

3-1-2001

Open Source Software Licensing: Using Copyright Law to Encourage Free Use

Natasha T. Horne

Follow this and additional works at: <https://readingroom.law.gsu.edu/gsulr>

 Part of the [Law Commons](#)

Recommended Citation

Natasha T. Horne, *Open Source Software Licensing: Using Copyright Law to Encourage Free Use*, 17 GA. ST. U. L. REV. (2001).
Available at: <https://readingroom.law.gsu.edu/gsulr/vol17/iss3/6>

This Article is brought to you for free and open access by the Publications at Reading Room. It has been accepted for inclusion in Georgia State University Law Review by an authorized editor of Reading Room. For more information, please contact mbutler@gsu.edu.

OPEN SOURCE SOFTWARE LICENSING: USING COPYRIGHT LAW TO ENCOURAGE FREE USE

INTRODUCTION

The open source movement is gaining widespread attention due to a number of recent developments in the computer software industry.¹ With these recent developments, it is no wonder that the term “open source software,” or “free software,”² has become one of the hottest buzz words in the computer industry.

The computer industry continues to evolve at amazing and unprecedented speed. The Internet has, in a very short amount of time, gone mainstream. But with this fast-paced development and with the increased use and acceptance of computers, the software industry struggles to meet the demands of its users. Users seek rapid releases, easy customization, and quick bug

1. See, e.g., Michael DeVonish, *Bill Doesn't Live Here Anymore*, Forbes.com Digital Tool, at <http://www.forbes.com/tool/html/98/nov/1109/feat.htm> (Nov. 9, 1998) (stating that Microsoft openly recognized that the open source operating system, Linux, is outselling its Microsoft Windows 98 package); Ernie Longmire, *Feeling the Force of Open Source (Why Free Software Suddenly Matters)*, BASIS ADVANTAGE MAG., (visited Nov. 7, 2000) <http://www.basis.com/advantage/mag-v3n1/opensource.html> (1999) (“Apache is, simply put, the most popular Web server in the world, and it's open source.”); Tim O'Reilly, *Measuring the Impact of Free Software*, Web Review, at <http://webreview.com/wr/pub/freeware/oreilly.html> (last visited Nov. 7, 2000) (recognizing that Sendmail and BIND, both open source projects, are essential components of the Internet infrastructure and that Perl (open source) is the most popular programming language used in developing Web sites like Yahoo! and Amazon.com); Tim O'Reilly, *The Open-Source Revolution*, RELEASE 1.0 (Esther Dyson's Monthly Rep.), Nov. 1998, available at <http://www.edventure.com/release1/1198.html> [hereinafter O'Reilly, *Revolution*] (remarking that “[s]uddenly open source is the Next Big Thing,” and reporting that Netscape, Sun, and Apple plan to convert their proprietary programs to open source, and Oracle and Corel announce plans to port their products to the Linux platform); see also Matthew Broersma, *Open Source Gathering Steam*, ZDNet News, at <http://www.zdnet.com/zdnn/stories/news/0,4586,2173057,00.html> (Dec. 7, 1998) (reporting on recent developments in the open source marketplace).

2. “Free software” is the term used by the Free Software Foundation. See generally Richard Stallman, *Why Free Software Should Be Free*, at <http://www.fsf.org/philosophy/shouldbefree.html> (last modified Apr. 24, 1992) (“The word ‘free’ in ‘free software’ refers to freedom, not price; the price paid for a copy of a free program may be zero, or small, or (rarely) quite large.”).

fixes. Some critics believe that today's software industry cannot survive in this fast-paced marketplace without making some revolutionary changes. Their answer? Open Source Software.³

Part I of this Note reviews the history and philosophies of the open source movement. Part II discusses the roles copyright and software licensing play in open source software development. Part III examines the licensing terms of several popular open source licenses used today. Part IV provides a few pointers for selecting a license. Finally, Part V suggests that the open source movement may be disproving the need for financial incentives under copyright law.

I. WHAT IS THE OPEN SOURCE MOVEMENT?

A. What Is "Open Source"?

Unknown to most, the open source movement has been around for decades since the beginning of the development of the UNIX platform.⁴ However, only recently has the term gained attention through the growing popularity of several open source projects and through the increasing popularity of the Internet.⁵

The open source movement revolves around the notion of making software "open source," or in other words, making source code freely available to anyone.⁶ The word "open" refers to making source code freely available to other developers so that they can enhance the software, modify the code, or fix software bugs and glitches.⁷ The word "source" refers to source code.⁸ Source code consists of statements written by a software

3. See Frank Hecker, *Setting Up Shop: The Business of Open-Source Software*, at <http://people.netscape.com/hecker/setting-up-shop.html> (last modified Aug. 3, 1998) ("Commercial software companies face many challenges in growing their business in today's fast-moving and competitive industry environment. Recently many people have proposed the use of an open-source development model as one possible way to address those challenges."); Longmire, *supra* note 1.

4. See O'Reilly, *Revolution*, *supra* note 1.

5. See *supra* note 1 and accompanying text.

6. See Longmire, *supra* note 1.

7. See Dale Dougherty, *The Origins of Free and Open Source Software*, Web Review, at <http://webreview.com/pub/freeware/origins.html> (last visited Nov. 7, 2000).

8. See Robert W. Gomulkiewicz, *How Copyleft Uses License Rights To Succeed in the Open Source Software Revolution and the Implications for Article 2B*, 36 HOUS. L. REV. 179, 180 (1999).

programmer in a programming language,⁹ such as C, C++, Pascal, Fortran, Java, or Basic. These statements instruct the computer as to what commands to process.¹⁰ Software tools convert source code into object code.¹¹ It is actually the object code that the computer executes and understands.¹² However, it is very difficult for humans to read or modify this code.¹³

Historically, companies have made software available to third parties only in object code form.¹⁴ Companies view their source code as intellectual property, or as a trade secret, and therefore, rarely release it to third parties.¹⁵ In the event a company does decide to release the source code, it releases it under very strict licensing terms.¹⁶ The Open Source Movement hopes to change this proprietary way of thinking.¹⁷

B. What Is the Open Source Movement?

The open source movement comprises a community consisting primarily of developers and organizations interested in revolutionizing the software industry.¹⁸ The movement believes that in order for the software industry to meet the

9. *See id.*; *see also Source Code and Object Code*, at <http://www.whatis.com> (last modified Dec. 21, 1998).

10. *See Gomulkiewicz*, *supra* note 8, at 180 n.5 (citing *Apple Computer, Inc. v. Franklin Computer, Corp.* 714 F.2d 1240, 1243 (3d Cir. 1983), and Josh McHugh, *For the Love of Hacking*, FORBES, Aug. 10, 1998, at 94, 98).

11. *See Gomulkiewicz*, *supra* note 8, at 181; *Source Code and Object Code*, *supra* note 9.

12. *See Source Code and Object Code*, *supra* note 9.

13. *See id.*

14. *See id.*

15. *See Gomulkiewicz*, *supra* note 8, at 181 n.10 (citing Robert W. Gomulkiewicz & Mary L. Williamson, *A Brief Defense of Mass Market Software License Agreements*, 22 RUTGERS COMPUTER & TECH. L.J. 335, 359-60 (1996) ("noting that source code licenses 'pertain to sensitive information that may represent a company's most valuable business asset'"), and Maureen A. O'Rourke, *Drawing the Boundary Between Copyright and Contract: Copyright Preemption of Software License Terms*, 45 DUKE L.J. 479, 493-94 & n.56 (1995) ("stating that most license contracts are confidential")).

16. *See id.*

17. *See Stallman*, *supra* note 2.

