

1999

## Intellectual Property, Contracts, and Reverse Engineering After ProCD: A Proposed Compromise for Computer Software

Anthony J. Mahajan

Follow this and additional works at: <https://ir.lawnet.fordham.edu/flr>



Part of the [Law Commons](#)

---

### Recommended Citation

Anthony J. Mahajan, *Intellectual Property, Contracts, and Reverse Engineering After ProCD: A Proposed Compromise for Computer Software*, 67 Fordham L. Rev. 3297 (1999).

Available at: <https://ir.lawnet.fordham.edu/flr/vol67/iss6/12>

This Article is brought to you for free and open access by FLASH: The Fordham Law Archive of Scholarship and History. It has been accepted for inclusion in Fordham Law Review by an authorized editor of FLASH: The Fordham Law Archive of Scholarship and History. For more information, please contact [tmelnick@law.fordham.edu](mailto:tmelnick@law.fordham.edu).

---

## Intellectual Property, Contracts, and Reverse Engineering After ProCD: A Proposed Compromise for Computer Software

### Cover Page Footnote

This Note is dedicated to my family in gratitude for their faithful encouragement and guidance throughout my life. I would also like to thank Lisa M. McIntyre for her unwavering support during the writing of this Note, and my friends for always brightening my day.

# INTELLECTUAL PROPERTY, CONTRACTS, AND REVERSE ENGINEERING AFTER *PROCD*: A PROPOSED COMPROMISE FOR COMPUTER SOFTWARE

Anthony J. Mahajan\*

Here's the formula for success in the coming era: Open, good.  
Closed, bad.<sup>1</sup>

## INTRODUCTION

The relation of computer software to the hardware components of a computer is analogous to the relationship between a blade and a razor.<sup>2</sup> Without software, a computer would not function in so many of the ways that modern users have come to appreciate, and may not function at all. Indeed, each and every program run on a computer transforms that computer into a new machine, one that follows the instructions dictated by the computer program.<sup>3</sup> Thus, one can easily comprehend the importance of software in the coming technological age.

When computers were first developed in the 1950s, a communitarian view of software development was the norm, and programmers openly shared their ideas with one another.<sup>4</sup> Program code, the instructions that tell a computer what to do,<sup>5</sup> was circulated without proprietary restriction.<sup>6</sup> Developers were free to use that code, incorporate it into their own programs, and adapt it to suit their programming needs.<sup>7</sup> A sense of pride and accomplishment, rather than the desire to make money, drove the creation of new software in this open and shared environment.<sup>8</sup> Then things changed. Corporations recognized the vast profitability of software, and they began to exer-

---

\* This Note is dedicated to my family in gratitude for their faithful encouragement and guidance throughout my life. I would also like to thank Lisa M. McIntyre for her unwavering support during the writing of this Note, and my friends for always brightening my day.

1. Peter Schwartz & Peter Leyden, *The Long Boom*, *Wired*, July 1997, at 115, 121 (noting the transition from an industrial to an information society).

2. See Harold Seneker & Jayne A. Pearl, *Software to Go*, *Forbes*, June 20, 1983, at 93, 93.

3. See Pamela Samuelson, *CONTU Revisited: The Case Against Copyright Protection for Computer Programs in Machine-Readable Form*, 1984 Duke L.J. 663, 680-81 (noting that computers are "chameleon-like" machines, and that the program determines what type of machine the computer will be).

4. See Steven Levy, *Code Warriors*, *Newsweek*, Jan. 18, 1999, at 60, 60 [hereinafter Levy, *Code Warriors*].

5. See *infra* note 53 and accompanying text.

6. See Steven Levy, *Hackers: Heroes of the Computer Revolution* 27-28 (1984).

7. See *id.*

8. See *id.* at 30-32.

cise tight controls over who was allowed access to their coveted code.<sup>9</sup> This marked a shift toward a closed proprietary model of software development.

Microsoft Corporation, one of the software development companies that capitalized upon this metamorphosis, is now on trial.<sup>10</sup> The government alleges that Microsoft has a monopoly in the operating systems market and uses it to control access by competitors to the software industry.<sup>11</sup> One reason for Microsoft's position of dominance is that its operating system has become the consumer standard.<sup>12</sup> Another contributing factor may be the expansion of intellectual property protection for computer programs.<sup>13</sup> Various aspects of computer programs may now receive copyright, patent, trade secret, and contract protection.<sup>14</sup> The cumulative effect of these rights generates concerns regarding the appropriate degree of protection for computer programs. Proponents of a high degree of proprietary protection argue that it will lead to better software because it gives a substantial incentive to developers to create new and innovative products.<sup>15</sup> Opponents argue that high levels of protection facilitate the seizure of material in the public domain and stifle competition.<sup>16</sup> Whether current proprietary protection for computer software should be limited to promote access and competition in the software industry is hotly debated, and is the subject of this Note.

---

9. See Paul Freiburger & Michael Swaine, *Fire in the Valley: The Making of the Personal Computer* 153-54 (1984); Levy, *Code Warriors*, *supra* note 4, at 60. Program code became a corporation's crown jewel, the gem to be locked away, safeguarded from prying eyes and hands. See *id.*

10. See *United States v. Microsoft Corp.*, 980 F. Supp. 537 (D.D.C. 1997); *United States v. Microsoft Corp.*, 159 F.R.D. 318 (D.D.C.), *rev'd*, 56 F.3d 1448 (D.C. Cir. 1995).

11. See Thomas A. Piraino, Jr., *An Antitrust Remedy or Monopoly Leveraging by Electronic Networks*, 93 Nw. U. L. Rev. 1, 3-5 (1998).

12. See *id.* at 3.

13. See Mark A. Lemley & David McGowan, *Could Java Change Everything?: The Competitive Propriety of a Proprietary Standard*, in *Second Annual Internet Law Institute* 453, 456 (PLI Pats., Copyrights, Trademarks, & Literary Prop. Course Handbook Series No. 520, 1998) [hereinafter Lemley & McGowan, *Competitive Propriety*].

14. See *infra* Part I.

15. See, e.g., Anthony L. Clapes, *Confessions of an Amicus Curiae: Technophobia, Law, and Creativity in the Digital Arts*, 19 U. Dayton L. Rev. 903, 906 (1994) (objecting to a reduced level of intellectual property protection for computer software); Arthur R. Miller, *Copyright Protection for Computer Programs, Databases, and Computer-Generated Works: Is Anything New Since CONTU?*, 106 Harv. L. Rev. 977, 988 (1993) (concluding that there is no reason to believe that courts are overprotecting software proprietors).

16. See, e.g., Michael J. Madison, *Legal-Ware: Contract and Copyright in the Digital Age*, 67 Fordham L. Rev. 1025, 1029 n.14, 1031 (1998) (arguing that the public domain cannot consist only of materials that information providers unilaterally choose to provide); Pamela Samuelson et al., *A Manifesto Concerning the Legal Protection of Computer Programs*, 94 Colum. L. Rev. 2308, 2365-71 (1994) (advocating a market-based approach to computer software protection).

Part I undertakes a comparative exploration of the intellectual property laws available to protect computer software. Part II argues that the balance between public and private rights of access to computer programs has been upset by the enforcement of license agreements that invariably prohibit reverse engineering, a process traditionally viewed as an appropriate means of promoting access and competition. Part III queries whether relying on the judiciary or the private market to restore the balance is the most efficient and desirable solution. Part IV concludes that a novel and intermediate approach is needed. This part suggests that Congress adopt a two-tiered standard, one that is emboldened by the Constitution: permit software producers to establish a level of true exclusive control over their works, but require the producers to disclose their source code after a pre-determined passage of time as a condition to maintaining federal copyright and patent protection.

### I. PROPRIETARY PROTECTION FOR COMPUTER PROGRAMS

The precise amount of proprietary protection available to computer programs is in flux. Courts differ regarding the extent of copyright, patent, trade secret, and contract protection properly afforded computer programs. Copyright and patent are federal statutes,<sup>17</sup> and trade secret and contract involve common law rights.<sup>18</sup> The overlap of both federal and state rights creates a comprehensive intellectual property scheme, and makes it less likely that a work will be pirated by second-comers. Yet, the intersection of federal and state rights is a troubled one, and, at times, the enforcement of various state rights may function to frustrate the objective of the federal statutes. This part examines and compares the evolving federal and state proprietary protections available to computer programs.

#### A. Copyright Protection for Computer Programs

The standards for obtaining copyright protection for a work are not high. A work need only contain a modicum of originality to be enti-

---

17. The underpinnings of the federal copyright and patent statutes are rooted in Article I of the United States Constitution, which provides that "Congress shall have Power . . . [t]o promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries." U.S. Const. art. I, § 8, cl. 8. The Supreme Court has observed that the objective of the federal intellectual property statutes is to strike a balance between private and public rights of access to information, and "to motivate the creative activity of authors . . . and to allow the public access to the products of their genius after the limited period of exclusive control has expired." *Sony Corp. v. Universal City Studios, Inc.*, 464 U.S. 417, 429 (1984); *see also Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 480 (1974) (proclaiming that the limited monopoly fostered by the federal intellectual property laws "will have a positive effect on society through the introduction of new products and processes of manufacture into the economy, and the emanations by way of increased employment and better lives for our citizens").

18. *See infra* Part I.C-D.

tioned to copyright protection.<sup>19</sup> Copyright in a work subsists until seventy years after the death of the author.<sup>20</sup> The owner of a copyright is granted the exclusive right to reproduce the work, prepare derivative works, distribute, display, and perform the copyrighted work.<sup>21</sup> A significant limitation of copyright, however, is that ownership of a valid copyright in a work does not foreclose the independent creation of that work by someone else.<sup>22</sup> Further, although copyright presupposes that a work will be disclosed to the public,<sup>23</sup> there is no explicit requirement, as there is under patent law,<sup>24</sup> that this disclosure occur as a condition to protection.<sup>25</sup>

Despite the broad protections offered by copyright law, there is a fundamental principle, known as the idea/expression dichotomy,<sup>26</sup> that can significantly limit the amount of protection afforded copyrighted works.<sup>27</sup> This principle dictates that protection does not extend to the ideas encapsulated in a work, but only to the original expression of those ideas.<sup>28</sup> The Supreme Court explained this distinction in the seminal case of *Baker v. Selden*.<sup>29</sup> In *Baker*, the complainant's testator held a copyright on a book that illustrated a bookkeeping system based on forms with ruled lines and headings.<sup>30</sup> He sued the defendant for copyright infringement after the defendant published a book modeled on the same system that simply contained a different arrangement of columns and headings.<sup>31</sup> The Supreme Court, noting that the accounting system was designed for use by the public,<sup>32</sup> held that just as the author of a mathematical science textbook cannot claim copyright in the methods of operation endorsed therein,<sup>33</sup> the complainant could not claim copyright in the method of double-entry accounting expressed in the book.<sup>34</sup> While the explana-

---

19. See *Feist Publications, Inc. v. Rural Tel. Serv. Co.*, 499 U.S. 340, 345 (1991).

20. See 17 U.S.C.A. § 302(a) (West Supp. 1999).

21. See 17 U.S.C. § 106 (1994).

22. See *Novelty Textile Mills, Inc. v. Joan Fabrics Corp.*, 558 F.2d 1090, 1092 n.2 (2d Cir. 1977).

23. See *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 480-81 (1974).

24. See 35 U.S.C. § 112 (1994).

25. See 17 U.S.C. § 301.

26. See *Harper & Row, Publishers, Inc. v. Nation Enters.*, 471 U.S. 539, 556 (1985).

27. See *Feist Publications, Inc. v. Rural Tel. Serv. Co.*, 499 U.S. 340, 350 (1991).

28. See *id.*

29. 101 U.S. 99 (1879).

30. See *id.* at 99-100.

31. See *id.*

32. See *id.* at 103.

33. See *id.*

34. See *id.* at 107.

tion of the system was clearly copyrightable, protection for the system itself could only be had, if at all, under "letters-patent."<sup>35</sup>

This long-standing distinction between an idea and its expression is codified in § 102(b) of the Copyright Act.<sup>36</sup> Section 102(b) provides that "[i]n no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work."<sup>37</sup> According to Congress, § 102(b) was not intended to enlarge or contract the scope of copyright protection previously afforded under the law;<sup>38</sup> it simply makes explicit the distinction between an idea and its expression.<sup>39</sup> Moreover, the legislative history makes clear that, although the ideas embodied in a work may be hidden or not readily perceptible by the human eye, whether exposed or not, the ideas nevertheless are not copyrightable.<sup>40</sup>

Several additional copyright doctrines function to separate ideas, which are not protected, from the expression of those ideas, which is protected. The doctrine of merger dictates that if there is only one or very few ways of expressing an idea, the expression is deemed to have merged with the idea.<sup>41</sup> If merger occurs, expression of the idea will not be protected; otherwise, copyright would improperly be conferred upon the idea itself.<sup>42</sup> In the computer context, merger occurs when specific instructions are the only and essential means of accomplishing a given task.<sup>43</sup> Additionally, under the judicially created doctrine of

---

35. *See id.* at 105. "Letters-patent" is a reference to the patent law. *See* Edward C. Walterscheid, *Patents and Manufacturing in the Early Republic*, 80 J. Pat. & Trademark Off. Soc'y 855, 857 (1998).

36. *See* 17 U.S.C. § 102(b) (1994).

37. *Id.*

38. *See* H.R. Rep. No. 94-1476, at 57 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5670.

39. *See id.*, *reprinted in* 1976 U.S.C.C.A.N. 5659, 5670.

40. Conversely, the original expression of an idea, even if not readily perceptible by the human eye, is protected. *See id.* at 52, *reprinted in* 1976 U.S.C.C.A.N. 5659, 5665. Similarly, even though a program may be in such a form that it may only be "perceived" by a machine, the program nonetheless may be entitled to copyright protection. *See, e.g.,* Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1248-51 (3d Cir. 1983) (rejecting contention that copyright is limited to works designed to be read by a human reader); Williams Elecs., Inc. v. Artic Int'l, Inc., 685 F.2d 870, 876-77 (3d Cir. 1982) (determining that object code represented on a Read Only Memory chip may be copyrighted).

