use of functional aspects of software suggests that courts should think more seriously about preemption arguments in cases pitting copyright interests against contract terms. As Judge Dyk pointed out in dissent in *Bowers v. Baystate*, the alternative will allow copyright holders, "by printing a few words on the outside of [their] product[s]," to eviscerate "any . . . of the protections Congress has afforded the public in the Copyright Act."[141] Instead of formalistically enforcing all contract clauses involving software (such as the anti-reverse engineering clauses), courts should examine such clauses critically to determine whether widespread deployment of such clauses could have the effect of undermining the concerns raised above.

---

137. *See generally* Joseph P. Liu, *Regulatory Copyright*, 83 N.C. L. REV. 87 (2004) (addressing the need for more standards in high-technology areas); Dan L. Burk, Anticirumvention Misuse, 50 U.C.L.A. L. REV. 1095 (2003) (advocating development of misuse doctrine to limit overreaching by DMCA claimants).

138. *See* Higgs, *supra* note 92, at 83 (suggesting various avenues available for judicial limitation of inappropriate expansion of the DMCA).

139. The courts should also continue to exercise the discretion already granted to them under standard copyright law in the context of reverse engineering. *See Lexmark*, 387 F.3d at 551–53 (Merritt, J., concurring). *But see Davidson & Assocs. v. Internet Gateway*, 334 F. Supp. 2d 1164 (E.D. Mo. 2004) (failing to exercise discretion).

140. *See* Jacqueline Lipton, *The Law of Unintended Consequences: The Digital Millennium Copyright Act and Interoperability*, 62 WASH. & LEE L. REV. (forthcoming 2005) (advocating a legislative "carve out" to the DMCA in cases involving interoperable replacement parts for tangible goods, where copyrightable code is only incidental to the product).

141. 320 F.3d 1317, 1337 (Fed. Cir. 2003).

More broadly, courts should continue to be extremely sensitive to the importance of competition and the ways in which copyright protection of computer software can act to hinder competition. Both the *Lexmark* and *Skylink* cases are examples of a broader issue highlighted by *Sega*, *Connectix*, and other cases. Whether courts are interpreting narrow provisions like the DMCA or broader doctrines like fair use, they should constantly be sensitive to both the need for robust competition in the software and hardware markets, as well as the potential for misuse of copyright, DMCA, and contract law to prevent such competition.[142]

V.
CONCLUSION

The flexibility of copyright law offers numerous vehicles for protecting consumer and competitor interests in software copyright cases. Over time, courts learned to use these "policy levers" to achieve a level of balance in software copyright law. While the DMCA and contract law may not appear to offer the same level of flexibility, in fact each one offers the potential to consider the same kinds of interests that motivated the courts in the software cases in the early and mid-1990s. Courts should take advantage of these policy levers to restore balance to the relationship between software developers and the public users.

142. *See* Dan L. Burk, *Anticircumvention Misuse*, 50 UCLA L. REV. 1095 (2003).