18. *See A Roundtable Discussion on Open Source*, Amazon.com, Computers & Technology, at <http://www.amazon.com> (last visited Nov. 7, 2000) (mentioning journalists' "amazement that a complete multiuser, multitasking operating system can be built by researchers, companies, and volunteers cooperating over the Internet").

demands of this rapidly changing marketplace efficiently, developers must make source code freely, or openly, available.¹⁹

The open source development model suggests several advantages. First, allowing everyone to view and modify the source code results in higher quality software and rapid innovation.²⁰ Users from around the world can easily participate in the development and testing of a software product via the Internet or email.²¹ The community places great value in the high level of expertise possessed by the developers who write, modify, and enhance the code.²²

Further, through this collaboration, the product will develop quickly and efficiently.²³ In contrast to a proprietary development model, no schedules govern development under the open source model.²⁴ For example, as the need for increased functionality arises, any developer can implement code from around the world at any time without prior permission from the owner of the code.²⁵ Also, by making source code open for the world to use, modify, and reuse, developers can use any other developer's enhancements and modifications to the code when they have a need for similar functionality.²⁶

On the other hand, the open source development model has several shortcomings. First, no one entity maintains control over common development issues, such as compatibility.²⁷ This

19. See Chris DiBona et al., *Prologue to OPEN SOURCES: VOICES OF THE OPEN SOURCE REVOLUTION* (O'REILLY OPEN SOURCE) (Chris DiBona ed., 1999), available at <http://www.oreilly.com/catalog/opensources/book/intro.html> (stating the Free Software Foundation's theory that "source code is fundamental to the furthering of computer science and freely available source code is truly necessary for innovation to continue").

20. See Richard P. Gabriel & William N. Joy, *Sun Community Source License Principles*, at <http://www.sun.com/981208/scsl/principles.html> (last visited Nov. 7, 2000); Longmire, *supra* note 1.

21. See Longmire, *supra* note 1 ("The glue that holds this entire process together is the Internet.").

22. See Gabriel & Joy, *supra* note 20 ("[T]he Open Source approach recognizes that the primary value of a piece of software is the expertise represented by the people who developed it.").

23. See Longmire, *supra* note 1.

24. See Gabriel & Joy, *supra* note 20.

25. See Longmire, *supra* note 1 ("When a security related bug is discovered in an open-source package . . . a fix is often available within hours of the problem's discovery.").

26. See Gabriel & Joy, *supra* note 20.

27. See *id.*

commonly leads to fragmentation, also called “forking.”²⁸ Fragmentation results when multiple varying versions of a software package emerge in the marketplace.²⁹ These varying versions can easily confuse consumers.³⁰ Further, there is no organization responsible for the product.³¹ This could present problems when software bugs are too difficult for the user to fix and too low a priority for anyone else to fix.³² Also, development under this model can be “chaotic and undirected.”³³ Development depends solely on the interest of the developers and is rarely managed by one organization or individual.³⁴ Finally, in an open source model, “there are limited financial incentives for improvements and innovations.”³⁵ Often, the developers volunteer their time and participate only in their spare time.³⁶

It is noteworthy, however, that there are differing theories on how “open” open source should be.³⁷ For example, should a developer who has modified or enhanced open source software be required to give these modifications or enhancements back to the open source community?³⁸ Should proprietary code that

28. *See id.*

29. *See* Stephen Shankland, “Open Source” Infighting Grows, CNET News.com, at <http://news.cnet.com/news> (Feb. 19, 1999) (defining code forking as the “problem . . . in which programmers take a piece of software down two or more different paths”).

30. *See id.* When software developers do not work toward one common goal in writing code, software can easily “split into different and incompatible versions.” *Id.* This can create brand confusion in the marketplace. *See id.* In fact, some critics say that should Linux splinter off into incompatible versions, much like Unix did in the 1970s and 1980s, it could cripple the popularity of the operating system in the marketplace. *See* Malcolm Maclachlan, *Developers Try To Keep Linux from Forking*, InformationWeek.com News Flash, at <http://www.informationweek.com/story/TWB19990303S0035> (Mar. 3, 1999).

31. *See* Gabriel & Joy, *supra* note 20.

32. *See id.*

33. *Id.*

34. *See id.*

35. *Id.*

36. *See id.*; Gomulkiewicz, *supra* note 8, at 191.

37. *See* Malcom Maclachlan, *Panel Targets Differences in Open Source Movement*, TechWeb, Tech. News, at <http://www.techweb.com/wire/story/TWB19990304S0018> (Mar. 4, 1999). “[T]here is a spectrum of open source and community source models that range from the altruistic software is free view to the commercial view, as well as a myriad of hybrids that lie somewhere in between.” Michael J. Schallop, *The IPR Paradox: Leveraging Intellectual Property Rights To Encourage Interoperability in the Network Computing Age*, 28 AIPLA Q.J. 195, 243 (2000).

38. The GNU General Public License, for example, allows anyone to use, modify, and distribute the original source code, as long as the user agrees to give his modifications back to the open source community. *See* Free Software Foundation, *GNU General Public*

has been combined with open source code fall under the open source licensing terms?³⁹ When developing software under an open source development model, it is essential to select a license that corresponds with one's theory of open source development.⁴⁰

II. OPEN SOURCE SOFTWARE LICENSING

Because software is very easy to copy but very "difficult and expensive to create,"⁴¹ it is important to protect software through copyright law.⁴² The Copyright Act of 1976⁴³ (hereinafter "the Act") allows a software developer to retain control over the use of his work, and it protects his software code from unlawful copying and distribution.⁴⁴ Copyright offers "an inexpensive means of protecting . . . software from unauthorized copying and misappropriation."⁴⁵

In 1980, Congress amended the Act to include copyright protection for computer programs.⁴⁶ Under the Act, computer programs are considered "literary works" and are protected from unauthorized copying.⁴⁷ The legislative history of

License Version 2, at <http://www.opensource.org/licenses/gpl-license.html> (last modified June 1991) [hereinafter *GNU General Public License*].

39. See, e.g., *id.* One commentator suggests that the GNU open source model is radical and "contains a very end user-centric brand of idealism that prevents long-term monopolies by preventing short-term monopolies, but at the price of severely constraining developers' opportunities for investment recovery." Stig Hackv n, *Not Quite Open Source, But Closer: An Analysis of Sun's New Java License*, Linux World.com (Dec. 1998), at <http://www.linuxworld.com/linuxworld/lw-1998-12/lw-12-java.html> (last visited Nov. 7, 2000). In contrast, other open source licenses allow "proprietary relicensing of derivative works as a practical matter of doing business, trusting the ethics and logic of cooperation to ensure the long-term survival of open source." *Id.*

40. See discussion *infra* Parts III-IV.

41. FREDERICK L. COOPER III, *LAW AND THE SOFTWARE MARKETER* 10 (1988).

42. See NAT'L COMM'N ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, *FINAL REPORT* (1998).

43. Copyright Act of 1976, Pub. L. No. 94-553, 90 Stat. 2541 (codified as amended at 17 U.S.C. §§ 101-1101 (1994)).

44. See 17 U.S.C. § 106 (1994).

45. COOPER, *supra* note 41, at 34.

46. Pub. L. 96-517, 94 Stat. 3015, 3028 (1980).

47. As defined in section 101 of the Act, a computer program consists of "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 (1994). Subsequent cases have interpreted the Act to provide protection from unlawful copying of a program's source or object code

section 102(b) makes clear that the “expression” adopted by the programmer is the copyrightable element in a computer program;⁴⁸ the actual processes or methods embodied in the program are not within the protectable scope of copyright law.⁴⁹

Copyright protects all forms of computer code and may also extend to program design documentation, including schematics and flowcharts;⁵⁰ user manuals and software documentation;⁵¹ the structure, sequencing, and organization of the code;⁵² and

version. *See* *Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240, 1243 (3d Cir. 1983) (source and object code), *cert. dismissed*, 464 U.S. 1033 (1984); *Williams Elec. Inc. v. Arctic Int'l, Inc.*, 685 F.2d 870, 876-77 (3d Cir. 1982) (object code); *Digital Communications Assocs., Inc. v. Softklone Distrib. Corp.*, 659 F. Supp. 449, 454 (N.D. Ga. 1987) (source and object code); *Midway Mfg. Co. v. Strohon*, 564 F. Supp. 741, 750 (N.D. Ill. 1983) (source and object code); *Hubco Data Prods., Corp. v. Mgmt. Assistance, Inc.*, 219 U.S.P.Q. (BNA) 450, 454 (D. Idaho 1983) (object code); *GCA Corp. v. Chance*, 217 U.S.P.Q. (BNA) 718, 720 (N.D. Cal. 1982) (source and object code).