41. *See* Kregos v. Associated Press, 937 F.2d 700, 705 (2d Cir. 1991).

42. *See* Bellsouth Adver. & Publ'g Corp. v. Donnelley Info. Publ'g, Inc., 999 F.2d 1436, 1442 (11th Cir. 1993).

43. *See* 5 Copyright, Congress & Technology: The Public Record 40 (Nicholas Henry ed., Oryx Press 1980) [hereinafter CONTU Final Report]; *see also* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 66 (D. Mass. 1990) (noting that the merger doctrine will deny copyright protection to certain aspects of computer programs). Regarding the application of merger to computer programs, one commentator noted that the "line [for determining when merger occurs] is a pragmatic one, drawn not on the basis of some metaphysical property of 'ideas,' but by balancing the

*scènes à faire*, certain features, symbols, or concepts that are indispensable to a particular discipline are not protected.<sup>44</sup> Courts that apply the *scènes à faire* doctrine to computer programs deny copyright protection to aspects of a program that are dictated by external factors, such as the mechanical specifications of a computer or widely accepted programming practices within the computer industry.<sup>45</sup> Finally, the expressive content of a work that is categorized as a "useful article" will not be protected if the expressive aspect of the work was dictated primarily by functional concerns.<sup>46</sup>

It is firmly established that computer programs are protected under the Copyright Act,<sup>47</sup> even though every computer program can be conceptualized, at least to the extent the program determines the steps to be taken in reaching a certain result, as a "procedure," "process," or "method of operation."<sup>48</sup> After the Commission on New Technological Uses ("CONTU") issued its findings,<sup>49</sup> the Act was amended in 1980 to explicitly extend copyright protection to computer programs.<sup>50</sup> Computer programs are classified under the statute as

---

need to protect the labors of authors with the desire to assure free access to ideas." 4 Melville B. Nimmer & David Nimmer, *Nimmer on Copyright* § 13.03[F][1], at 13-125 (1998).

44. *See* Atari, Inc. v. North Am. Philips Consumer Elecs. Corp., 672 F.2d 607, 616 (7th Cir. 1982).

45. *See* Softel, Inc. v. Dragon Med. & Scientific Communications, Inc., 118 F.3d 955, 963 (2d Cir. 1997).

46. Computer programs are purely functional in that they cause a computer to perform a particular task or set of tasks. *See* Dennis S. Karjala, *The Relative Roles of Patent and Copyright in the Protection of Computer Programs*, 17 J. Marshall J. Computer & Info. L. 41, 42 (1998) [hereinafter Karjala, *Relative Roles*]. Because separating the functional aspects from the nonfunctional aspects of program code would leave little to protect, however, the protection of computer programs under copyright is an exception to the useful article doctrine. *See* Dennis S. Karjala, *Copyright Protection of Computer Documents, Reverse Engineering, and Professor Miller*, 19 U. Dayton L. Rev. 975, 986 (1994) [hereinafter Karjala, *Copyright Protection*].

47. *See* Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1247-48 (3d Cir. 1983).

48. 17 U.S.C. § 102(b) (1994); Jack E. Brown, "Analytical Dissection" of Copyrighted Computer Software—Complicating the Simple and Confounding the Complex, 25 Ariz. St. L.J. 801, 831-32 (1993). If conceptualized in this fashion, computer programs would be denied protection under copyright law by the express terms of § 102(b). *See* Samuelson et al., *supra* note 16, at 2351.

49. *See* CONTU Final Report, *supra* note 43, at 21-27. CONTU was charged by the 93rd Congress with determining the extent of copyright protection for computer programs. *See* Pub. L. No. 93-573, tit. II, 88 Stat. 1873, 1873-74 (1974). CONTU, rejecting trade secret and patent protection as viable alternatives, found that copyright was the intellectual property protection best suited for computer programs. *See* Jack M. Haynes, *Computer Software: Intellectual Property Protection in the United States and Japan*, 13 J. Marshall J. Computer & Info. L. 245, 254 (1995).

50. *See* Computer Software Copyright Act of 1980, Pub. L. No. 96-517, 94 Stat. 3015, 3028 (codified at 17 U.S.C. §§ 101, 117 (1994)). Computer programs are defined as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101. "Statements or instructions" refers to program source code and object code. *See infra* note 53 and accompanying text. The Act was further amended to exempt from liability owners of computer pro-



"literary works," which by definition are not limited to the expression of words, but also include "numbers, or . . . numerical symbols or indicia, regardless of the nature of the material objects . . . in which they are embodied."<sup>51</sup> Moreover, courts, heeding the congressional mandate that computer programs are copyrightable, have devised complicated tests to separate the protectible aspects of computer programs from those that are non-protectible.<sup>52</sup> Thus, current copyright law protects all original expressive aspects of a computer program, including its source code and object code.<sup>53</sup>

### B. Patent Protection for Computer Programs

Patent law derives from the same constitutional clause as does copyright law,<sup>54</sup> and it possesses similar incentives and objectives: inventors are provided with a limited monopoly on their product, in the hope that this encouragement will stimulate and increase growth in the useful arts.<sup>55</sup> Patent law, however, sets higher subject matter and

---

grams who make or authorize the making of copies of, or adaptations to, computer programs when necessary to operate the program. See 94 Stat. at 3028.

51. 17 U.S.C. § 101.

52. See *Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 982 F.2d 693, 706-12 (2d Cir. 1992). Virtually all courts addressing software copyright cases after 1992 have followed *Altai*. See Mark A. Lemley, *Convergence in the Law of Software Copyright?*, 10 High Tech. L.J. 1, 14-15 (1995). But see *Lotus Dev. Corp. v. Borland Int'l, Inc.*, 49 F.3d 807, 814-15 (1st Cir. 1995) (refusing to adopt the abstraction-filtration-comparison test set forth in *Altai*), *aff'd per curiam*, 516 U.S. 233 (1996).

53. This Note undertakes to address only a program's literal aspects, its source and object code, even though there are additional nonliteral elements of a computer program that may be afforded protection, such as the program's graphical display. After a program's basic structure and design has been determined, it must be reflected in language the computer can comprehend. See *Altai*, 982 F.2d at 698. First, the programmer must write source code that reflects the program's structure in a programming language such as COBOL or FORTRAN. See *id.* For the programmer, this drafting process is comparable to a "novelist fleshing out the broad outline of his plot by crafting from words and sentences the paragraphs that convey the ideas." *Id.* (quoting Mark T. Kretschmer, Note, *Copyright Protection for Software Architecture: Just Say No!*, 1988 Colum. Bus. L. Rev. 823, 826). Source code is the textual form of a computer program and is comprised of two elements: the quasi-mathematical form of instructions that will instruct the computer, and instructive comments inserted by the computer programmer to help other readers understand the source code. See Andrew Johnson-Laird, *Software Reverse Engineering in the Real World*, 19 U. Dayton L. Rev. 843, 851 n.17 (1994). Because the source code cannot be executed by a computer, the second step is to translate or "compile" the source code into machine-readable object code. See *Altai*, 982 F.2d at 698. Object code cannot be understood by humans; it is a binary language consisting of ones and zeros which instructs the computer. See *id.* Typically, only the program in object code form is distributed to the public in an effort to preserve trade secret protection because humans cannot understand object code, therefore, the trade secrets embodied therein cannot be discovered. See *infra* Part I.C.

54. See *supra* note 17.

55. See David A. Rice, *Public Goods, Private Contract and Public Policy: Federal Preemption of Software License Prohibitions Against Reverse Engineering*, 53 U. Pitt. L. Rev. 543, 567-68 (1992).

eligibility standards than does copyright law.<sup>56</sup> Qualifying subject matter is limited to "any new and useful process, machine, manufacture, or composition of matter."<sup>57</sup> A patent may not be claimed on subject matter that was known or anticipated in the prior art, or was obvious to a person with ordinary skill in the subject area.<sup>58</sup> These stringent requirements ensure that patent protection is conferred only upon those inventions that truly advance the useful arts.<sup>59</sup>

Abstract ideas, laws of nature, and natural phenomena traditionally are ineligible for patent protection;<sup>60</sup> while copyright protects the expression of ideas,<sup>61</sup> a patent may only be claimed on the utilitarian embodiment of an idea.<sup>62</sup> In other words, copyright law generally eschews protection of functional works, while patent law protects only those works that are functional.<sup>63</sup> Further, while copyright law creates limited monopoly rights in original expression but does not prohibit others from using the ideas or knowledge expressed therein, the holder of a patent is granted the right to exclude others from making, using, or selling those ideas functionally embodied by the patent.<sup>64</sup> Finally, patent law, unlike copyright law, protects against subsequent independent inventors.<sup>65</sup>

There also are practical considerations involved in determining whether a particular work is best protected under a copyright or a patent. While copyright protection is not as exclusive as patent protection, copyright subsists for the life of the author plus seventy years, or 120 years if the author is a corporation.<sup>66</sup> Patent protection, on the other hand, lasts for twenty years from the date of application, after which time the public is allowed free access to the invention.<sup>67</sup> Moreover, the costs associated with obtaining a patent are far higher than those associated with a copyright. The patent process is time-consuming and complicated, and a patent application may cost more than

---

56. See A. Samuel Oddi, *An Uneasier Case for Copyright than for Patent Protection of Computer Programs*, 72 Neb. L. Rev. 351, 384 (1993).

57. 35 U.S.C. § 101 (1994).

58. See *id.* §§ 102, 103.

59. See Vincent Chiappetta, *Patentability of Computer Software Instruction as an "Article of Manufacture:" Software as Such as the Right Stuff*, 17 J. Marshall J. Computer & Info. L. 89, 100 (1998).

60. See *In re Alappat*, 33 F.3d 1526, 1542 (Fed. Cir. 1994).

61. See *supra* Part I.A.

62. See Allen B. Wagner, *Patenting Computer Science: Are Computer Instruction Writings Patentable?*, 17 J. Marshall J. Computer & Info. L. 5, 33 (1998).

63. See Karjala, *Relative Roles*, *supra* note 46, at 41-42.

64. One who, without authority, "makes, uses or sells any patented invention" infringes a patent. 35 U.S.C. § 271(a) (1994).

65. See David Bender & Anthony R. Barkume, *Patents for Software-Related Inventions*, 5 Software L.J. 279, 287 (1992).

66. See 17 U.S.C.A. § 302(a) (West Supp. 1999).

67. See 35 U.S.C. § 154(a)(2).

\$20,000.<sup>68</sup> In addition, there are substantial filing fees, patent issuance fees, and maintenance fees required to maintain the patent for its duration.<sup>69</sup> Copyright protection, on the other hand, may be secured for the life of the copyright by registration and the payment of a nominal filing fee.<sup>70</sup>

In recent years, patent protection has become increasingly available to computer programs.<sup>71</sup> As computer programs begin to receive both patent and copyright protection, the traditional distinction between patent and copyright law—that patent law protects functional innovations while copyright law eschews protection of functional works<sup>72</sup>—has eroded with reference to computer programs because Congress has carved out an exception for computer programs under copyright.<sup>73</sup> The extent of patent protection available for computer programs, however, has yet to be clearly delineated by the courts or Congress.<sup>74</sup> Two recent cases from the Federal Circuit both rejected the “printed matter” doctrine.<sup>75</sup> This judicially conceived doctrine was invoked by United States Patent and Trademark Office (“PTO”) to deny patent protection to inventions manifested in human-readable medium in the form of printed lines, characters, words, and digits<sup>76</sup>—as is computer program code. The court’s rejection of the printed matter doctrine is indicative of the broadening scope of patentability for computer programs.<sup>77</sup>

---

68. See John G. Mills, *Possible Defenses to Complaints for Copyright Infringement and Reverse Engineering of Computer Software: Implications for Antitrust and I.P. Law*, 80 J. Pat. & Trademark Off. Soc’y 101, 130 (1998).

69. See *id.*

70. See 37 C.F.R. § 202.3 (1998).

71. See Mark A. Lemley & David W. O’Brien, *Encouraging Software Reuse*, 49 Stan. L. Rev. 255, 259 (1997).

72. See Karjala, *Relative Roles*, *supra* note 46, at 41-42.

73. See *supra* note 46; see also Chiappetta, *supra* note 59, at 156 n.285 (noting the distinction between patent and copyright protection of computer software).

74. See Wagner, *supra* note 62, at 6. The Patent and Trademark Office (“PTO”) historically had viewed software components as unpatentable mathematical formulations existing independently of functional structures or processes. See Lemley & O’Brien, *supra* note 71, at 280-81. Thus, patent claims routinely were rejected by the courts and the PTO unless the claim included an accompanying physical structure or embodiment. See, e.g., *In re Pardo*, 684 F.2d 912, 915 (C.C.P.A. 1982) (describing the two-part test for patentability as involving a determination first whether a mathematical algorithm is recited, and secondly, whether the algorithm is applied to physical elements).

75. See *In re Beauregard*, 53 F.3d 1583, 1584 (Fed. Cir. 1995) (noting that the Commissioner of Patents and Trademarks had conceded that computer software on computer-readable media is patentable subject matter); *In re Lowry*, 32 F.3d 1579, 1583 (Fed. Cir. 1994) (holding that the application of the printed matter doctrine was inappropriate where the information in question was to be processed by a machine rather than by the human mind).

76. See Jeffrey R. Kuester et al., *A New Frontier in Patents: Patent Claims to Propagated Signals*, 17 J. Marshall J. Computer & Info. L. 75, 77 (1998).