48. “Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law.” H.R. REP. NO. 94-1476 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5667.

49. “In 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.” 17 U.S.C. § 102(b) (1994). Courts have used varying techniques to distinguish between copyrightable expression and the unprotectable methods and processes in computer programs. *See generally* Bruce G. Joseph, *Copyright Protection of Computer Software and Compilations*, in *ADVANCED SEMINAR ON COPYRIGHT LAW 1999*, at 7 (PLI Patents, Copyrights, Trademarks & Literary Property Course, Handbook Series No. GO-0075, 1999).

50. *See* *Synercom Tech., Inc. v. Univ. Computing Co.*, 462 F. Supp. 1003, 1013 n.5 (N.D. Tex. 1978); STEPHEN FISHMAN, *COPYRIGHT YOUR SOFTWARE 2/3* (1994).

51. *See* *Williams v. Arndt*, 626 F. Supp. 571, 578 (D. Mass. 1985); *Synercom Tech.*, 462 F. Supp. at 1014; FISHMAN, *supra* note 50, at 2/3.

52. *See* *Johnson Controls, Inc. v. Phoenix Control Sys., Inc.*, 886 F.2d 1173, 1175 (9th Cir. 1989) (“[S]tructure, sequence, and organization of the [program] are copyrightable.”); *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1239 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987) (“[C]opyright protection of computer programs may extend beyond the programs’ code to their structure, sequence, and organization.”); *Telemarketing Res. v. Symantec Corp.*, 12 U.S.P.Q.2d (BNA) 1991, 1993 (N.D. Cal. 1989) (“[C]opyright applies to the structure and organization of a computer program.”); *Pearl Sys., Inc. v. Competition Elec., Inc.*, 8 U.S.P.Q.2d (BNA) 1520, 1525 (S.D. Fla. 1988) (“Copyright protection of computer software is not limited to the text of the source or object code.”); *SAS Inst., Inc. v. S&H Computer Sys., Inc.*, 605 F. Supp. 816, 830 (M.D. Tenn. 1985) (“[C]opying of the organization and structural details” may constitute infringement.). *But see* *Plains Cotton Coop. Ass’n v. Goodpasture Computer Serv., Inc.*, 807 F.2d 1256, 1262 (5th Cir. 1987) (holding that sequence and organization, when dictated by market forces, is a non-copyrightable idea rather than copyrightable expression), *reh’g denied*, 813 F.2d 407 (5th Cir. 1987), and *cert. denied*, 484 U.S. 821

the program's user interface.⁵³ Copyright also extends to new material that the author adds to a previously existing work.⁵⁴ For example, a derivative work may include an updated version of an existing program or a translation of a program to another language platform.⁵⁵

Once a work is copyrighted, the author has secured a bundle of exclusive rights. Under section 106, these rights include the right to reproduce the work; the right to create "derivative works" based on the work; the right to distribute copies of the work to the public by sale, license, or lease; the right to perform the work publicly; and the right to display the work publicly.⁵⁶

Although registration of a copyright is not required,⁵⁷ it is beneficial. Once registered, an author is "presumed the owner of the copyright in the material deposited with the registration."⁵⁸ The information submitted with the copyright registration form is presumed to be true in the absence of proof to the contrary.⁵⁹ If the author registers his work before infringement or within three months of publishing, and the author prevails in an infringement suit, he receives two additional benefits.⁶⁰ First, in the event of litigation, the court can order the infringer to pay attorney's fees and court costs.⁶¹ Second, the author may elect to have the court award statutory

(1987); FISHMAN, *supra* note 50, at 2/3.

53. See FISHMAN, *supra* note 50, at 2/3. *But see* Lotus Dev. Corp. v. Borland Int'l, Inc., 49 F.3d 807, 815 (1st Cir. 1995), *aff'd*, 516 U.S. 233 (1996) (holding that Lotus' menu hierarchy was not copyrightable because it constituted a method of operation).

54. See 17 U.S.C. § 103(b) (1994) ("The copyright in a . . . derivative work extends only to the material contributed by the author of such work, as distinguished from the preexisting material employed in the work . . .").

55. See 1 INTELLECTUAL PROPERTY COUNSELING & LITIGATION § 11.06 (2000).

56. 17 U.S.C. § 106 (1994).

57. *Id.* § 408(a) ("[R]egistration is not a condition of copyright protection.").

58. *Id.* § 410(c) ("In any judicial proceedings the certificate of a registration made before or within five years after first publication of the work shall constitute prima facie evidence of the validity of the copyright and of the facts stated in the certificate."); see FISHMAN, *supra* note 50, at 4/3.

59. 17 U.S.C. § 410(c) (1994).

60. See FISHMAN, *supra* note 50, at 4/3.

61. 17 U.S.C. § 412 (1994) ("In any action under this title, . . . no award of statutory damages or of attorney's fees . . . shall be made for . . . (2) any infringement of copyright commenced after first publication of the work and before the effective date of its registration, unless such registration is made within three months after the first publication of the work."); FISHMAN, *supra* note 50, at 4/4.

damages up to \$100,000 per infringement without having to establish damages actually suffered.⁶²

Infringement occurs when someone “exploits one or more of the . . . owner’s exclusive rights without . . . permission.”⁶³ This usually involves the unauthorized reproduction or distribution of the work, or the unauthorized creation of derivative works based on it.⁶⁴

A. Licensing Generally

When an owner of a work “decides to transfer to one or more people the use and enjoyment of part or all of his ‘bundle of legal rights,’” yet wishes to retain ownership of these rights, he uses a license.⁶⁵ The license “defines the nature and extent of [the granted] permission.”⁶⁶ Normally, the original owner of the bundle of rights grants the license.⁶⁷ Sometimes, however, a license allows a licensee to grant sub-licenses.⁶⁸

B. Proprietary Software Licensing

In a proprietary software development model, code is first copyrighted and then distributed under a license agreement that gives its users special rights.⁶⁹ Proprietary licenses usually “restrict software to execute-only format” and limit the number of installations allowed per copy of software.⁷⁰ Generally, only those users who have bought the software can use it.⁷¹ Further, as mentioned above, a developer or a developing organization

62. 17 U.S.C. § 412(2) (1994). Section 504(c)(1) of the Act allows for an award of statutory damages “in a sum of not less than \$500 or more than \$20,000 as the court considers just.” *Id.* § 504(c)(1). When the court finds that infringement was committed willfully, “the court in its discretion may increase the award of statutory damages to a sum of not more than \$100,000.” *Id.* § 504(c)(2). For a more detailed discussion about infringement of copyrighted software, see David W. Carstens, *Legal Protection of Computer Software: Patents, Copyrights, and Trade Secrets*, 20 J. CONTEMP. L. 13, 59-64 (1994).

63. FISHMAN, *supra* note 50, at 27.

64. *See id.*

65. COOPER, *supra* note 41, at 12.

66. FISHMAN, *supra* note 50, at 12/9.

67. *See* COOPER, *supra* note 41, at 12.

68. *See id.*

69. *See* Gabriel & Joy, *supra* note 20.

70. *Id.*

71. *See id.*

rarely makes its source code available.⁷² When it is made available by the software developer, it is usually at an extra charge, and it is only made available for limited purposes.⁷³ Licensing agreements in a proprietary market help the owners “preserve their market share, obtain the maximum return on each transaction, and . . . safeguard intellectual property rights.”⁷⁴

C. Open Source Software Licensing

Ironically, licensing and copyright law are central to the open source movement, as well. Licensing is essential in allowing the original developer to maintain control over the future free use of his source code and to license away rights normally protected by copyright law.⁷⁵ Once an open source developer has copyrighted his code, thereby establishing property rights in the code, he will use licensing terms that carry out his desired open source development model.⁷⁶

D. Open Source Licensing Terms

As mentioned above, licensing terms will vary depending on a developer’s theory on how “open” his open source software should be. Developers should thoroughly review the terms in the licensing agreement before they begin working on an open source project or before they select a license to govern their work.⁷⁷ The following paragraphs highlight a few common

72. *See id.*

73. *See id.*

74. FISHMAN, *supra* note 50, at 12/9.

75. *See GNU General Public License, supra* note 38, at Preamble. “[T]he General Public License is intended to guarantee your freedom to share and change free software—to make sure that software is free for all its users.” *Id.* Though the “simplest way to make a program free is to put it in the public domain . . . uncopyrighted . . . [u]ncooperative people . . . [could] convert the program into proprietary software . . .” Free Software Foundation, *What Is Copyleft?*, at <http://www.fsf.org/copyleft/copyleft.html> (last visited Nov. 7, 2000).

76. The idea is “to use copyright law, ‘but flip[] it over to serve the opposite of its usual purpose: instead of a means of privatizing software, [copyright] becomes a means of keeping software free.’” Teresa Hill, Note, *Fragmenting the Copyleft Movement: The Public Will Not Prevail*, 1999 UTAH L. REV. 797, 798 & nn.8 & 10 (1999) (quoting Richard Stallman, *The GNU Project*, at <http://prep.ai.mit.edu/gnu/thegnuproject.html> (last visited Mar. 6, 1999)).

77. *See* discussion *infra* Parts III-IV.

licensing terms used in various open source development models.

1. Code Cannot Be Used in Proprietary or Commercial Software

Some licenses may restrict the use of open source code in proprietary or commercial software.⁷⁸ Developers who voluntarily release code to the general public without having received financial return may want to use a license to keep companies or individuals from reaping the financial benefits of their efforts.⁷⁹

2. Author's Attribution and Integrity

Many developers who work on open source projects volunteer their work in exchange for the recognition and reputation well-written code gives them within the development community.⁸⁰ Therefore, developers often seek licensing terms that require future licensees to acknowledge them in derived works that use their code.⁸¹ Licensees may be able to meet this requirement by retaining copyright notices on the code as they pass it on or modify it.⁸² The license may also require that the licensee label the derivative work with a different version number, or that he distribute the source code unmodified along with a mechanism that combines the code with any future modifications when the software is compiled for use by the computer.⁸³ This way, the original code remains intact, and future licensees do not attribute later modifications to the original developer.⁸⁴ Licenses may also “prohibit the use of the name of the author of given code to endorse or promote products derived from that code.”⁸⁵

78. See, e.g., *The Aladdin Free Public License Version 9*, at <http://dv.go.dlr.de/fresh/unix/src/misc/ghost/aladdin/PUBLIC> (last modified Sept. 18, 2000) [hereinafter *Aladdin Free Public License*].

79. See Debian GNU/Linux, *What Does Free Mean? Or What Do You Mean by Open Software?*, at <http://www.debian.org/intro/free> (last visited Nov. 7, 2000).

80. See Gomulkiewicz, *supra* note 8, at 187-88, n.58 (citing Eric S. Raymond, *Homesteading the Noosphere*, available at <http://www.tuxedo.org/esr/writings/homesteading/homesteading-6.html> (last visited Dec. 19, 1998)).

81. See, e.g., *The BSD License*, at <http://www.opensource.org/licenses/bsd-license.html> (last visited Nov. 7, 2000).

82. See Gomulkiewicz, *supra* note 8, at 187.

83. See *id.* at 188.

84. See *id.*

85. *Id.*; see, e.g., *The BSD License*, *supra* note 81 (“Neither name of the

3. *Forced Distribution*

Licenses often contain terms that require future developers who modify or enhance open source code to return their modifications and enhancements to the open source community.⁸⁶ Allowing a licensee to modify the source code or fix bugs without returning the modifications or enhancements to the public is often thought to counter several of the advantages of developing code under an open source model, such as the benefit of rapid product enhancement through collaboration.⁸⁷

4. *Forced Use of the Same License*

Some licenses require that works derived from the source code be subject to the same licensing terms of the original license.⁸⁸ The Open Source Movement needs this to protect the openness of the source code and to keep the original author's licensing wishes intact.⁸⁹

5. *Distribution of Source Code*

Open access to the source code is fundamental to the Open Source Movement because it allows for easy repair and modification of a program.⁹⁰ Thus, some licenses require that the licensee distribute software with its source code.⁹¹ If a developer cannot distribute a product with its source code, then usually the license requires that the developer give a well-known (or well-advertised) means of obtaining the source code

<ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.”).

86. See, e.g., *GNU General Public License*, *supra* note 38.

87. See *Rationale for the Open Source Definition*, at <http://www.opensource.org/osd-rationale.html> (last visited Nov. 7, 2000). (“Since our purpose is to make [software] evolution easy, we require that modification be made easy.”).

88. See, e.g., *GNU General Public License*, *supra* note 38.

89. See *id.* (Preamble).

90. See *Rationale for the Open Source Definition*, *supra* note 87.

91. See, e.g., *Mozilla Public License Version 1.1*, at <http://www.mozilla.org/MPL/MPL-1.1.html> (last visited Nov. 7, 2000) [hereinafter *Mozilla Public License*].

without additional charge (excluding non-content-based distribution fees).⁹²

6. Unencumbered Redistribution of Derivative Works

At the heart of the Open Source Movement is the notion that licensees should be freely able to view, modify, and reuse the original source code.⁹³ Thus, licenses usually explicitly grant licensees the right to create derivative works without first acquiring permission or paying royalties.⁹⁴ Further, the license usually explicitly permits the licensee to distribute the software built from modified or derivative source code.⁹⁵ However, licenses differ on whether the developer must make the source code of derivative works available.⁹⁶

7. No Warranties

Most open source licenses require that licensees accept the software “as is.”⁹⁷ This shifts the risk away from the code developer to the user.⁹⁸ If open source developers were required to assume liability for defects in their code, they would be less willing to contribute code voluntarily or without cost.⁹⁹

The need to deflect risk may lead to the incorporation of “additional contractual terms to standard open source

92. See Bruce Perens, *The Open Source Definition*, in OPEN SOURCES: VOICES FROM THE OPEN SOURCE REVOLUTION (O'REILLY OPEN SOURCE), (Chris DiBona ed., 1999), available at <http://www.oreilly.com/catalog/opensources/book/perens.html>.

93. See Longmire, *supra* note 1; *The Open Source Definition Version 1.7*, at <http://www.opensource.org/osd.html> (last visited Nov. 7, 2000) [hereinafter *The Open Source Definition*] (“The license must allow modifications and derived works . . .”).

94. See *The Open Source Definition*, *supra* note 93 (“The license may not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license may not require a royalty or other fee for such sale.”).

95. See, e.g., *The BSD License*, *supra* note 81 (“Redistribution and use in source and binary forms, with or without modification, are permitted provided that . . . [certain] conditions are met . . .”).

96. Compare *GNU General Public License*, *supra* note 38, with *The BSD License*, *supra* note 81.

97. See, e.g., *Mozilla Public License*, *supra* note 91.

98. See Gomulkiewicz, *supra* note 8, at 191.

99. See *id.* at 191-92. “[L]ow risk’ . . . means low barriers to entry . . .” *Id.* at 192. Everyone, including people who cannot afford legal protection, can contribute to the code-writing process. See *id.* Low risk will also promote and encourage development. See *id.* (citing Virginia I. Postrel, *Hooray for Risk*, FORBES, Dec. 4, 1995, at 106).

licenses.”¹⁰⁰ For example, “choice of law and choice of venue clauses may become more important as certain jurisdictions make it more difficult to disclaim warranties, limit liability, or enforce standard-form agreements.”¹⁰¹

8. Self-Perpetuating License Terms

Licenses often require that the licensing terms apply to everyone to whom the software is redistributed.¹⁰² They also require that licensees pass on the licensing terms unaltered to subsequent licensees.¹⁰³ This keeps the licensing terms (and the intent of the original developer) intact.¹⁰⁴

9. Non-Discrimination and Non-Contamination

Some open source licenses require that the licensee not discriminate against any individual or group, and that it not restrict the use of the software in a particular field or endeavor.¹⁰⁵ Further, they often require that the license not place restrictions on or apply to separate software programs, such as proprietary software, that has been distributed along with the open source code.¹⁰⁶

100. *Id.* at 192.

101. *Id.*

102. See Perens, *supra* note 92; *The Open Source Definition*, *supra* note 93.

103. See, e.g., *GNU General Public License*, *supra* note 38.

104. See Perens, *supra* note 92.

105. See *The Open Source Definition*, *supra* note 93. “In order to get the maximum benefit from the process, the maximum diversity of persons and groups should be equally eligible to contribute to open sources.” *Rationale for the Open Source Definition*, *supra* note 87.