77. See also *State St. Bank & Trust Co. v. Signature Fin. Group, Inc.*, 149 F.3d 1368, 1373-74 (Fed. Cir. 1998) (refusing to apply the mathematical algorithm excep-

The guidelines promulgated by the PTO regarding computer-related inventions ("Examination Guidelines") signal further acceptance of the principle that computer programs on computer disks are patentable.<sup>78</sup> According to the Examination Guidelines, these programs define "structural and functional interrelationships between the computer program and the medium which permit [their] functionality to be realized."<sup>79</sup> The Examination Guidelines distinguish between patentable and nonpatentable material by differentiating functional from nonfunctional descriptive material.<sup>80</sup> Descriptive material, such as computer code, is patentable only if it is structurally and functionally interrelated to other aspects of the invention.<sup>81</sup> Given that computer programs are intrinsically different from melodies,<sup>82</sup> while it "would exalt form over substance" to allow music to become patentable merely because it is encoded on computer-readable medium,<sup>83</sup> this is the precise result approved by the Examination Guidelines for computer programs.<sup>84</sup> Presently, the PTO has 300 patent examiners evaluating over 40,000 applications for patents on computer programs and computer program-related inventions.<sup>85</sup> At least 80,000 active software patents will exist by the turn of the century.<sup>86</sup>

---

tion to computer software where it produces "a useful, concrete and tangible result"), *cert. denied*, 119 S. Ct. 851 (1999).

78. See Examination Guidelines for Computer-Related Inventions, 61 Fed. Reg. 7478, 7482, 7487 n.1 (1996) [hereinafter Examination Guidelines]; see also *Patent and Trademark Office: Software Embodied on Diskette Is Patentable Subject Matter*, PTO Declares, 50 Pat. Trademark & Copyright J. (BNA) 3, 3 (1995) (observing that the PTO "declared that computer programs embodied in a tangible medium, such as a floppy diskette, are patentable subject matter").

79. Examination Guidelines, *supra* note 78, at 7482.

80. See *id.* at 7481.

81. See *id.*

82. See Julie E. Cohen, *Reverse Engineering and the Rise of Electronic Vigilantism: Intellectual Property Implication of "Lock-Out" Programs*, 68 S. Cal. L. Rev. 1091, 1112 (1995) [hereinafter Cohen, *Electronic Vigilantism*].

83. See Examination Guidelines, *supra* note 78, at 7481.

84. The Examination Guidelines state that "[w]hen functional descriptive material [such as computer code] is recorded on some computer-readable medium it becomes structurally and functionally interrelated to the medium and will be statutory in most cases." *Id.*

85. See Mills, *supra* note 68, at 129. A discussion of whether the recent decisions and PTO Guidelines impermissibly intrude into the sphere of copyright is beyond the scope of this Note. Compare *Mazer v. Stein*, 347 U.S. 201, 217 (1954) ("Neither the Copyright Statute nor any other says that because a thing is patentable it may not be copyrighted."), with *Wagner*, *supra* note 62, at 36 (arguing that patent protection for computer programs is inconsistent with copyright fundamentals).

86. See Robert C. Haldiman, *Ephemeral Infringement by Customizable Software: Liability Structures to Promote New Technology and Protect the Old*, 80 J. Pat. & Trademark Off. Soc'y 651, 674 n.13 (1998).

### C. Trade Secret Protection for Computer Programs

Common law trade secret doctrines may also protect proprietary rights in computer programs.<sup>87</sup> The Uniform Trade Secrets Act<sup>88</sup> defines a trade secret as follows:

[I]nformation, including a formula, pattern, compilation, program, device, method, technique, or process, that:

(i) derives independent economic value, actual or potential, from not being generally known to, and not being readily ascertainable by proper means by, other persons who can obtain economic value from its disclosure or use, and

(ii) is the subject of efforts that are reasonable under the circumstances to maintain its secrecy.<sup>89</sup>

The maintenance of absolute secrecy of a trade secret is not a prerequisite to the protection of a trade secret;<sup>90</sup> only reasonable safeguards that protect against the discovery of a trade secret are required.<sup>91</sup> Factors considered in determining whether the secrecy of a trade secret has been preserved may include: (1) the extent to which the informa-

---

87. See, e.g., *Softel, Inc. v. Dragon Med. & Scientific Communications, Inc.*, 118 F.3d 955, 968 (2d Cir. 1997) (finding that the lower court's conclusion that the individual software elements of plaintiff's computer program were not protectable did not address plaintiff's claim that the combination of the elements was a trade secret); *Rivendell Forest Prod., Ltd. v. Georgia-Pacific Corp.*, 28 F.3d 1042, 1045 (10th Cir. 1994) (recognizing that a "trade secret can exist in a combination of characteristics and components, each of which, by itself, is in the public domain, but the unified process, design and operation of which, in unique combination, affords a competitive advantage and is a protectable secret" (quoting *Imperial Chem. Indus., Ltd. v. Nat'l Distillers & Chem. Corp.*, 342 F.2d 737, 742 (2d Cir. 1965)); *Harbor Software, Inc. v. Applied Sys., Inc.*, 887 F. Supp. 86, 90 (S.D.N.Y. 1995) (noting that "the overall *design* of a software program may be protectable as a trade secret, even if the individual *components* of that program are common knowledge in the programming industry" (emphasis in original)). Although state trade secret law does not create true intellectual property rights, as do the copyright and patent laws, it does establish standards of conduct that protect against unethical or unfair methods of gaining access to industrial technology and knowledge. See *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 475-76 (1974).

88. Unif. Trade Secrets Act (amended 1985), 14 U.L.A. 433 (1990). Over forty states have adopted some version of the Act. See Arthur J. Schwab & David J. Porter, *Federal Protection of Trade Secrets: Understanding the Economic Espionage Act of 1996*, J. Proprietary Rights, Apr. 1998, at 2, 2.

89. Unif. Trade Secrets Act § 1, 14 U.L.A. at 438. States that have not adopted the Uniform Trade Secrets Act, including New York, generally look to the first Restatement of Torts when defining a trade secret. See *Softel*, 118 F.3d at 968. The Restatement defines a trade secret as "any formula, pattern, device or compilation of information which is used in one's business, and which gives him an opportunity to obtain an advantage over competitors who do not know or use it." Restatement of Torts § 757 cmt. b. (1939)

90. See *Rockwell Graphic Sys., Inc. v. DEV Indus., Inc.*, 925 F.2d 174, 177 (7th Cir. 1991); *Anacomp, Inc. v. Shell Knob Servs., Inc.*, No. 93 Civ. 4003, 1994 U.S. Dist. LEXIS 223, at \*27 (S.D.N.Y. Jan. 7, 1994), *aff'd*, 29 F.3d 621 (2d Cir. 1994).

91. See Restatement of Torts § 757 cmt. b. Trade secrets may theoretically be protected in perpetuity.

tion is known to persons both outside the business, e.g., customers, and inside the business, e.g., employees; (2) the precautions taken to preserve the secrecy of the trade secret; (3) the value of the information to the holder as against competitors; (4) the amount of money and effort expended in obtaining or developing the information; and (5) how valuable the information would be to competitors, or the time and expense it would take another to acquire and duplicate the information.<sup>92</sup> Thus, wide-scale distribution to the public of a certain product will not, by itself, preclude trade secret protection.<sup>93</sup> Accordingly, because object code can be "read" only by a machine, the distribution of a program in *object* code form preserves the distributor's right under state trade secret laws to claim misappropriation of the program's *source* code.<sup>94</sup>

Trade secret law, however, may be held preempted by § 301 of the Copyright Act if it proscribes conduct equivalent to that prohibited by copyright law.<sup>95</sup> If preemption occurs, redress under state trade secret laws will be foreclosed.<sup>96</sup> To escape preemption, state-created rights must be predicated on a right extending beyond the exclusive rights conferred by § 106 of the Copyright Act.<sup>97</sup> That right, known as an "extra element," must make the nature of the state claim qualitatively different from a federal copyright infringement claim.<sup>98</sup> For example, a trade secret claim based solely on the unauthorized copying of a

---

92. See *Gates Rubber Co. v. Bando Chem. Indus., Ltd.*, 9 F.3d 823, 848 (10th Cir. 1993).

93. For example, the recipe for Coca-Cola is one of the best-kept trade secrets in the world. See *Coca-Cola Bottling Co. v. Coca-Cola Co.*, 107 F.R.D. 288, 289 (D. Del. 1985).

94. See, e.g., *Trandes Corp. v. Guy F. Atkinson Co.*, 996 F.2d 655, 663-64 (4th Cir. 1993) (concluding that distribution of disks containing object code did not destroy the secrecy of the source code); *Data Gen. Corp. v. Grumman Sys. Support Corp.*, 825 F. Supp. 340, 359 (D. Mass. 1993) (holding that the contents of a computer program distributed only in object code format were protectable trade secrets), *aff'd*, 36 F.3d 1147 (1st Cir. 1994); *Barr-Mullin, Inc. v. Browning*, 424 S.E.2d 226, 230 (N.C. Ct. App. 1993) (finding that source code was not "readily ascertainable" from object code, and therefore could be considered a trade secret); cf. *Foreign Sale of MS-DOS Is Contributory Infringement*, 48 Pat. Trademark & Copyright J. (BNA) 165, 166 (1994) (reporting that the court in *Stac Elecs. v. Microsoft Corp.*, CV-93-413-ER (C.D. Cal. 5/13/94 and 6/8/94), held that trade secret protection had been lost by distribution of copies of software to customers who could have reverse engineered it and discovered the trade secrets).

95. See *Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 982 F.2d 693, 716 (2d Cir. 1992). Section 301 provides, in pertinent part, that "[n]othing in this title annuls or limits any rights or remedies under the common law or statutes of any State with respect to . . . activities violating legal or equitable rights that are not equivalent to any of the exclusive rights within the general scope of copyright as specified by section 106." 17 U.S.C. § 301(b) (1994). The patent laws do not contain an express preemption section and state trade secret laws may coexist with the federal patent laws. See *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 478-79 (1974).

96. See *Altai*, 982 F.2d at 716-17.

97. See *Trandes*, 996 F.2d at 659.

98. See *Altai*, 982 F.2d at 716.

copyrighted work would be preempted by § 301 because that right falls within the general scope of copyright law.<sup>99</sup> On the other hand, rights that are different in kind from the rights conferred by copyright include breaches of confidential relationships, agreements, fiduciary duties, and the improper appropriation of a trade secret.<sup>100</sup>

Copyright preemption of trade secret claims may be avoided through the use of contractual agreements, popularly known as "shrinkwrap licenses." A shrinkwrap license typically offers terms that the purchaser of the software accepts by opening the cellophane wrapper that encases the package containing the computer software.<sup>101</sup> The license terms most often prohibit the licensee from transferring, sublicensing, renting, leasing, conveying, copying, modifying, translating, converting to another programming language, or reverse engineering the software.<sup>102</sup> These contracts enable computer software manufacturers to protect material, including ideas, processes, and systems, that might otherwise be unprotected.<sup>103</sup> It is the contractual breach of promise,<sup>104</sup> rather than obligations that arise by operation of law, that supplies the "extra element" to qualitatively distinguish trade secret claims from copyright infringement claims based solely on unauthorized copying.<sup>105</sup> The license prohibitions

---

99. See *id.* at 717.

100. A trade secret is appropriated improperly by "theft, bribery, misrepresentation, breach or inducement of a breach of a duty to maintain secrecy, or espionage through electronic or other means." Unif. Trade Secrets Act § 1 (amended 1985), 14 U.L.A. 433, 437 (1990). The Restatement's definition is virtually identical. See Restatement (Third) of Unfair Competition § 43 (1995) ("'Improper' means of acquiring another's trade secret . . . include theft, fraud, unauthorized interception of communications, inducement of or knowing participation in a breach of confidence, and other means either wrongful in themselves or wrongful under the circumstances of the case.").

101. See 5 Nimmer & Nimmer, *supra* note 43, § 27.02[B], at 27-14. "Click here" licenses, also referred to as "clickwrap" licenses, are the electronic equivalent of shrinkwrap licenses. See Charles R. McManis, *The Privatization (or "Shrink-Wrap") of American Copyright Law*, 87 Cal. L. Rev. 173, 173 (1999) [hereinafter McManis, *Privatization*]. This type of license appears on a user's computer screen when the user first loads a computer program, and requires the user to "click" his acceptance of the terms of the license before he uses the software. See Madison, *supra* note 16, at 1058. Both types of licenses will be referred to herein as shrinkwrap licenses.

102. See Mark Lemley, *Intellectual Property and Shrinkwrap Licenses*, 68 S. Cal. L. Rev. 1239, 1246 (1995) [hereinafter Lemley, *Intellectual Property*]. For an example of a typical shrinkwrap license, see *Vault Corp. v. Quaid Software Ltd.*, 847 F.2d 255, 257 n.2 (5th Cir. 1988).

103. See Madison, *supra* note 16, at 1029 n.11. Both patent and copyright laws protect only those ideas that are either functionally or expressively embodied in a work. See *supra* Part I.A-B.

104. The contractual breach of promise may also be referred to as a breach of trust or confidential relationship. See Maureen A. O'Rourke, *Drawing the Boundary Between Copyright and Contract: Copyright Preemption of Software License Terms*, 45 Duke L.J. 479, 523 (1995) [hereinafter O'Rourke, *Drawing the Boundary*].