106. See *The Open Source Definition*, *supra* note 93. For example, under the Open Source Definition, a license may not require that all programs distributed on the same distribution medium be open source. See Perens, *supra* note 92. In contrast, however, a license may “contaminate” other code if it is linked at runtime. See *Rationale for the Open Source Definition*, *supra* note 87; see also *GNU General Public License*, *supra* note 38.

III. OPEN SOURCE LICENSES

Several examples of open source software licenses exist on the Internet.¹⁰⁷ Before selecting a license or contributing to an open source project, it is important that the developer understand the licensing terms and restrictions of the agreement.¹⁰⁸ The following sections compare and contrast the most popular open source licenses that are available on the Internet.

A. The GNU General Public License (GPL)

The fundamental goal of the Free Software Foundation¹⁰⁹ (hereinafter “FSF”) is to keep free software “free” or, in other words, to allow anyone to use, modify, and distribute original software code and any derived versions thereof freely.¹¹⁰ In order to accomplish this, the FSF has developed a licensing theory called “copyleft” (which is a pun on copyright).¹¹¹ Under this theory, the GPL¹¹² allows a user to copy, modify, and distribute GPL’d code, so long as the user agrees to pass on to other users these licensing rights unimpaired with all derivative versions of the code.¹¹³ Therefore, anyone who develops software

107. See, e.g., *GNU General Public License*, *supra* note 38; *The BSD License*, *supra* note 81; *Mozilla Public License*, *supra* note 91.

108. See Perens, *supra* note 92.

109. “The Free Software Foundation (FSF) is dedicated to eliminating restrictions on copying, redistribution, understanding, and modification of computer programs.” *Free Software Foundation*, at <http://www.fsf.org/fsf/fsf.html> (last modified May 2, 1999).

110. See *id.*; Larry Seltzer, *License To Drive Software Development*, *PC MAG.*, Mar. 23, 1998, available at <http://www.zdnet.com/pcmag/features/opensource/intro2.html>; *GNU General Public License*, *supra* note 38, at Preamble (“The GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users.”); Stallman, *supra* note 2.

111. See generally Free Software Foundation, *supra* note 75.

112. For an article discussing the enforceability of the GPL, see Patrick K. Bobko, *Linux and General Public Licenses: Can Copyright Keep “Open Source” Software Free?*, 28 *AIPLA Q.J.* 81 (2000).

113. See *GNU General Public License*, *supra* note 38. Interestingly, as Tim O’Reilly, founder of O’Reilly and Associates, has pointed out, the GPL assumes that software will be distributed. See *GPL Problem?*, O’Reilly Column, at http://www.oreilly.com/ask_tim/gpl_problem.html (last visited Oct. 17, 1999) (Tim O’Reilly’s response to a question from Fred Mobach). However, many applications, such as Amazon.com or maps.yahoo.com, are not distributed. See *id.* Rather, they can be built on top of open source software without being required under the license to give improvements back to the open source community. See *id.*

based on code licensed under the GPL must grant the public free use, modification, and distribution of the derived work.¹¹⁴ Thus, if the developer uses GPL'd code in a proprietary program containing proprietary source code, then the developer must make available the proprietary source code under the terms of the GPL, unless the sections not derived from the GPL'd code can be reasonably considered, and are distributable, as independent and separate works.¹¹⁵ This "tainting effect" may pose problems for commercial entities, such as Netscape, that are considering converting proprietary software into an open source product that includes either a third party technology or that shares source code with other products that they are not converting to open source.¹¹⁶

In addition to agreeing not to establish proprietary rights in the software itself, or in subsequent versions thereof, and to pass on changes and modifications to the software, users must agree to provide the source code to anyone they give the object code.¹¹⁷ Therefore, developers must ensure that anyone who receives the software from them in object code form has unconditional access to the source code.¹¹⁸ A developer can meet this requirement by distributing the software with the source code or by providing instructions on how to access the source code.¹¹⁹ Further, developers must include notices of the GPL in the software.¹²⁰ Thus, all source files must show notice of the application of the GPL to the code and any derived works.¹²¹ Any modified and distributed version of the software must also indicate that it is not the original and must provide the above notices on the computer screen during program execution.¹²²

114. A derivative work based on the original code includes any work that is distributed or published "that in whole or in part contains or is derived from the program or any part thereof . . ." *GNU General Public License*, *supra* note 38, § 2b.

115. *See id.* § 2(c). However, "mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License." *Id.*

116. *See* Hecker, *supra* note 3; *Netscape Public License FAQ*, at <http://www.mozilla.org/MPL/FAQ.html> (last visited Nov. 7, 2000).

117. *See GNU General Public License*, *supra* note 38, § 3(a)-(c).

118. *See id.*

119. *See id.*

120. *See id.* § 1.

121. *See id.*

122. *See id.* § 2(a), (c).

Finally, licensees must accept the software without warranties of any kind.¹²³

A licensee accepts the terms of the license by modifying or distributing the program's source code (or the source code of a derived work).¹²⁴ Additionally, any attempt to copy, distribute, or sub-license the code under terms not expressly provided for in the GPL will automatically terminate one's rights under the license.¹²⁵

B. The Berkeley Software Distribution License (BSD)

The BSD license, first used for the Berkeley Software Distribution of Unix, is known to be the least restrictive of the open source licenses available today.¹²⁶ The license is unlike the GPL in many ways.

The BSD license grants unlimited rights to use and distribute modified or unmodified source and binary code, provided that the licensee has met a number of conditions.¹²⁷ First, licensees must agree to keep intact the copyright notices, licensing terms, and disclaimer of warranties with each distribution of the source code.¹²⁸ Similarly, when distributing the code in binary form, licensees must reproduce the copyright notices, licensing terms, and disclaimer of warranties in the documentation and other materials distributed with the code.¹²⁹ Lastly, the BSD license restricts licensees from using the name of the university, or the names of its contributors, to endorse or promote derivative works without prior permission.¹³⁰

123. *See id.* §§ 11-12.

124. *See id.* § 5.

125. *See id.* § 4.

126. *See* Hecker, *supra* note 3; Seltzer, *supra* note 110.

127. *See The BSD License*, *supra* note 81.

128. *See id.*

129. *See id.*

130. *See id.* Unlike other open source licenses, earlier versions of the BSD license also required that licensees include an advertising clause in all advertising materials that mention features or use of the software. While the absence of the previously mentioned terms often subjected the license to much criticism, it was the advertising clause that seemed to subject the license to the most criticism. *See* Hecker, *supra* note 3. However, this advertising requirement has recently been reneged from all prior versions of the license. *See* Posting of Lloyd Lim, lim@bmerc.berkeley.edu, to openmash-cvs@openmash.org (Dec. 11, 2000), available at <http://www-plateau.cs.berkeley.edu/mhonarc/openmash-cvs/msg00052.html>.

Unlike the GPL and most other open source licenses, the BSD license does not require that any modifications or enhancements of the original code be contributed back to the open source community.¹³¹ Similarly, the BSD license does not contain any provisions that keep licensees from making the code proprietary.¹³² Though many open source advocates prefer licenses that require licensees to provide source code of derivative works back to the open source community,¹³³ the BSD license may be an appropriate option for commercial entities because it avoids the “tainting effect” of the GPL and provides licensees the option of making their enhancements and modifications proprietary.¹³⁴

C. The Aladdin License

The Aladdin license resembles the GPL but provides additional restrictions.¹³⁵ First, the license explicitly prohibits a

131. See Hecker, *supra* note 3.

132. See *id.*; *Netscape Public License FAQ*, *supra* note 116.

133. See Hecker, *supra* note 3.

134. See Brian Behlendorf, *Open Source as a Business Strategy*, in *OPEN SOURCES: VOICES OF THE OPEN SOURCE REVOLUTION (O'REILLY OPEN SOURCE)* (Chris DiBona ed., 1999), available at <http://www.oreilly.com/catalog/opensources/book/brian.html>.

135. Similar to the requirements of the GPL, if a developer plans to distribute or publish any work that contains or is derived from code licensed under Aladdin, then (1) the work must contain prominent notices stating it is a modified version and the dates of the changes made; (2) the work must be licensed as a whole to all third parties under the terms of the license; (3) copyright notices, instruction on how to view the license, and disclaimers of warranties must be displayed each time the work commences operation, if the work reads commands interactively when run; (4) the work must be accompanied by the complete corresponding source code; (5) any distributed written or printed material must include a written copy of the license or a notice that the work is covered by the license and instructions on how to view the license; and (6) no further restrictions other than those enumerated in the license may be imposed on the recipient of the code. See *Aladdin Free Public License*, *supra* note 78, § 2(c)(i)-(vi). However, the Aladdin license claims that it is not an open source license because it places restrictions on the distribution of the program. See *id.*

Aladdin Enterprises respects and supports the philosophy of the Open Source Definition, and shares the desire of the GNU project to keep licensed software freely redistributable in both source and object form . . . [However, t]his License attempts to ensure that those who receive, redistribute, and contribute to the licensed Program according to the Open Source and Free Software philosophies have the right to do so, while retaining for the developer(s) . . . the power to make those who use the Program to enhance the value of commercial products pay for the privilege of doing so.

Id.

commercial organization from accepting money for the software, except in limited circumstances to cover the costs of distribution of the code.¹³⁶ Second, licensees may not distribute a “free” version of the software in a distribution medium containing paid-for software.¹³⁷

As long as the licensee complies with the restrictions of the Aladdin license and distributes an unmodified copy of the license with the code (though license modifications may include a description of the licensed work and the law of the country where the work was created), the licensee may freely copy and distribute literal copies, modified copies, or derived versions of the software’s source code throughout the world, in any medium.¹³⁸

As mentioned above, the Aladdin license explicitly restricts distribution of the program or any derivative works by a commercial organization to a third party if it receives payment in connection with the distribution.¹³⁹ The license allows distribution fees as long as the fees are not content-dependent.¹⁴⁰ Also, the software must be distributed independent of any other product or service and may not be distributed in a medium containing paid-for software.¹⁴¹

D. The Mozilla Public License (MPL) and the Netscape Public License (NPL)

In March 1998, Netscape announced its plan to release its source code for the Netscape Communicator and to implement an open source strategy.¹⁴² Specifically, Netscape announced

136. *See id.* § 2(a).

137. *See id.*

The following methods of distribution involving payment shall not in and of themselves be a violation . . . [:] (ii) Distributing the Program on removable computer-readable media, provided that the files containing the Program are reproduced entirely . . . , that all information on such media be re-distributable for non-commercial purposes without charge, and that such media are distributed by themselves (except for accompanying documentation) independent of any other product or service.

Id.

138. *See id.* § 1.

139. *See id.* § 2(a).

140. *See id.* § 2(a)(i). This means that the fees should be the same for products consisting of the same amount of random data. *See id.*

141. *See id.* § 2(a).

142. *See Netscape Communicator Source Code Frequently Asked Questions, at*

two reasons for releasing the Communicator source code. First, using an open source development model will result in a higher quality, more full-featured product due to increased scrutiny of the code and the expansion of the developer community.¹⁴³ Second, releasing the source code will encourage other developers to use Netscape technology in their own products, perhaps resulting in more user accessibility and broader platform support.¹⁴⁴

The MPL and the NPL, authored by Netscape Communications, strike “a balance between the BSD license and the GPL.”¹⁴⁵ In developing these licenses, Netscape attempted “to strike a middle ground between promoting free source development by commercial enterprises and protecting free source developers.”¹⁴⁶ These licenses were implemented in order to ensure that developers return their modifications to the open source community, and they assist Netscape in retaining specific rights that allow it to continue proprietary development of other packages and maintain existing contracts with third parties.¹⁴⁷ These licenses serve as a good model for any commercial company looking to develop an open source product, to convert a proprietary product to open source, or to

<http://www.mozilla.org/MPL/FAQ.html> (last modified Sept. 24, 1999).

143. *See id.*

144. *See id.*

145. Stig Hackv n, *A Quick Survey of Open Source Licenses*, at <http://devlinux.org/lm/survey.html> (last modified Apr. 5, 1999). Netscape did not initially release its code under the GPL for a number of reasons. *See Netscape Public License FAQ*, *supra* note 116 (Question 10). First, under the GPL, Netscape would be required to release the source code of third party components, which it has no right to do. *See id.* Second, in order to be able to distribute its code worldwide, Netscape may not make the source code for cryptography available in an unrestricted way. *See id.* The GPL would require Netscape to release portions of the Communicator source code which it must keep proprietary. *See id.* Third, Netscape wants commercial developers to contribute to the open source project without being required to release the source code for their programs under all circumstances. *See id.* Surprisingly, however, Netscape recently announced its plan to relicense the code under an MPL-GPL dual license. *See Mozilla Relicensing FAQ*, at <http://www.mozilla.org/MPL/mozilla-reliance-faq.html> (last modified Aug. 16, 2000). Mozilla.org has decided to go forward with the attempt to dual-license in order to bring the “mozilla source . . . to as large an audience of developers as possible . . . and to eliminate as many obstacles for contribution and reuse of the code as possible.” *Id.*

146. *Netscape Public License FAQ*, *supra* note 116.

147. *See id.*

make open source extensions or additions to proprietary products.¹⁴⁸

1. MPL

Similar to the GPL, the MPL requires that licensees make modified or new files freely and publicly available in source code form.¹⁴⁹ Unlike the GPL, however, one interesting aspect of the MPL is that it allows developers to distribute their own files with “covered files” (any software or code that is covered by the MPL) under any licensing terms, provided that the developer has not modified the actual covered files.¹⁵⁰ Thus, the licensee need not make the source code for a proprietary work available even if the work contains unmodified MPL’d code.¹⁵¹ However, if the licensee has modified the covered files, then the developer must distribute the derived work under the open source MPL terms.¹⁵² Modifications to the software include: (1) changing anything within one of the files contained in the source code, (2) placing excerpts of source code from one of the files into a new file, or (3) renaming a file or combining two or more files contained in the source code.¹⁵³ However, adding a new file that does not contain any of the original source code or subsequent modified code is not a modification and does not fall under the terms of the MPL.¹⁵⁴ This is true even if the new file is referenced by changes made in the source code, although those changes constitute a modification.¹⁵⁵ However, the licensee must make changes to the MPL-covered source code freely available.¹⁵⁶

Additionally, the MPL contains several terms that protect the project and its developers against patent issues surrounding code that has been contributed to the project.¹⁵⁷ It basically

148. *See, e.g.,* Seltzer, *supra* note 110.

149. *See Mozilla Public License, supra* note 91, § 3.2.

150. *See id.* §§ 3.1-3.6.

151. *See id.*; Hackvån, *supra* note 145.

152. *See Mozilla Public License, supra* note 91, § 3.1.

153. *See id.* § 1.9; Hecker, *supra* note 3; *Netscape Public License FAQ, supra* note 116.

154. *See* Hecker, *supra* note 3; *Netscape Public License FAQ, supra* note 116.

155. *See Netscape Public License FAQ, supra* note 116.

156. *See Mozilla Public License, supra* note 91, § 3.2; *Netscape Public License FAQ, supra* note 116.

157. *See* Behlendorf, *supra* note 134.

requires that all contributors release all “patent rights that may be exposed by the code.”¹⁵⁸ This prevents people from contributing patented code to the project and then later attempting to collect patent fees for the use of the code.¹⁵⁹ Note, however, that this does not prevent anyone from contributing code for which someone else owns a patent.¹⁶⁰ One commentator has criticized this waiver clause because it requires the waiver of patent claims to the entire Mozilla code, not just to the specific code that has been contributed by the developer.¹⁶¹

Another interesting addition to this license is the legal.txt file that must accompany all distributions.¹⁶² The text file covers several issues, including liability, arbitration, and included code under dispute, under a patent, or under another limiting license.¹⁶³ Netscape decided that these issues should be clearly disclosed so that each developer can understand the issues surrounding the code to the greatest extent possible.¹⁶⁴

2. NPL

The NPL, a variation of the MPL, is used specifically with the Netscape Communicator source code.¹⁶⁵ Netscape needed a separate license for the Communicator source code to protect existing contracts still in force that govern shared source code with other proprietary products, and that govern the licensing of source code to third parties.¹⁶⁶

158. *Id.*; see *Mozilla Public License*, *supra* note 91, § 2.