105. See *Frontline Test Equip., Inc. v. Greenleaf Software, Inc.*, 10 F. Supp. 2d 583, 592 (W.D. Va. 1998).

against reverse engineering or otherwise modifying a program also help to preserve the secrecy of computer software, which is necessary for trade secret protection.<sup>106</sup> Thus, state trade secret claims may proceed alongside federal copyright claims.<sup>107</sup>

#### D. *Contract Protection for Computer Programs*

Early in the history of computer software development, a question existed regarding whether software was protectable under copyright, patent, or trade secret law.<sup>108</sup> Congress did not amend the Copyright Act to explicitly include computer programs until 1980,<sup>109</sup> and patent protection for certain types of software was not accepted by the courts until 1994.<sup>110</sup> To combat this uncertainty, computer software manufacturers, perceiving that intellectual property protection was lacking, devised shrinkwrap licenses to obtain the necessary proprietary protection for their products.<sup>111</sup> Shrinkwrap licenses were routinely held unenforceable,<sup>112</sup> however, until the Seventh Circuit's recent holding in *ProCD, Inc. v. Zeidenberg*.<sup>113</sup>

In *ProCD*, the defendant, Matthew Zeidenberg, had purchased a copy of ProCD's Select Phone computer program and database containing over 95,000,000 phone listings compiled from approximately 3000 public telephone books.<sup>114</sup> The Select Phone program was sold subject to a shrinkwrap license that prohibited the purchaser from making "the Software or the Listings in whole or in part available to any other user in any networked or time-shared environment, or transfer[ing] the Listings in whole or in part to any computer other than the computer used to access the Listings."<sup>115</sup> In violation of this provision, Zeidenberg extracted the phone listings from the Select

---

106. See *supra* notes 90-91 and accompanying text.

107. See 1 Nimmer & Nimmer, *supra* note 43, § 1.01[B][1], at 1-10 to -17.

108. See Stephen Fraser, *Canada-United States Trade Issues: Back from Purgatory? Why Computer Software "Shrink-Wrap" Licenses Should Be Laid to Rest*, 6 Tul. J. Int'l & Comp. L. 183, 189 (1998).

109. See *supra* note 50 and accompanying text.

110. See *In re Alappat*, 33 F.3d 1526, 1542-45 (Fed. Cir. 1994) (holding that patentability is not precluded even though computer program contains mathematical subject matter).

111. See Thomas M.S. Hemnes, *Restraints on Alienation, Equitable Servitudes, and the Feudal Nature of Computer Software Licensing*, 71 Denv. U. L. Rev. 577, 578-80 (1994).

112. See, e.g., *Step-Saver Data Sys., Inc. v. Wyse Tech.*, 939 F.2d 91, 105-06 (3d Cir. 1991) (rejecting the defendant's argument that, under the U.C.C., shrinkwrap licenses were binding modifications); *Vault Corp. v. Quaid Software Ltd.*, 847 F.2d 255, 269-70 (5th Cir. 1988) (affirming lower court's holding that shrinkwrap license at issue was an unenforceable contract of adhesion); Lemley, *Intellectual Property*, *supra* note 102, at 1248-53 (observing that before *ProCD*, shrinkwrap licenses were not enforced).

113. 86 F.3d 1447 (7th Cir. 1996).

114. See *ProCD, Inc. v. Zeidenberg*, 908 F. Supp. 640, 644-45 (W.D. Wis.), *rev'd*, 86 F.3d 1447 (7th Cir. 1996).

115. *Id.* at 645.



Phone program, included them in his own database program, and made the package available on his company's website on the Internet.<sup>116</sup> ProCD then sued Zeidenberg and his company, Silken Mountain Web Services, for copyright infringement, breach of the license agreement, and unfair competition.<sup>117</sup>

The district court granted the defendants' motion for summary judgment, holding that Zeidenberg's use of ProCD's computer program fell within § 117 of the Copyright Act because Zeidenberg used ProCD's program only to access the data supplied by ProCD.<sup>118</sup> Although the Select Phone program was within the scope of the Copyright Act as a compilation of facts, according to the court, the telephone and address data contained therein were not copyrightable because they were not sufficiently original.<sup>119</sup> The court also held that the shrinkwrap license was not breached because the defendants had not assented to its terms,<sup>120</sup> and that even if the license were enforceable, the agreement was "preempted by federal copyright law to the extent plaintiff intended it to apply to uncopyrightable data."<sup>121</sup>

The Seventh Circuit, in an opinion by Judge Easterbrook, reversed the district court.<sup>122</sup> After finding shrinkwrap licenses generally enforceable,<sup>123</sup> the court held that § 301 of the Copyright Act did not preempt their enforcement.<sup>124</sup> In so holding, the court drew a distinction between the rights conferred by copyright, which are enforceable rights against the entire world regardless of the existence of an agreement, and rights created by contract, the enforceability of which depend on proof of a contractual agreement.<sup>125</sup> According to the Seventh Circuit, § 301 does not interfere with private transactions in

---

116. *See id.*

117. *See id.* at 644.

118. *See id.* at 649-50. Section 117 provides, in pertinent part:

[I]t is not an infringement for the owner of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

(1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner . . . .

17 U.S.C. § 117 (1994).

119. *See ProCD*, 908 F. Supp. at 656-57. The Supreme Court, in *Feist Publications, Inc. v. Rural Telephone Service Co.*, 499 U.S. 340 (1991), reiterated that copyright will only be afforded those works that satisfy the constitutionally mandated standard of originality. *See id.* at 347-48. The Court further stated that it is a well-established proposition that facts, such as the phone and address listings in *ProCD*, and ideas are not copyrightable. *See id.* at 344.

120. *See ProCD*, 908 F. Supp. at 644.

121. *Id.*

122. *See ProCD, Inc. v. Zeidenberg*, 86 F.3d 1447, 1449 (7th Cir. 1996).

123. *See id.* at 1452-53.

124. *See id.* at 1454-55.

125. *See id.*

intellectual property;<sup>126</sup> thus, private parties are free to set up between themselves controlling rights and restrictions that are not equivalent to any of the exclusive rights within the scope of copyright.<sup>127</sup>

Courts and commentators have generally accepted *ProCD* as confirmation that shrinkwrap licenses are now enforceable.<sup>128</sup> Moreover, the drafters of the proposed Uniform Computer Information Transactions Act also have adopted the position that shrinkwrap licenses are valid.<sup>129</sup> Under the proposed draft, the licenses are enforceable as long as purchasers have the opportunity to review and manifest assent to the terms of the license, along with the ability to reject the terms and return the software for a refund.<sup>130</sup>

The enforceability of shrinkwrap licenses has far-reaching implications for the intellectual property regime generally, and the use and development of computer programs in particular, because the licenses supplement the federal intellectual property regime with the added protection of trade secret and contract. Indeed, commentators have argued that shrinkwrap licenses threaten to engulf pre-existing intellectual property schemes, bringing about the privatization of intellectual property protection for digital works.<sup>131</sup> Whether this is a good idea is debatable;<sup>132</sup> if left unchecked, however, the private sphere will

---

126. The Seventh Circuit stated that a "copyright is a right against the world. Contracts, by contrast, generally affect only their parties; strangers may do as they please, so contracts do not create 'exclusive rights.'" See *id.* at 1454.

127. See *id.* at 1455.

128. See, e.g., *Expeditors Int'l of Wash., Inc. v. Direct Line Cargo Management Servs., Inc.*, 995 F. Supp. 468, 483 (D.N.J. 1998) (noting that a contract claim is not equivalent to rights under copyright law because a promise is required); *Lattie v. Murdach*, 42 U.S.P.Q.2d (BNA) 1240, 1244 (N.D. Cal. 1997) (concluding that a contract claim is not equivalent to exclusive copyright rights); *Architectronics, Inc. v. Control Sys., Inc.*, 935 F. Supp. 425, 438-41 (S.D.N.Y. 1996) (recognizing that the consensus among courts and commentators is that breach of contract claims are not pre-empted by the Copyright Act); Madison, *supra* note 16, at 1053 & n.104 (citing commentators).

129. See National Conference of Comm'rs on Unif. State Laws, *Uniform Commercial Code Article 2B: Software Contracts and Licenses of Information* § 2B-208 ("Mass-Market Licenses") (visited Apr. 11, 1999) <<http://www.law.uh.edu/ucc2b/080198/080198.html>> (Aug. 1, 1998 draft) [hereinafter *Uniform Computer Act*]. The former Article 2B for Licenses, when finalized, will no longer supplement the UCC with provision for electronic contracts and software licensing agreements, but instead will be promulgated as the Uniform Computer Information Transactions Act. See National Conference of Comm'rs on Unif. State Laws & Am. Law Inst., *NCCUSL to Promulgate Freestanding Uniform Computer Information Transactions Act: ALI and NCCUSL Announce That Legal Rules for Computer Information Will Not Be Part of UCC* (visited Apr. 16, 1999) <<http://www.ali.org/ali/pr040799.htm>>.

130. See *Uniform Computer Act*, *supra* note 129, § 2B-208.

131. See Madison, *supra* note 16, at 1029-31; McManis, *Privatization*, *supra* note 101, at 173.

132. See Tom W. Bell, *Fair Use v. Fared Use: The Impact of Automated Rights Management on Copyright's Fair Use Doctrine*, 76 N.C. L. Rev. 557, 616-17 (1998) (arguing that a system of private contract and "fared use" may offer greater access to proprietary works).

likely dictate the future constitution of computer software in the public sphere entirely.<sup>133</sup>

With the additional protection of contract, exclusive proprietary rights that extend beyond those conferred by the federal statutes may now be established between individuals.<sup>134</sup> Additionally, contractual provisions help to preserve trade secret claims that might otherwise be preempted by the Copyright Act.<sup>135</sup> After *ProCD*, proprietary protection for computer programs may now be expressed in terms of the line between federal copyright and patent rights, which are good against the world and favor disclosure,<sup>136</sup> and state trade secret and contract rights enforceable against particular individuals that inhibit access.

Determining the extent to which proprietary rights to computer software should extend beyond that which federal intellectual property laws bestow can best be accomplished by examining the debate surrounding the use of reverse engineering. Computer programs are distributed in object code format in an effort to preserve trade secret protection. The ideas embedded in the object code cannot be perceived by humans unless the object code is converted back to human-readable source code. This may be accomplished through the process of reverse engineering, which is generally permitted under the federal statutes. Thus, as discussed below in part II, an apparent conflict arises when contractual provisions prohibiting reverse engineering are enforced.

## II. FRAMING THE DEBATE AGAINST A REVERSE ENGINEERING BACKDROP

This part defines the process of reverse engineering and examines different purposes for which reverse engineering is undertaken. Additionally, this part explores the status of reverse engineering under federal and state laws.

### A. Reverse Engineering Generally

Intellectual property law attempts to strike an appropriate balance between the individual's private property interest and the general public's interest in having access to knowledge and innovation.<sup>137</sup> The tension between the competing objectives is clear, and the imperfect balance is in constant need of re-calibration. With regard to computer

---

133. See Madison, *supra* note 16, at 1031-32.

134. See Mark A. Lemley, *Beyond Preemption: The Law and Policy of Intellectual Property Licensing*, 87 Cal. L. Rev. 111, 148 (1999) [hereinafter Lemley, *Beyond Preemption*].

135. See *supra* notes 97-107 and accompanying text.

136. See Lemley & O'Brien, *supra* note 71, at 276.

137. See J.H. Reichman, *Legal Hybrids Between the Patent and Copyright Paradigms*, 94 Colum. L. Rev. 2432, 2436-42 (1994).

programs, perhaps the most rapidly developing and evolving intellectual property today, the raging debate among commentators asks how this balance is best achieved: to what extent should computer programs be in the public domain?

The debate is best illustrated in the context of determining the legality of reverse engineering computer programs. Reverse engineering has been described simply as a "fair and honest means . . . [of] starting with the known product and working backward to divine the process which aided in its development or manufacture."<sup>138</sup> In the mechanical world, reverse engineering is simply the process of taking something apart to determine what makes it tick.<sup>139</sup> When applied to computer programs, however, reverse engineering becomes more complicated.<sup>140</sup> Generally, reverse engineering is the process of deriving a program's human-readable source code from its object code.<sup>141</sup>

Whether to permit reverse engineering is a question that entails the balancing of competing public and private objectives. First, one might reverse engineer a program solely to study and learn from the program, or to use the source code to teach computer programming.<sup>142</sup> Second, reverse engineering makes programs more effective and reliable by allowing the debugging, customization, and modification of a program's source code.<sup>143</sup> Third, one may desire to reverse engineer a program to develop a compatible or interoperable product.<sup>144</sup> Finally, reverse engineering may be undertaken in an effort to produce a functionally equivalent market substitute for the original program.<sup>145</sup> The first three purposes arguably are desirable and in the public interest, while the fourth poses the greatest danger to the interests of software producers due to the relative ease with which a computer program

---

138. *Kewanee Oil Co. v. Bicron Corp.*, 416 U.S. 470, 476 (1974).

139. See Johnson-Laird, *supra* note 53, at 843.

140. For a detailed technical discussion regarding the complexities of reverse engineering, see *id.*

141. Reverse engineering is necessary for users and developers who desire to examine the structure and technical parameters of a computer program because object code is not intelligible to humans. See Charles R. McManis, *Intellectual Property Protection and Reverse Engineering of Computer Programs in the United States and the European Community*, 8 High Tech. L.J. 25, 29 (1993) [hereinafter McManis, *Intellectual Property*].

142. See G. Gervaise Davis III, *Scope of Protection of Computer-Based Works: Reverse Engineering, Clean Rooms and Decompilation*, in 15th Annual Computer Law Institute 115, 142-45 (PLI Pats., Copyrights, Trademarks, & Literary Prop. Course Handbook Series No. 370, 1993). Reverse engineering for educational purposes likely is a "fair use" under 17 U.S.C. § 107. See *id.* at 142-43. Indeed, program source code is a valuable tool used to teach computer programming. See Clayton Walnum, *Windows 98 Programming Secrets* (1998) (enclosing on compact disk the source code for all programs in tutorial book).