159. See Behlendorf, *supra* note 134.

160. See *id.*

161. See *id.* Suppose, for example, that a company wishes to contribute code to the Mozilla project. It owns several patents on processes implemented by the Mozilla code for which it receives large amounts of money from other companies for the permission to use these processes. Once this company has agreed to the terms of the MPL and has waived patent claims over the entire Mozilla code, anyone, including the companies who license the use of the patented processes, can now take the Mozilla code (which is open source) and implement it in his products without paying the company any fees to license the patent. See *id.*

162. See *Mozilla Public License*, *supra* note 91, § 3.4.

163. See *id.*

164. See Behlendorf, *supra* note 134.

165. See *Netscape Public License Version 1.1*, at <http://www.mozilla.org/MPL/NPL-1.1.html> (last visited Nov. 7, 2000) [hereinafter *Netscape Public License*].

166. See Hecker, *supra* note 3.

Thus, the NPL comprises the terms of the MPL but grants Netscape two additional rights.¹⁶⁷ First, the license grants Netscape the right to use code licensed under the NPL in other products without having those products fall under the NPL.¹⁶⁸ Second, the license grants Netscape the right to re-license source code that falls under the NPL, including any additions made by non-Netscape developers, to third parties under terms other than those in the NPL.¹⁶⁹

E. Sun's Community Source License (SCSL)

Sun has distinguished its license from other open source licenses by calling it a "community source" license.¹⁷⁰ The community source development model differs from the open source development model in two significant ways. First, Sun requires and enforces compatibility among released versions of the software, and second, Sun allows proprietary modifications and extensions to the software.¹⁷¹ Sun created community source development principles to blend the best aspects of the proprietary and open source license models.¹⁷² Its primary goal in implementing this model is to balance the need of an organization to innovate rapidly while maintaining proprietary advantages.¹⁷³

Sun recognizes several advantages in using this model. First, it protects a developer's intellectual property rights in his software.¹⁷⁴ While a developer must give "error corrections . . . back to the [Sun] community, [his] other modifications can remain proprietary . . ." ¹⁷⁵ Further, unlike under a pure open source model, a single organization is responsible for the original code base.¹⁷⁶ Therefore, the "infrastructure . . . of the

167. *See id.*

168. *See id.*; *Netscape Public License*, *supra* note 165, at Amendments § V.

169. *See* Hecker, *supra* note 3; *Netscape Public License*, *supra* note 165, at Amendments § V.

170. *See generally* Gabriel & Joy, *supra* note 20.

171. *See id.*; *Licensing Overview*, at <http://www.sun.com/jini/licensing/overview.html> (last visited Nov. 7, 2000).

172. *See* Gabriel & Joy, *supra* note 20; *Licensing Overview*, *supra* note 171.

173. *See Licensing Overview*, *supra* note 171.

174. *See id.*

175. Gabriel & Joy, *supra* note 20.

176. *See id.*

original code and the upgrades to it are owned by the developing organization” and are shared.¹⁷⁷ However, the community may contribute its modifications and grant rights to the community for use of the modifications.¹⁷⁸

By keeping control over the source code, Sun ensures compatibility among the varying versions of software.¹⁷⁹ All internally distributed code must pass test suites.¹⁸⁰ The license requires commercial distributors “to use relatively recent upgraded code” and to pass the test suites.¹⁸¹ For extensions to the code, the license requires the community members to supply their specifications and test suites.¹⁸²

However, like the open source model, no central organization sets development schedules and priorities.¹⁸³ Further, innovation and releases are not subject to a schedule, though the code must first pass a conformance suite.¹⁸⁴

In order to implement this model, Sun has established two different levels of licenses: the Research Use license and the Commercial Use license.¹⁸⁵ The Research Use license is a click-wrap license available on the Internet, while the Commercial Use license must actually be signed and executed by both the developer and Sun.¹⁸⁶

The actual licensing terms are likely to vary for each technology or product.¹⁸⁷ Therefore, for the purpose of this Note,

177. *Id.*

178. *See id.*

179. *See id.*

180. *See id.*

181. *Id.*

182. *See id.*

183. *See id.*

184. *See id.*

185. *See id.* Sun previously had implemented a third licensing level called the “Internal Deployment” license. *See Jini™ Network Technology FAQs*, at <http://www.sun.com/jini/faqs/index.html> (last visited Nov. 7, 2000). However, because the contractual requirements for the Internal Deployment license and the Commercial Use license were similar, Sun decided to collapse the two licensing levels into one. *See id.* Those developers still operating under the Internal Use license must accept the terms of the new SCSL and sign and return a copy of the Commercial Use Supplement and its Technology Specific Attachment. *See id.*

186. *See Sun Community Source Licensing Frequently Asked Questions*, at <http://www.sun.com/software/communitysource/faq.html> (last visited Nov. 7, 2000).

187. *See Mike Loukides, Some Thoughts on the Sun Community Source License*, O’Reilly Java Center, at http://www.java.oreilly.com/news/loukides_0399.html (last modified Mar. 1999) [hereinafter Loukides, *Some Thoughts*] (“[T]he SCSL is more of a

the following paragraphs will discuss generally the terms of the Sun Community Source License Version 3.0 JINI™ Technology Specific Attachment v. 1.0 (hereinafter “Jini™ license”).¹⁸⁸

The Research Use license primarily grants all licensees the royalty-free rights to use, reproduce, and modify the original and upgraded source code for research; to distribute copies to other licensees and students; and to use test kits to develop and test code.¹⁸⁹ Error corrections must be returned to the community,¹⁹⁰ must carry notices that the code is subject to the terms of SCSL, and must be distributed with a file that documents the changes made.¹⁹¹ Other modifications, such as performance enhancements, adaptations, and extensions, may remain proprietary.¹⁹² However, interfaces to these modifications must be open and specified.¹⁹³

Once the technology has been evaluated and adopted by the community under the Research Use license, the code can be released internally (within the licensee’s business or organization) or externally under the Commercial Use license.¹⁹⁴ The Commercial Use license grants the licensee the right to reproduce and distribute compliant code that passes certain test suites and conforms to certain specifications.¹⁹⁵ Depending on

template than the license itself. There will be separate Java and Jini instantiations of the licenses; and there are differences between the two.” Mr. Loukides has metaphorically described the SCSL as “a chameleon.” See Mike Loukides, *The Chameleon and the Virus: More Thoughts on Java’s Community License*, O’Reilly Java Center, at http://java.oreilly.com/news/java_license_0399.html (last modified Mar. 1999) [hereinafter Loukides, *Chameleon*].

The license essentially changes its color based on the kind of software in which the Java source code is included. If it’s free Open Source software, it’s like an Open Source license in that it allows source code redistribution without restrictions or fees. If it’s commercial software, it’s more like a commercial license, with fees payable to Sun.

Id.

188. See generally *Sun Community Source License Version 3.0 Jini™ Technology Specific Attachment v. 1.0*, at http://www.sun.com/jini/licensing/SCSL3_JiniTSA1.html (last visited Nov. 7, 2000) [hereinafter *Jini™ License*].

189. See *id.* § III; *Licensing Overview*, *supra* note 171.

190. See *Jini™ License*, *supra* note 188, § IV; *Licensing Overview*, *supra* note 171. The community consists of any development organization or individual that has agreed to the SCSL. See *Jini™ License*, *supra* note 188, § I.