143. See McManis, *Intellectual Property*, *supra* note 141, at 29-30.

144. See *id.* at 30-31.

145. See *id.* at 31.

may be appropriated by second-comers.<sup>146</sup> Framed more narrowly, the question is whether the goals in the public interest may be achieved in a manner that retains sufficient access restrictions to prevent the outright piracy of computer programs.<sup>147</sup>

Whether the interests of the public sphere are more appropriately delineated by statute or left to a private ordering is the underlying dilemma. Although reverse engineering is not explicitly permitted under the federal copyright and patent statutes, the act of reverse engineering alone is not likely to constitute infringement.<sup>148</sup> Moreover, reverse engineering has long been considered a legal means of discovering a trade secret.<sup>149</sup> Thus, the process of reverse engineering can be viewed as a tool employed by the public to permissibly calibrate the balance between public and private rights. After *ProCD*, however, right holders are able to prohibit the public from engaging in reverse engineering by private contract.<sup>150</sup> Where this occurs, the private sphere arguably has tipped the scales too heavily in its favor.<sup>151</sup>

## B. *The Extent to Which Reverse Engineering Is Permitted Under Federal and State Laws*

### 1. Reverse Engineering and Copyright Law

Courts have carved out a limited "fair use"<sup>152</sup> exception to an action for copyright infringement based on copying necessarily involved in

---

146. Computer program code is especially vulnerable to fast, precise, and inexpensive copying. See Dennis S. Karjala, *Copyright Protection of Computer Program Structure*, 64 Brook. L. Rev. 519, 532-33 (1998) [hereinafter Karjala, *Program Structure*]. Therefore, allowing unauthorized users to appropriate the benefits of a producer's investment implicates the classic "public goods" problem. See Robert Cooter & Thomas Ulen, *Law and Economics* 108-12 (1988). This occurs when producers, worried that they will not be able to recoup their investment in a good before its value is appropriated by free riders, undersupply the market for the good. See *id.*; see also Samuelson et al., *supra* note 16, at 2337-41 (discussing the public goods problem and market failure associated with computer software).

147. See Madison, *supra* note 16, at 1097 (questioning how to conceive of the public domain in a fashion that allows private restrictions on access to a socially acceptable degree).

148. See *infra* Part II.B.1-2.

149. See *infra* Part II.B.3.

150. See *infra* Part II.B.4.

151. See David A. Rice, *Digital Information as Property & Product: U.C.C. Article 2B*, 22 U. Dayton L. Rev. 621, 636-38 (1997) (arguing that object code must be reverse engineered in order to free up the unprotected ideas embedded therein).

152. For the statutory foundation of the fair use doctrine, see 17 U.S.C. § 107 (1994). The statute requires a consideration of four factors:

- (1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes;
- (2) the nature of the copyrighted work;
- (3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and
- (4) the effect of the use upon the potential market for or value of the copyrighted work.

the process of reverse engineering because the ideas embedded in computer programs may not otherwise be ascertainable.<sup>153</sup> While older decisions refused to recognize a right to reverse engineer,<sup>154</sup> modern case law clearly authorizes it under certain circumstances.<sup>155</sup> For example, in *Sega Enterprises Ltd. v. Accolade, Inc.*,<sup>156</sup> Accolade desired to make game programs for Sega's Genesis home entertainment system, but the Genesis system was equipped with a "lock-out" program that did not allow games to be played that did not contain Sega's initialization code.<sup>157</sup> Rather than obtain a license from Sega for the code, Accolade reverse engineered Sega's program, figured out the initialization code, and copied it into Accolade's own game programs, thereby making them compatible with the Genesis system.<sup>158</sup> Sega sued, alleging, among other things, copyright infringement.<sup>159</sup> The trial court held that the copies of Sega's initialization code necessarily made by Accolade during the process of reverse engineering likely constituted copyright infringement, and that Accolade's copying presumptively was not a fair use because it was primarily for commercial purposes.<sup>160</sup> The Ninth Circuit reversed, holding that reverse engineering will be considered a fair use if there is a legitimate reason for reverse engineering, and it is the sole method of gaining access to the ideas and functional elements embodied in a computer program.<sup>161</sup> This reasoning applies even if the transaction was, in fact, commercial in nature.<sup>162</sup>

Courts that have considered the same issue have reached similar holdings.<sup>163</sup> Reverse engineering a product to develop a competing product and to create an interoperable product are therefore reasons regarded by the courts as legitimate under certain circumstances. No case, however, holds that there is a universal right to reverse engineer

---

*Id.*

153. See Mark A. Lemley & David McGowan, *Legal Implications of Network Economic Effects*, 86 Calif. L. Rev. 479, 524-25 (1998) [hereinafter Lemley & McGowan, *Legal Implications*].

154. See *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240, 1253 (3d Cir. 1983); *Digital Communications Assocs., Inc. v. Softklone Distrib. Corp.*, 659 F. Supp. 449, 460 (N.D. Ga. 1987).

155. See *Bateman v. Mnemonics, Inc.*, 79 F.3d 1532, 1539 n.18 (11th Cir. 1996) (holding that reverse engineering a computer program to reveal the unprotectable ideas embedded in the object code is a fair use); *Atari Games Corp. v. Nintendo of Am. Inc.*, 975 F.2d 832, 843 (Fed. Cir. 1992) (same); *DSC Communications Corp. v. Pulse Communications Inc.*, 976 F. Supp. 359, 364 (E.D. Va. 1997) (same).

156. 977 F.2d 1510 (9th Cir. 1992).

157. See *id.* at 1514-15.

158. See *id.* at 1514-16.

159. See *id.* at 1516.

160. See *Sega Enters. Ltd. v. Accolade, Inc.*, 785 F. Supp. 1392, 1396-99 (N.D. Cal.), *aff'd in part, rev'd in part*, 977 F.2d 1510 (9th Cir. 1992).

161. See *Sega*, 977 F.2d at 1527.

162. See *id.* at 1522-23.

163. See *supra* note 155 and accompanying text.

for any purpose or in all circumstances.<sup>164</sup> Therefore, while the public generally may reverse engineer materials in the private domain, courts, relying on the doctrine of fair use, are able to preserve the balance between public and private rights to copyrighted works.

## 2. Reverse Engineering and Patent Law

With regard to patented computer programs, if one makes, uses, or sells products that infringe the patent, reverse engineering will not be a defense to an action for patent infringement, even if done for commercial research purposes.<sup>165</sup> Patent protection, however, is premised not upon the secrecy of an invention, but rather upon the ability to preclude others from making, using, or selling any physical embodiment of the ideas functionally embodied in the invention.<sup>166</sup> Therefore, the unavailability of reverse engineering is theoretically mitigated because patent law explicitly mandates that protection be granted only upon disclosure of the invention to the public.<sup>167</sup> The invention must be disclosed in sufficient detail to enable one skilled in the art to make and use the invention.<sup>168</sup> Thus, even after patent protection is awarded for a new, useful, and nonobvious invention, the disclosure requirements allow others to examine and study the underlying ideas, theories, and principles with an eye toward "designing around" the patented program and building upon the ideas embodied therein.<sup>169</sup>

The reverse engineering of a patented program, however, may be necessary for a number of reasons. Although those who do not disclose a program's code risk losing patent protection for that aspect of their program,<sup>170</sup> patent disclosure requirements do not mandate that the source code of the patented program be disclosed.<sup>171</sup> Additionally, reverse engineering may be needed to gain access to unpatented components of the program or to discover information not disclosed in sufficient detail.<sup>172</sup> Therefore, if undertaken for the purposes of compatibility, competition, or study, as long as the reverse engineer does not use the discovered knowledge to make, use, or sell the patented product, it is likely that the mere act of reverse engineering does

---

164. See Lemley & McGowan, *Legal Implications*, *supra* note 153, at 525.

165. See *id.* at 524.

166. See 35 U.S.C. § 271(a) (1994).

167. See *id.* § 112.

168. See *id.*

169. See *Slimfold Mfg. Co. v. Kinkead Indus., Inc.*, 932 F.2d 1453, 1457 (Fed. Cir. 1991).

170. See Lemley & O'Brien, *supra* note 71, at 295 n.259.

171. See *Robotic Vision Sys., Inc. v. View Eng'g, Inc.*, 112 F.3d 1163, 1166 (Fed. Cir. 1997); Lawrence D. Graham & Richard O. Zerbe, Jr., *Economically Efficient Treatment of Computer Software: Reverse Engineering, Protection, and Disclosure*, 22 Rutgers Computer & Tech. L.J. 61, 96-97 (1996).

172. See Lemley & McGowan, *Legal Implications*, *supra* note 153, at 524 n.195.

not constitute patent infringement.<sup>173</sup> Thus, the balance between public and private rights is also preserved under the patent laws. While the public may reverse engineer a patented program to learn about it and build upon the ideas embedded therein, if the reverse engineer uses the knowledge acquired to make, use, or sell a replica of the patented program, the statutory rights of the private owner will be infringed.<sup>174</sup>

### 3. Reverse Engineering and Trade Secret Law

Trade secret protection, standing alone, cannot protect against the reverse engineering of a product by competitors.<sup>175</sup> Thus, to preserve trade secret protection for computer source code, it was necessary for manufacturers to take security measures beyond simply distributing the program in object code form, such as password access and encryption.<sup>176</sup> Although these measures made comprehension of the ideas underlying the program difficult, the general public's right to attempt to understand the program was preserved by the absence of any prohibition against reverse engineering. Accordingly, the unrestricted public right to reverse engineer served as a check on the undue constriction of the public sphere by private owners, and helped to maintain the balance between the public and private spheres. Now, however, because an indispensable provision in any shrinkwrap license is a prohibition on reverse engineering, if shrinkwrap licenses are enforceable, the delicate balance has been shattered. This is due to the fact that the public's right to access the ideas embedded in computer programs is wholly circumscribed by private legislation.<sup>177</sup>

### 4. Reverse Engineering and Private Contract

Shrinkwrap licenses, as the means by which private prohibitions against reverse engineering are enforced, allow private owners to overrule case law providing that reverse engineering is a permissible means of promoting competition and compatibility.<sup>178</sup> Under federal

---

173. See Graham & Zerbe, *supra* note 171, at 99.

174. See 35 U.S.C. § 271(a) (1994).

175. See, e.g., *Cataphote Corp. v. Hudson*, 422 F.2d 1290, 1293 (5th Cir. 1970) (holding that it is not a misappropriation to discover a trade secret by reverse engineering); Restatement (Third) of Unfair Competition § 43 (1995) ("Independent discovery and analysis of publicly available products or information are not improper means of acquisition.").

176. See Maureen A. O'Rourke, *Fencing Cyberspace: Drawing Borders in a Virtual World*, 82 Minn. L. Rev. 609, 612-13 (1998) [hereinafter O'Rourke, *Fencing Cyberspace*].

177. See Madison, *supra* note 16, at 1130. If reverse engineering is forbidden by contract, software developers may be granted de facto monopolies over ideas, processes, and systems that have not met the standards for copyright or patent protection. See *DSC Communications Corp. v. Pulse Communications Inc.*, 976 F. Supp. 359, 364 (E.D. Va. 1997); Samuelson et al., *supra* note 16, at 2390 n.329.

178. See *supra* Part II.B.1.



statutes, reverse engineering alone will not constitute infringement in most cases, and, under state trade secret law, if a trade secret is appropriated by reverse engineering, an action for misappropriation will not lie.<sup>179</sup> Where reverse engineering is prohibited by contract, however, there is no similar exception to an action for breach of contract brought against one who reverse engineers while contractually bound. Accordingly, private contract is the overriding force that allows private right owners to maintain control over the balance between public and private rights.<sup>180</sup> Private owners may employ contracts as a means of enforcing their own view as to what should be the appropriate balance between public and private rights. Without a means, such as the right to reverse engineer, to influence this private determination, the public sphere inevitably shrinks.

The best method of restoring the balance between public and private rights to computer software—that is, if one first accepts the premise that the balance is askew—is open to question. The judiciary may be the entity to restore this balance most effectively, as incremental decisions may, over time, produce a stable set of rules that reflect a desirable construction of the public sphere.<sup>181</sup> Another approach would be to leave it to the private market to correct the balance. Finally, congressional action may be necessary to implement broad policy determinations regarding the appropriate constitution of the public sphere. Relying solely on the judiciary or the private market to restore the balance, however, may not protect the public sphere from undue constriction at the hands of private owners. The next part addresses this concern.

### III. RESTORING THE BALANCE

This part considers the various approaches that have been adopted by courts addressing private constraints that unduly hinder the objectives of the federal intellectual property laws. This part also evaluates the argument that, if left unchecked, private market forces will ultimately determine the appropriate balance between public and private rights.

#### A. *Judicial Intervention*

The judiciary possesses the ability to enforce federal intellectual property rights over state-created rights in at least three ways: preemption, “first sale,” and misuse. This section explores each of these approaches.

---

179. See *supra* Part II.B.1–3.

180. See McManis, *Privatization*, *supra* note 101, at 176.

181. See Madison, *supra* note 16, at 1137–38.

## 1. The Three Approaches

### a. Preemption

The first approach to addressing private prohibitions against reverse engineering is preemption theory. This argument posits that state law rights comparable to or extending beyond those conferred by federal laws should be preempted if those rights confer proprietary rights in unprotectable aspects or components of works otherwise entitled to protection.<sup>182</sup> In other words, Congress, through the federal patent and copyright statutes, has made a judgment as to the appropriate level of protection to be granted those works eligible for federal protection, and the states have no authority to interpose their own schemes or alter Congress's judgment.<sup>183</sup> This approach would sanction reverse engineering as a permissible tool, one used to free up unprotected ideas concealed in a program's object code, allow others to build upon them, promote study and compatibility, and encourage competition. Moreover, preemption would prevent states from granting to their citizens enforceable rights, which would vary from state to state, extending beyond those conferred by federal statutes.

The preemption of private prohibitions against reverse engineering presumably would be achieved either vis-à-vis § 301 of the Copyright Act or, more likely, the Supremacy Clause of the U.S. Constitution. As previously noted, the general consensus is that private contractual rights are not equivalent to any of the exclusive rights within the province of copyright due to the existence of a contractual obligation.<sup>184</sup> A recent exception is *Symantec Corp. v. McAfee Associates, Inc.*<sup>185</sup> In that case, the court held that a software license prohibition against reverse engineering was preempted by § 301.<sup>186</sup> The *Symantec* court provided no legal reasoning or justification for its holding; whether subsequent courts will adopt its holding remains to be seen.

The Supremacy Clause,<sup>187</sup> the constitutional foundation of § 301,<sup>188</sup> may provide a more viable alternative to federal preemption of state-created rights.<sup>189</sup> The Supremacy Clause preempts state law in areas

---

182. See Julie E. Cohen, *Copyright and the Jurisprudence of Self-Help*, 13 Berkeley Tech. L.J. 1089, 1130-31 (1998) [hereinafter Cohen, *Self-Help*].

183. See Maureen A. O'Rourke, *Copyright Preemption After the ProCD Case: A Market-Based Approach*, 12 Berkeley Tech. L.J. 53, 80 (1997) [hereinafter O'Rourke, *Copyright Preemption*].

184. See *supra* notes 128-30 and accompanying text.

185. No. C-97-20367-JF, 1998 WL 740798 (N.D. Cal. June 9, 1998).

186. See *id.* at \*4-\*5.

187. The Supremacy Clause provides, in pertinent part, that the "Constitution, and the Laws of the United States which shall be made in Pursuance thereof . . . shall be the supreme Law of the Land." U.S. Const. art. VI, § 2.

188. See H.R. Rep. No. 94-1476, at 131 (1976), reprinted in 1976 U.S.C.C.A.N. 5659, 5747.

189. Indeed, Congress, when enacting § 301, did not consider the possibility of intervening state contractual rights that would permit the displacement of copyright. See Cohen, *Self-Help*, *supra* note 182, at 1130. Additionally, the *ProCD* court may

where Congress has mandated by express decree,<sup>190</sup> where congressional regulation in a certain area gives rise to the inference that Congress left no room for supplementary state regulation,<sup>191</sup> and where a state law conflicts with federal law to such a degree that the state law hinders the purposes and objectives of Congress.<sup>192</sup> Congress has not spoken directly on the issue of federal preemption of state rights in the computer program context, and generally has permitted states to supplement federal intellectual property protection with their own schemes to some degree.<sup>193</sup> The third method of Supremacy Clause preemption, however, may provide courts with the ability to deal flexibly with preemption issues. Courts faced with determining whether a state right hinders the purpose and objectives of Congress in passing the patent and copyright statutes are provided with two differing viewpoints regarding congressional purpose. One viewpoint is that Congress wanted to further the public good by allowing the public access to innovative works and inventions.<sup>194</sup> A second, equally valid, viewpoint is that Congress intended to reward authors and inventors by providing them with a limited monopoly over their works and inhibiting third-party appropriation.<sup>195</sup>

Preemption theory, however, may not be adequate to establish an appropriate balance between public and private rights. The *ProCD* court implied that prohibitions against reverse engineering would not be preempted because they "serve the same procompetitive functions as does the law of trade secrets."<sup>196</sup> Moreover, preemption theory alone may not be sufficient to preserve the objective of the intellectual property laws.<sup>197</sup> It is undesirable to preempt the entire field of licensing contracts, and § 301 preemption may not provide courts with the ability to preempt individual contractual terms while leaving the remainder of the contract intact.<sup>198</sup> Therefore, as one commentator has suggested, the application of field preemption to contracts is akin

---

have failed to consider preemption under the Supremacy Clause. See D.C. Toedt III, *COUNTERPOINT: Shrinkwrap License Enforceability Issues*, Computer Law., Sept. 1996, at 7, 8-9.

190. See *California Fed. Sav. & Loan Ass'n v. Guerra*, 479 U.S. 272, 280-81 (1987).

191. See *id.*

192. See *id.* at 281.

193. See Cohen, *Self-Help*, *supra* note 182, at 1130-31.

194. See *supra* note 17.

195. See *supra* note 17.

196. *ProCD, Inc. v. Zeidenberg*, 86 F.3d 1447, 1455 (7th Cir. 1996) (citation omitted). But see McManis, *Privatization*, *supra* note 101, at 182-84 (contending that such provisions should be preempted). The court presumably would analogize prohibitions against reverse engineering to encryption devices, which similarly hinder comprehension of a program's source code.

197. See Lemley, *Beyond Preemption*, *supra* note 134, at 144-51.

198. See *id.* at 145.

to "swinging a sledgehammer at a gnat."<sup>199</sup> Moreover, Supremacy Clause preemption poses difficulties for courts because it is often unclear whether a contract term directly conflicts with a federal statute.<sup>200</sup> This uncertainty may reduce the effectiveness and reliability of preemption theory.<sup>201</sup>

b. "First Sale" Doctrine

A second approach to addressing state-created rights, which was recently adopted by one post-*ProCD* court, involves a broader judicial policy determination than does preemption. In *Novell, Inc. v. Network Trade Center, Inc.*,<sup>202</sup> the defendant, a computer program distributor, was in the practice of purchasing computer programs from the plaintiff, applying for the plaintiff's "upgrade" for those programs, and then selling the upgrade for a profit to the end-users of the program.<sup>203</sup> The plaintiff subsequently announced that distributors would no longer be permitted to sell its upgrades, and that end-users who wanted the upgrades would have to purchase them directly from the plaintiff.<sup>204</sup> The defendant persisted in selling the plaintiff's upgrades to end-users, and the plaintiff sued.<sup>205</sup> One of the plaintiff's claims was that the defendant, by selling unauthorized upgrades to end-users, caused end-users to infringe the plaintiff's copyright when they loaded the upgrades onto their computers without valid licenses from the plaintiff.<sup>206</sup> The defendants contended that the end-users were owners of the computer programs, not licensees, who were entitled under the "first sale" doctrine to use the program without regard to the rights of the plaintiff as copyright holder.<sup>207</sup> The court agreed with the defendants, holding that sales of computer software are governed by the UCC, the purchaser is an "owner" by way of the sale, and that the "first sale" doctrine was applicable.<sup>208</sup> Therefore, the shrinkwrap licenses included with plaintiff's software were invalid insofar as they purported to maintain title in the software in the plaintiff.<sup>209</sup> Thus, according to the court, because computer programs are goods, shrink-

---

199. *Id.*; see also Apik Minassian, Comment, *The Death of Copyright: Enforceability of Shrinkwrap Licensing Agreements*, 45 UCLA L. Rev. 569, 601-07 (1997) (urging a more nuanced approach to preemption under § 301).

200. See Lemley, *Beyond Preemption*, *supra* note 134, at 146.

201. See *id.*

202. 25 F. Supp. 2d 1218 (D. Utah 1997).

203. See *id.* at 1222.

204. See *id.*

205. See *id.*

206. See *id.* at 1229.

207. See *id.* The "first sale" doctrine provides that "the owner of a particular copy . . . lawfully made under this title, or any person authorized by such owner, is entitled, without the authority of the copyright owner, to sell or otherwise dispose of the possession of that copy . . ." 17 U.S.C. § 109(a) (1994).

208. See *Novell*, 25 F. Supp. 2d at 1230.

209. See *id.*

wrap licenses are, on the whole, unenforceable because title is not retained by the program manufacturer after sale.

The position adopted by the *Novell* court, however, is inconsistent with the weight of authority.<sup>210</sup> Moreover, the *Novell* court effectively foreclosed redress to state contractual licensing rights by holding that computer programs are sold as goods. Although retailers and distributors of computer programs would be able to rely on shrinkwrap licenses, a producer's right of control would be terminated after the first sale.<sup>211</sup> Therefore, unbridled application of the first sale doctrine, as in *Novell*, may pose a significant disincentive to producers that contemplate investing in the development of new products.<sup>212</sup> The court's holding represented a broad policy determination of the nature better left to Congress.

### c. Copyright/Patent "Misuse" Doctrines

A third approach that gives implicit power to the courts to mediate between public and private rights to proprietary information relies on an expanded role of the patent and copyright "misuse" doctrines.<sup>213</sup> Copyright misuse occurs when a copyright is used to secure an exclusive right or limited monopoly beyond that granted by copyright law and which is contrary to public policy.<sup>214</sup> The patent misuse doctrine forbids the use of a patent in a similar manner or to accomplish similar ends.<sup>215</sup> The preeminent case on misuse in the computer software context is *Lasercomb America, Inc. v. Reynolds*.<sup>216</sup> In *Lasercomb*, the plaintiff had developed computer software that assisted in the designing of steel rule dies used to cut and score paper and cardboard.<sup>217</sup> The corporate defendant, Holiday Steel Rule Die Corporation, licensed several copies of the software from the plaintiff and created its own, virtually identical, program that it sold under its own label.<sup>218</sup> The plaintiff sued for copyright infringement, and the defendants asserted a defense of copyright misuse.<sup>219</sup> On the plaintiff's motion for summary judgment, the trial court found that the defendants' misuse

---

210. See *supra* notes 128-29 and accompanying text.

211. See Raymond T. Nimmer, *Breaking Barriers: The Relation Between Contract and Intellectual Property Law*, 13 Berkeley Tech. L.J. 827, 883 (1998) [hereinafter Nimmer, *Breaking Barriers*].

212. See *id.* at 831 ("The claim that first sale rights control is . . . a matter of turning the relationship between contract and copyright on its head.").

213. The misuse doctrine gives courts a means by which to prevent over-reaching to achieve anti-competitive ends. See *id.* at 870-71.

214. See *Lasercomb Am., Inc. v. Reynolds*, 911 F.2d 970, 977 (4th Cir. 1990).

215. See *Morton Salt Co. v. G.S. Suppiger Co.*, 314 U.S. 488, 491-92 (1942).

216. 911 F.2d 970 (4th Cir. 1990).

217. See *id.* at 971.

218. See *id.*

219. See *id.* at 972.

defense failed to create a genuine issue of fact.<sup>220</sup> The Fourth Circuit reversed, finding evidence of misuse in the fact that Lasercomb's standard license agreement prohibited licensees from creating competing software programs during the ninety-nine year term of the license.<sup>221</sup> According to the court, "this anticompetitive restraint" could exist "longer than the life of the copyright itself."<sup>222</sup> The court, after implying that patent misuse is analogous to copyright misuse,<sup>223</sup> further stated that the determinative question was whether the copyright was used in a manner that violated the public policy underlying a grant of copyright.<sup>224</sup> Thus, under *Lasercomb*, depriving licensees of the ability to create competing products is violative of the public policy of copyright if the prohibition extends for a longer term than does the copyright itself. Under this reasoning, given that contracts may be binding indefinitely,<sup>225</sup> contractual prohibitions against reverse engineering are likely unenforceable.<sup>226</sup>

The boundaries of the misuse doctrine have yet to be fleshed out completely by the courts.<sup>227</sup> It remains unclear which licensing provisions would be held violative of public policy, and what precisely would be considered an effort to secure an exclusive right or limited monopoly in excess of that granted by the federal copyright and patent laws.<sup>228</sup> Whether misuse is applicable to state trade secret claims is also unclear.<sup>229</sup> Finally, misuse has been recognized only as a defense to an action, and has yet to be permitted as an affirmative independent cause of action.<sup>230</sup>

## 2. Criticisms of Judicial Intervention

It is questionable whether the judicial forum is the most efficient means of determining when, and to what extent, public access to the ideas embedded in computer programs should be permitted.<sup>231</sup> The

---

220. See *Lasercomb Am., Inc. v. Holiday Steel Rule Die Corp.*, 656 F. Supp. 612, 616 (M.D.N.C. 1987) *rev'd*, 911 F.2d 970 (1990).

221. See *Lasercomb*, 911 F.2d at 978.

222. *Id.* (footnote omitted).

223. See *id.* at 977.

224. See *id.* at 978.

225. See *P.C. Films Corp. v. MGM/UA Home Video, Inc.*, 138 F.3d 453, 457-58 (2d Cir.) (holding that a "perpetual" license is enforceable, at least for the term of the copyright), *cert. denied*, 119 S. Ct. 542 (1998). Moreover, section 2B-308 ("Duration of Contract") provides a default rule of a perpetual license in certain instances. See Uniform Computer Act, *supra* note 129, § 2B-308(2)(b).

226. See Cohen, *Electronic Vigilantism*, *supra* note 82, at 1194-98.

227. See William E. Thomson, Jr. & Margaret Y. Chu, *Overstepping the Bounds: Copyright Misuse*, Computer Law., Nov. 1998, at 1, 5.

228. See *id.*

229. See *id.*

230. See Madison, *supra* note 16, at 1116 n.382.

231. Cf. *id.* at 1139 (advocating an express delegation of authority from Congress to federal courts in order to preserve the public domain). Allowing a court to strike contractual provisions based on its view of what proper policy should be, however,

last word from Congress was that computer programs are copyrightable.<sup>232</sup> In fact, CONTU expressly rejected trade secret and patent protection as an alternative to copyright protection for computer software.<sup>233</sup> Now, in addition to trade secret and patent protection, contractual provisions may also be relied upon for proprietary protection. Congress has not addressed the high level of intellectual property protection afforded computer programs. Therefore, whether it is desirable for courts to impose their own policy judgments regarding computer technologies in lieu of congressional direction on those matters is open to debate.

Both the public and private right owners deserve to know in advance the precise level of proprietary protection they can expect so that they may conduct themselves accordingly. Courts have the ability to make policy only on a case-by-case basis. Moreover, courts historically have resisted prodding to police the private market.<sup>234</sup> Further, a judicially constructed solution would place upon courts the onus of determining and responding to rapidly evolving market conditions and exigencies. Because courts have limited ability to gather and interpret information regarding industry practice, the risk of judicial error is quite high.<sup>235</sup> Finally, and most compelling, with regard to computer programs, it is not clear that the recognition of a public right to reverse engineer represents a desirable, or even workable, means of promoting access to computer software.<sup>236</sup>

### B. *A Private-Market Ordering*

While some have advocated judicial intervention, others suggest that only through the enforcement of private agreements will computer software be adequately protected. This view finds its support in the lack of preemption of contractual rights, economic theory, and the eroding of incentives to creation.

---

arguably undermines societal benefits produced by the commercialization of information. See Nimmer, *Breaking Barriers*, *supra* note 211, at 872.

232. See *supra* note 49 and accompanying text.

233. See Haynes, *supra* note 49, at 254.

234. Most courts refuse to recognize an independent good faith cause of action under Article 2 of the UCC. See Steven H. Hilfinger, Note, *K.M.C. Co. v. Irving Trust Co.: Discretionary Financing and the Implied Duty of Good Faith*, 81 Nw. U. L. Rev. 539, 550 (1987).

235. See Robert G. Bone, *A New Look at Trade Secret Law: Doctrine in Search of Justification*, 86 Cal. L. Rev. 241, 296 (1998).

236. The common theme that joins the approaches previously discussed is that prohibitions against reverse engineering represent an improper extension of intellectual property rights. See Nimmer, *Breaking Barriers*, *supra* note 211, at 867. Whether prohibitions against reverse engineering improperly extend, or merely buttress, pre-existing intellectual property rights hinges upon diametrically opposed constructions of the intellectual property laws. See *infra* notes 237-41 and accompanying text.

# 1. The Three Justifications

## a. *Lack of Preemption*

The argument that reverse engineering should be prohibited relies on the premise that the Constitution provides for intellectual property rights to give authors and inventors an incentive to creation.<sup>237</sup> Therefore, if producers believe that their works may be appropriated prematurely by others before they have an opportunity to recoup their investments, they will tend to undersupply the market for those works, which will lead to market failure.<sup>238</sup> Accordingly, these advocates argue, private producers must be permitted to prohibit reverse engineering to preserve this incentive. That private markets should determine the appropriate degree of re-use of information is the premise of the proposed draft of the Uniform Computer Information Transactions Act.<sup>239</sup>

Under this view, the federal intellectual property statutes represent a default system of rights that help to avoid the transaction costs involved in contracting with individual purchasers of proprietary material.<sup>240</sup> Nothing, however, prohibits the parties from creating their own set of proprietary rights between themselves through private contracts.<sup>241</sup> Therefore, if the private contractual prohibitions are not objectionable on grounds applicable to contracts in general, they are not subject to preemption by the federal laws.<sup>242</sup> In other words, freedom of contract reigns. Arguably, this view of the federal intellectual property laws as a set of default rules was implicitly adopted by the Seventh Circuit in *ProCD*.<sup>243</sup>

## b. *Economic Theory*

Given that computer program code is subject to quick and easy copying,<sup>244</sup> economic theory also sanctions prohibitions against reverse engineering. Under the microeconomic theory of perfect competition, the relevant market consists of products that not only are numerous, but also are fungible commodities.<sup>245</sup> Intellectual property furthers this model of competition and learning by allowing individuals access to others' ideas so that they may build incrementally upon those ideas.<sup>246</sup> Therefore, the intellectual property model fosters a

---

237. See *supra* note 17.

238. See *supra* note 146.

239. See Madison, *supra* note 16, at 1124.

240. See O'Rourke, *Copyright Preemption*, *supra* note 183, at 78.

241. See *id.*

242. See *id.*

243. See Madison, *supra* note 16, at 1124; O'Rourke, *Copyright Preemption*, *supra* note 183, at 81.

244. See *supra* note 146.

245. See Clapes, *supra* note 15, at 961.

246. See Kathryn D. Piele, *Three Years After Campbell v. Acuff-Rose Music, Inc.: What Is Fair Game for Parodists?*, 18 Loy. L.A. Ent. L.J. 75, 99 (1997).



system of market competition that results in competition based on price.<sup>247</sup> Yet competition based on price may be less than ideal in the computer program context. Firms competing on the basis of price may realize less profits than they would in a system of competition based on innovation and differentiated products.<sup>248</sup> The greater profits realized in a system of competition based on innovation may be utilized in the research and development of new products.<sup>249</sup> Thus, the recognition of a public right to reverse engineer is inconsistent with innovation, for free-riding competitors are allowed to simply disassemble and change pre-existing programs.<sup>250</sup> A prohibition against reverse engineering, on the other hand, forces innovation through the development of new technologies, not simply the reformulation of existing ones. The public interest in greater consumer choice and new innovative products is thereby furthered.

In addition, a right to reverse engineer may not be required to promote public access to computer software; natural market forces may further this goal without prompting from the courts and Congress.<sup>251</sup> Indeed, recent market phenomena occurring in the software industry suggest that natural market forces will prompt migration away from a closed proprietary model of computer software development and competition. The surging popularity of open source software,<sup>252</sup> which is software made available to users, typically over the Internet, in readable and comprehensible source code format, is evidence of this trend.<sup>253</sup> Open source software, however, does not necessarily mean the software is distributed for free,<sup>254</sup> and manufacturers of computer programs are able to maintain their proprietary rights in the original source code using various licenses.<sup>255</sup> The open source model

---

247. See Clapes, *supra* note 15, at 947-48.

248. See *id.* at 947.

249. See *id.* at 962.

250. See *id.* at 963; Miller, *supra* note 15, at 1026-27.

251. See Miller, *supra* note 15, at 1031.

252. See, e.g., Amy Harmon, *Open-Source Software Takes Surprise Leap in Popularity*, San Diego Union-Trib., Feb. 9, 1999, at 3 (stating that open source software was one of the year's most striking developments); Mark Leibovich, *The Spreading Grass-Roots Threat to Microsoft: Eric Raymond Crusades for "Open-Source" Software*, Wash. Post, Dec. 3, 1998, at A1 (noting that open source software has recently "forged dramatic mainstream inroads").

253. See Harvey Blume, *If You Love Your Online Business Set Your Code Free*, New Media, Jan. 1999, at 26, 26.

254. See Kevin Tolly, *A Reality Check on "Open Source"*, Network World, Feb. 8, 1999, at 22, 22.

255. See, e.g., Leander Kahney & Polly Sprenger, *Apple's Open-Source Movement*, (Mar. 15, 1999) <<http://www.wired.com/news/technology/story/18488.html?wnpgl>> (discussing the difference between free software and open source software); Malcolm MacLachlan, *Jini: Start of "Semi-Open" Source Movement*, (Jan. 28, 1999) <[http://www.techweb.com/printableArticle?doc\\_idB19990128S0001](http://www.techweb.com/printableArticle?doc_idB19990128S0001)> (noting that the Sun Community Source License is a combination of open source and proprietary models); Peter G. Neumann, *Robust Open-Source Software*, Comm. of the ACM, Feb. 1998, at 128, 128 (providing examples of open, but proprietary, software). The licenses typi-

of software development greatly reduces a company's research, development, and distribution costs because it allows the software to be refined by developers all over the world.<sup>256</sup> Because of the number of people who work on perfecting the software, open source software is more reliable and fails less often than closed source software.<sup>257</sup> Users of open source software are free to modify, customize, and otherwise adapt the program to their needs.<sup>258</sup> Additionally, the Internet is successful, in large part, because it operates as an open system of code.<sup>259</sup> Commentators predict that libraries of code will soon be available to users and programmers who desire to modify it and use it to suit their individual needs.<sup>260</sup>

The move toward an open source and reusable code system, widely regarded as a paradigm shift away from closed proprietary software,<sup>261</sup> is an example of competition by innovation, and signals a movement toward increased competition in markets for software sales, support, maintenance, and repair services.<sup>262</sup> The shift is strong evidence that competition in the software industry does not hinge solely on proprietary rights in program code. Whereas the closed proprietary model of computer software is analogous to a car sold with its hood welded shut,<sup>263</sup> the new open model of computer software recognizes the public's right to tinker. Public access to source code is viewed as commercially desirable, provided, of course, that private owners can retain their proprietary rights, through licenses, in the program code.<sup>264</sup> Accordingly, the necessity of reserving to the public a

---

cally allow users to alter the original source code, but all modifications must be sent back to the right holder. See Deborah Gage, *Should Java Go Open Source?* (Feb. 12, 1999) <<http://www.zdnet.com/zdnn/stories/news/0,4586,2208772,00.html>>.

256. See Blume, *supra* note 253, at 29.

257. See Charles C. Mann, *Programs to the People*, Tech. Rev., Jan.-Feb. 1999, at 36, 41.

258. See *id.* at 38.

259. See Blume, *supra* note 253, at 29.

260. See Haldiman, *supra* note 86, at 654. Object-orientated programming will make portions of computer code easily reusable. See *id.* at 653. Additionally, automated rights management technologies will provide manufacturers with the ability to police the use of their code, and allow users to make provision with the proprietary right holder for the use of that code so that they are not considered infringers. See Bell, *supra* note 132, at 564-67.

261. See, e.g., Josh McHugh, *For the Love of Hacking*, Forbes, Aug. 10, 1998, at 94, 100 (noting that open source software is an alternative to closed commercial software and arguing that perhaps Microsoft does not truly have a monopoly in the software industry); Levy, *Code Warriors*, *supra* note 4, at 60 (documenting shift towards open source software); Leibovich, *supra* note 252 (stating that open source software poses a significant threat to Microsoft's continued domination of the software industry); John Markoff, *Apple Adopts "Open Source" Code Strategy*, N.Y. Times, Mar. 17, 1999, at C1 (noting that Apple has embraced open source software).

262. Tim O'Reilly, *The Open-Source Revolution*, Release 1.0, Nov. 1998, at 3, 8-16.

263. See McHugh, *supra* note 261, at 100.

264. See *supra* note 256 and accompanying text.

right to reverse engineer to promote access is questionable at best, and imperfect at least.<sup>265</sup>

c. *Disincentive to Producers*

The recognition of a public right to reverse engineer may provide a significant disincentive to creators of computer programs. The research and development costs associated with program development are extremely high.<sup>266</sup> In the past, program producers were somewhat insulated from reverse engineering because it was an extremely difficult and uncertain process.<sup>267</sup> Therefore, manufacturers were provided with the lead-time and ability to recoup their investments before competitors were able to reverse engineer their programs and develop competing products.<sup>268</sup> Now, however, recent market developments may have greatly simplified the practice of reverse engineering, making it more difficult for program manufacturers to recoup their investments before a competing market substitute is launched. Numerous reverse engineering programs have become commercially available.<sup>269</sup> The widespread adoption of Java programming<sup>270</sup> raises additional fears among computer program manufacturers due to the ease with which Java programming may be reverse engineered.<sup>271</sup> If reverse engineering is becoming easier and more efficient, the argument for the enforceability of private prohibitions against reverse engineering gains renewed vigor.<sup>272</sup>

---

265. See Lemley & McGowan, *Legal Implications*, *supra* note 153, at 523-30.

266. See Troy Paredes, *Copyright Misuse and Tying: Will Courts Stop Misusing Misuse?*, 9 High Tech. L.J. 271, 309-15 (1994).

267. See Karjala, *Program Structure*, *supra* note 146, at 535.

268. See *id.*

269. See Craig Menefee, *PC Briefs* (July 2, 1998) <<http://www.newsbytes.com/news/98/114070.html>>; Godfrey Nolan, *Java: Easy Target for Illegal Re-Engineers* (Mar. 3, 1997) <[http://www.computerwire.com/cbr/1997/2206\\_29a.htm](http://www.computerwire.com/cbr/1997/2206_29a.htm)>.

270. See Lemley & McGowan, *Competitive Proprietary*, *supra* note 13, at 485.

271. See Nolan, *supra* note 269.

272. Moreover, U.S. policy regarding the legality of reverse engineering computer programs is ambiguous. When Japan considered expressly incorporating a reverse engineering exception into its copyright laws, the United States vigorously opposed the proposal, fearing that the exception would allow American software to be unduly appropriated by Japanese firms. See Graham & Zerbe, *supra* note 171, at 69. Further, while certain federal intellectual property statutes expressly recognize a right to reverse engineer, the copyright and patent statutes contain no similar exception. In the Semiconductor Chip Protection Act of 1984, see Pub. L. No. 98-620, tit. III, 98 Stat. 3347 (codified at 17 U.S.C. §§ 901-914 (1994)), Congress expressly recognized the right to reverse engineer. See 17 U.S.C. § 906(a); see also *Brooktree Corp. v. Advanced Micro Devices, Inc.*, 977 F.2d 1555, 1565-67 (Fed. Cir. 1992) (discussing the Act). Thus, the negative implication arising from the absence of specific allowance for reverse engineering under the copyright and patent statutes is that such a right was never intended by Congress, but was read into the law by courts addressing particularized market conditions. Evidence that supports this argument is that recently proposed legislation, if enacted, may have the impact of making reverse engineering a federal felony. See Mills, *supra* note 68, at 127-28. Additionally, the recently enacted Digital Millennium Copyright Act prohibits circumvention of "technological meas-

## 2. Criticisms of Private Agreements

Just as judicial intervention has been criticized, it is not clear that the private market will ultimately establish an appropriate balance between public and private rights. The debate concerning reverse engineering is magnified because shrinkwrap licenses are now generally enforced.<sup>273</sup> Private right holders may employ licenses in an effort to retain a level of control over their code not conferred by federal statutes.<sup>274</sup> By a simple license provision, a software producer may create exclusive rights virtually unlimited in scope. Those licenses often are employed to maintain rights through secrecy and proprietary control; sometimes they are used to preserve rights after public disclosure.<sup>275</sup> Shrinkwrap licenses also help to ensure that trade secret claims will not be preempted by providing the "extra element" of breach of promise or agreement.<sup>276</sup> Moreover, although shrinkwrap license provisions may be objectionable upon grounds applicable to contracts in general, the exclusive rights upon which private parties may agree are not circumscribed by statute, as they are under the federal copyright and patent statutes.

The exclusive proprietary rights conferred by the federal intellectual property statutes are limited both in time and in extent.<sup>277</sup> On the other hand, the rights available under state laws are not as restricted.<sup>278</sup> Contract protection for computer programs, in addition to copyright, patent, and trade secret protection, is a recent development that threatens to confer upon proprietary owners absolute control over their products for an indefinite period of time.<sup>279</sup> Therefore, the fundamental difference between the exclusive rights conferred under the federal statutes, and the rights conferred by state law, is that licenses and trade secrets, unlike copyrights and patents, are limited only by the private market. Given shrinkwrap's potential to swallow the intellectual property regime,<sup>280</sup> however, leaving this ordering solely to the private market may be disadvantageous to the public interest.

Intellectual property does not endeavor to confer upon owners unrestricted control over their creations; the objective is to give owners

---

ures" that control access to copyrighted works. See Digital Millennium Copyright Act, Pub. L. No. 105-304, 112 Stat. 2860 (1998). Query whether compiling source code into object code is a technological measure that controls access to a copyrighted program. If it is, one may argue that reverse engineering circumvents this process, and is, therefore, prohibited by the Act.

273. See *supra* notes 177-80 and accompanying text.

274. See Cohen, *Self-Help*, *supra* note 182, at 1090.

275. See *supra* note 255 and accompanying text.

276. See *supra* notes 97-107 and accompanying text.

277. See *supra* Part I.A-B.

278. See *supra* Part I.C-D.

279. See *supra* note 225. As a practical matter, however, the software is likely to be obsolete long before the expiration of the license.

280. See *supra* notes 131-33 and accompanying text.

only a sufficient incentive to create.<sup>281</sup> Indeed, “[s]ufficient incentive” . . . is something less than ‘perfect control.’”<sup>282</sup> Therefore, at some point, it arguably becomes inefficient to maintain that private owners may properly withhold works from the public sphere to preserve the incentive to creation.<sup>283</sup> Moreover, insofar as owners are granted exclusive control over their works, the cost of new works is raised, and, in certain situations, the public will not have access to the works.<sup>284</sup> Finally, a high level of exclusive control allows producers to prevent second-comers from building upon the unprotected ideas embodied in their works, which frustrates and unduly impedes the creation of “second-generation” works.<sup>285</sup>

The conflict that reverse engineering has posed is illustrative of the same tensions that govern intellectual property law: the difficulty in drawing the line between public and private rights. With regard to computer programs, access to a program’s source code, through reverse engineering or otherwise, is at times desirable.<sup>286</sup> Yet, unrestricted access to a program’s source code will create an imbalance in the intellectual property equation by depriving producers of the ability to recoup their investments.<sup>287</sup> If this occurs, there will exist no incentive to develop new products, which would be contrary to the public interest.<sup>288</sup> Therefore, as discussed below in part IV, to the extent that this dilemma cannot be addressed adequately by the courts or the private market alone, it is appropriate for Congress to intervene.

#### IV. AN INTERMEDIATE COMPROMISE FOR COMPUTER SOFTWARE

Some have proposed a compulsory licensing system for computer program code,<sup>289</sup> which would be the most “open” solution because it would allow interested parties unrestricted access to another’s code upon the payment of a compulsory fee.<sup>290</sup> Others have proposed a complete ban on reverse engineering, which would foster a “closed” system.<sup>291</sup> A more desirable solution would attempt to regulate be-

---

281. See *supra* note 17.

282. Lawrence Lessig, *Intellectual Property and Code*, 11 St. John’s J. Legal Comment. 635, 638 (1996).

283. See Lemley, *Beyond Preemption*, *supra* note 134, at 125; see also Lemley & McGowan, *Legal Implications*, *supra* note 153, at 526 (“[I]t does not make economic sense to give complete control over information to owners of intellectual property.”).

284. See Lemley, *Beyond Preemption*, *supra* note 134, at 124.

285. *Id.* at 125.

286. See *supra* notes 142-44 and accompanying text.

287. See *supra* notes 145-46 and accompanying text.

288. See *supra* notes 145-46 and accompanying text.

289. See Peter S. Menell, *Tailoring Legal Protection for Computer Software*, 39 Stan. L. Rev. 1329, 1371 (1987); Samuelson et al., *supra* note 16, at 2414-15.

290. See Samuelson et al., *supra* note 16, at 2414.

291. See *id.* at 2392-93 (“[A] complete ban [on reverse engineering] is contrary to basic norms of competition law.”).

tween these two extremes. The ultimate solution must be a compromise, not treading too heavily on the rights of owners, nor allowing those owners to wield their rights as a club against competition. The definitive and unambiguous solution must strike an appropriate balance between proprietary right holders and the public. To that end, Congress is the governing body primarily charged with determining public policy.<sup>292</sup> A decree by Congress would give right holders and the public advance notice of the proprietary rights available to them, and would also help to establish a middle ground between the judiciary and the private market.

Congress should give bolder force and effect to the constitutionally derived phrase "limited exclusive control."<sup>293</sup> Congress should allow proprietary owners the freedom to establish private state law rights, in addition to those rights conferred by federal laws, to provide authors and inventors of computer programs with a period of true exclusive control over their works.<sup>294</sup> Proprietary right owners would be permitted to use the federal patent and copyright laws as a default set of rules, which could be expanded or constricted as private individuals see fit. This way, producers are still given tremendous first-mover incentives, and are imbued with confidence that their proprietary rights are secure for the period of exclusive control.<sup>295</sup> During this period of limited competition, producers would be provided with the opportunity to establish their works as the consumer standard,<sup>296</sup> and to reap substantial profits. Fear that one's product will be appropriated prematurely by others would be alleviated. Thus, producers would be provided with the incentive to invest, facilitated by profits reaped during the period of exclusive control, in the creation of new and innovative products, rather than in the protection of previously marketed products.<sup>297</sup> Moreover, a high level of exclusive control in the

---

292. See Madison, *supra* note 16, at 1135.

293. See *supra* note 17.

294. If a program can be reverse engineered by competitors, the rights holder necessarily does not have exclusive control over his work. Further, as the intellectual property laws do not create monopolies, see Clapes, *supra* note 15, at 948-49, contracts necessary to help establish a period of true exclusive control.

295. Giving producers the ability to protect their software under copyright, patent, trade secret, and contract laws will provide them with the ability to preserve their natural lead-time in the market, which lead-time is necessary to prevent market failure. See Samuelson et al., *supra* note 16, at 2413.

296. Indeed, there is considerable value in familiarizing consumers with particular software. See *id.* at 2375. Second-comers will find it difficult to lure consumers away from software with which they are familiar and have spent time to learn. See *id.*

297. If producers are permitted to prevent the reverse engineering of their software, this will lessen the need for the producer to monitor the market for reverse engineering technologies, and expend resources on thwarting attempts at reverse engineering and on monitoring legislative efforts to address reverse engineering. See O'Rourke, *Drawing the Boundary*, *supra* note 104, at 499.

software industry may in fact be beneficial under certain circumstances.<sup>298</sup>

The period of exclusive control, however, should be limited by congressional decree that, after a period of time, as a condition to maintaining federal copyright and patent protection, program producers must disclose their source code.<sup>299</sup> This requirement effectively forecloses redress under state trade secret laws for those owners who desire to maintain federal intellectual property protection for their products.<sup>300</sup> State trade secret rights, and the ability to maintain the secrecy of one's software interfaces and compatibility controls, would be exchanged for certain and extensive protection under the federal statutes and contract laws.<sup>301</sup>

The rights conferred under the copyright and patent laws, of course, are not lost upon disclosure of a work to the public. Admittedly, private right owners have an interest in seeing that their creations are not appropriated by others, and this interest leads naturally to a desire to shroud their works in secrecy. The compromise is that in phase one, the period in which both state contract and trade secret and federal copyright and patent rights are entertained, producers can be certain of a high level of proprietary protection. In this phase, producers would be permitted to prohibit reverse engineering. Further, although reverse engineering would become unnecessary after disclosure, producers would maintain rights in their code under contract laws, rather than have those rights foreclosed by a general approach to federal preemption of shrinkwrap contracts. Moreover, owners that desire to maintain the secrecy of their source code are able to

---

298. Microsoft's dominance undeniably has produced numerous societal benefits, such as standardization. Professor Mark Patterson notes that, even if competition is eliminated, a standard-enhancing move may increase efficiency overall because consumers will benefit from the increased adoption of the standard. See Mark R. Patterson, *Coercion, Deception, and Other Demand-Increasing Practices in Antitrust Law*, 66 Antitrust L.J. 1, 74 n.323 (1997).

299. After the expiration of the standard-setting time period, the government, by requiring disclosure, would open the standard in order to encourage competition. See Lemley & McGowan, *Competitive Propriety*, *supra* note 13, at 506 (arguing that the law should encourage open standards); *supra* note 1 and accompanying text (contending that "openness" is good). The time at which disclosure must occur may be based on the date of the first commercial distribution of the software, see 17 U.S.C. §§ 904(a), 908(a) (1994), and should vary depending on the nature of the software program. For example, Congress could require that the source code to operating systems be disclosed after a shorter time period than application programs. For a discussion of the differences between operating systems and application programs, see Samuelson, *supra* note 3, at 676-82.

300. Nevertheless, this approach would be less intrusive upon the rights of software producers than a "public distribution equals publication" rule, which would deny trade secret protection to a program's source code even if the program was publicly distributed in object code format. Cohen, *Electronic Vigilantism*, *supra* note 82, at 1113-14.

301. Trade secrets are only protectible as long as no one discovers them. See *supra* note 91 and accompanying text.

"opt out" of protection under the federal statutes, and rely instead upon trade secret protection and contract to protect their products.<sup>302</sup>

Disclosure of a program's source code need not be made to the public at large. Rather, Congress should require only that the source code be placed on file with the respective statutory office.<sup>303</sup> Interested parties would be allowed to view the source code upon the payment of a processing fee. Part of this fee could be used to maintain the office, and any excess could be paid to the authors of the computer programs. The relevant office would keep records of who examined the source code, making it easier for right owners to prove access to their works if alleging copyright infringement.<sup>304</sup>

Additionally, even if a competitor is allowed to view a program's source code, that competitor, if he desires to produce a market substitute, must write his own program code,<sup>305</sup> or license aspects of the code from its owner.<sup>306</sup> Thus, a disclosure requirement is the first step toward promoting greater software re-use because those who desire to use another's code must know who owns what. If licensing permission from the owner is withheld, a second-comer would be free to use the ideas embedded in the code to produce a competing or interoperable

---

302. In fact, state law protection may provide more effective protection than federal intellectual property statutes in certain circumstances. See Jane C. Ginsburg, *Copyright, Common Law, and Sui Generis Protection of Databases in the United States and Abroad*, 66 U. Cin. L. Rev. 151, 151 (1997). At least one commentator has advocated an opt out approach to federal intellectual property. See Bell, *supra* note 132, at 615-16; see also Lemley, *Beyond Preemption*, *supra* note 134, at 150 (agreeing that an opt out scheme may be quite attractive to intellectual property owners).

303. The Copyright Office registration requirements mandate only that very small portions of a copyrighted program's source code be placed on file. See 37 C.F.R. § 202.20(c)(2)(vii) (1998). The Patent Office does not require that a program's source code be disclosed. See *supra* note 171 and accompanying text.

304. A plaintiff alleging copyright infringement must prove two elements: (1) ownership of a valid copyright and (2) copying of original elements of the work. See *Feist Publications, Inc. v. Rural Tel. Serv. Co.*, 499 U.S. 340, 361 (1991). Copying may be established through indirect evidence, including evidence that the alleged infringer had access to the copyrighted work. See *Castle Rock Entertainment, Inc. v. Carol Publ'g Group, Inc.*, 150 F.3d 132, 137 (2d Cir. 1998).

305. This Note does not go so far as to propose a system of compulsory licensing, for the success of the product in the marketplace in the phase in which both federal and state law rights are available may vary, and to cut off all control over the use of one's code after disclosure may not allow producers to recoup their investments in their software. Moreover, a recent study of the economics of information concluded that, provided competitors write their own source code, imitative competition in the software industry should be encouraged. See Karjala, *Program Structure*, *supra* note 146, at 536-38.

306. Right owners will object that their program code can be appropriated easily by second-comers who make minor changes and claim the disguised code as their own. See Miller, *supra* note 15, at 1026. The response is that simply changing another's code to hide all traces of illicit copying is not an easy task. See Cohen, *Electronic Vigilantism*, *supra* note 82, at 1123. In any case, this author is confident that courts are capable of determining whether one program so closely resembles another that it is infringing.



product. Clearly, both the public and private interests would be furthered by this balanced approach.

#### CONCLUSION

Unquestionably, computers will play an ever-expanding role in the lives of people in the generations to come. It is in the interests of both software producers and the public that the level of proprietary protection afforded computer programs be determined and clearly demarcated. Although much ink has been spilled regarding the appropriate level of protection that should be available to computer programs, the development of intellectual property protection for computer programs has been characterized by a haphazard and disjointed approach by individual courts addressing varying factual circumstances. This Note is yet another attempt at balancing the intellectual property equation. The approach suggested herein is advantageous because it endeavors to remain true to both oft-conflicting constitutional commands that authors and inventors be provided with an incentive to creation, and that this incentive is provided primarily for the benefit of the public. If Congress does not address the issue, however, determining the level of intellectual property protection for computer programs will be a task foisted upon the courts. Leaving that burden to the courts would be harmful because intellectual property protection for computer software, a nationally and internationally marketed product, will vary by jurisdiction. Moreover, if left unchecked, private owners may "shrink" the public sphere to an undesirable degree. Clearly, congressional action is needed. In the absence of congressional decree, this Note's approach, like many others, will remain little more than a good idea.

*Notes & Observations*