191. See *Jini™ License*, *supra* note 188, § IV; see also Gabriel & Joy, *supra* note 20.

192. See *Jini™ License*, *supra* note 188, § IV; see also Gabriel & Joy, *supra* note 20.

193. See *Jini™ License*, *supra* note 188, § IV; see also Gabriel & Joy, *supra* note 20.

194. See *Jini™ License*, *supra* note 188, Commercial Use Supp. § III (f)-(i).

195. See *id.*; *Licensing Overview*, *supra* note 171.

the technology licensed, some models require that royalties, trademark fees, or both, be paid to Sun under the Commercial Use license.¹⁹⁶

Before commercial distribution, the license must be signed and executed by Sun and the licensee, and the product must pass certain compatibility tests.¹⁹⁷ In some cases, products distributed under this license must carry the appropriate Sun logo, requiring the execution of a separate trademark licensing agreement with Sun.¹⁹⁸

Similar to the open source licenses discussed above, the SCSL provides a disclaimer of warranties and a limitation of liability clause.¹⁹⁹ Further, it also includes terms that prohibit the licensee from challenging Sun's ownership or use of Sun trademarks, from attempting to register any Sun trademarks, and from incorporating any Sun trademarks into his own trademarks, product names, service marks, company names, or domain names.²⁰⁰

Sun's implementation of its community source model has received mixed reviews from the open source community.²⁰¹ However, the license provides an interesting example of how commercial entities can take advantage of the benefits of developing under an open source model while still maintaining proprietary advantages.²⁰²

IV. SELECTING A LICENSE

A developer should carefully review licensing terms before selecting a license to govern his code or before contributing code to an open source project.²⁰³ At a minimum, a license

196. See *Licensing Overview*, *supra* note 171; *Sun Community Source Licensing Frequently Asked Questions*, *supra* note 188.

197. See *Licensing Overview*, *supra* note 171.

198. See *id.*

199. See *Jini™ License*, *supra* note 188, § V.

200. See *id.*

201. See Loukides, *Chameleon*, *supra* note 187 (praising the SCSL as a "chameleon license . . . [that] can entice developers to the Open Source model"); Malcolm Maclachlan, *Jini: Start Of 'Semi-Open' Source Movement*, TechWeb, Tech. News, at <http://www.techweb.com/wire/story/TWB19990128S0001> (Jan. 28, 1999); Malcolm Maclachlan, *Sun's Open Source Move Gets Mixed Reviews*, TechWeb, Tech. News, at <http://www.techweb.com/wire/story/TWB19981211S0001> (Dec. 11, 1998).

202. See Loukides, *Some Thoughts*, *supra* note 187.

203. Developers should avoid drafting their own licenses. See Perens, *supra* note 92.

should include a disclaimer of warranties to protect open source contributors from liability.²⁰⁴ The absence of such a disclaimer would potentially discourage open source developers from contributing code because of the risk of liability.²⁰⁵ Further, in order to ensure that all derivations of the code will fall under the same licensing terms, the developer will need self-perpetuating licensing terms in the license.²⁰⁶ A developer must also ensure that the license is carefully drafted to grant him the rights to use, copy, modify, and distribute the original code, rights normally protected by copyright law.²⁰⁷ Then, the developer should review the license to ensure that it reflects his open source development model.²⁰⁸ Questions regarding whether future contributors must give their modifications back to the open source community, whether fees can be charged for the distribution of the code, and whether copyright notices must be prominently displayed should guide the developer in making this decision and must be answered before the appropriate license can be selected.²⁰⁹

V. THE OPEN SOURCE MOVEMENT—TIPPING THE BALANCE IN FAVOR OF FREE USE?

The fundamental goal of the United States' intellectual property regime is to balance "free access to information . . . on the one hand, and the encouragement of investment in innovation through the exclusive rights of . . . copyright . . . laws on the other hand."²¹⁰ Nonetheless, copyright law effectively

"The propagation of many different and incompatible licenses works to the detriment of Open Source software because fragments of one program cannot be used in another program with an incompatible license." *Id.*

204. See Gomulkiewicz, *supra* note 8, at 191; see also Perens, *supra* note 92.

205. Developers are "unwilling to assume the risk of a multi-million dollar class action lawsuit as the consequence of pursuing their passion for hacking code." Gomulkiewicz, *supra* note 8, at 191-92 (citations omitted).

206. "The open source movement could not operate without non-negotiated, standard-form, take-it-or-leave-it mass-market licenses." *Id.* at 190 (citations omitted).

207. See Loukides, *Some Thoughts*, *supra* note 187.

208. See Gomulkiewicz, *supra* note 8, at 191; see also Perens, *supra* note 92.

209. See Perens, *supra* note 92.

210. Mark A. Haynes, *Commentary: Black Holes of Innovation in the Software Arts*, 14 BERKELEY TECH. L.J. 567, 567 (1999). Society believes that without copyright protection, "the low cost of copying such works will induce competitors to enter and 'steal' another's product without penalty." Steve P. Calandrillo, *An Economic Analysis*

gives authors a monopoly in their work.²¹¹ However, some scholars have reasoned that society would still develop socially useful works in the absence of financial incentives.²¹² They argue that “simple personal satisfaction, the quest for respect and esteem, and the power of convention” also induce society to create.²¹³ Critics also question whether awarding rights in a work for the life of the author plus seventy years provides any additional incentive to create.²¹⁴

The Open Source Movement appears to dispel the need for such financial incentives to encourage creativity.²¹⁵ By making the source code available to anyone to use, modify, enhance, and redistribute, the movement is effectively encouraging productivity.²¹⁶ Developers from around the world collaborate to create a more reliable, robust product.²¹⁷ Moreover, open source software development is more in tune with the needs of society.²¹⁸ Most importantly, anyone can reuse code to create additional beneficial products.²¹⁹

of Property Rights in Information: Justifications and Problems of Exclusive Rights, Incentives To Generate Information, and the Alternative of a Government-Run Reward System, 9 FORDHAM INTELL. PROP. MEDIA & ENT. L.J. 301, 303 (1998). Thus, “innovators will have little hope of recovering their investment, [and] the production of information will be seriously curtailed . . .” *Id.*

211. See Calandrillo, *supra* note 210, at 304.

212. See *id.* at 305. In his article, Mr. Calandrillo states that “the desire for personal satisfaction, respect and esteem may motivate creators to produce,” even without financial incentive. *Id.* at 316-17. However, he later reasons that software development “depends significantly upon pecuniary inducements” because the costs of development are “often enormous, while the cost of duplicating the end result is virtually zero.” *Id.* at 325.

213. *Id.* at 305 (citations omitted).

214. *Id.* at 334-35. Current copyright law protects the work for the life of the author plus seventy years. See 17 U.S.C. § 304 (1994).

215. “[I]t would be shortsighted of those in the computer industry to believe that monetary reward is the primary concern of the Open Source’s best programmers.” DiBona et al., *supra* note 19.

216. For example, when the “Ping o’ Death’ exploit was revealed in 1997 . . . Linux had fix patches within hours. Closed-source [operating systems] didn’t plug the hole for months.” *Frequently Asked Questions About Open Source*, at <http://www.opensource.org/faq.html> (last visited Nov. 7, 2000).

217. See Longmire, *supra* note 1.

218. See *id.*

219. See *id.* “[I]ndustry needs the innovation science can provide . . . Open source development drives progress not just in computer science, but in the computer industry as well.” DiBona et al., *supra* note 19.

Nonetheless, the Open Source Movement must rely on copyright protection and software licensing to effect its development model.²²⁰ Without protection, developers would not have control over the future free use and dissemination of their code.²²¹

CONCLUSION

Though many theories exist on how “open” open source software should be, one thing is certain: copyright protection and software licensing are essential in implementing any open source development model.²²² Developers and commercial entities should thoroughly review licensing terms before contributing to any open source project or selecting an open source license for their work.²²³ These terms will dictate, for example, to what extent licensees are encouraged or required to return enhancements and modifications to the open source community, to what extent derived works are governed by the license, whether prior contributors must be acknowledged in works using their code, and whether open source code may be combined or distributed with proprietary software.²²⁴ Further, depending on its terms, an open source license can be an effective means of taking advantage of the benefits of developing under an open source model, while at the same time attending to commercial business needs (such as the development of proprietary products in addition to open source products) or retaining certain rights in the code.²²⁵

Natasha T. Horne

220. See Anthony J. Mahajan, *Intellectual Property, Contracts, and Reverse Engineering After ProCD: A Proposed Compromise for Computer Software*, 67 *FORDHAM L. REV.* 3297, 3328 (1999) (stating that “public access to source code is . . . commercially desirable [so long as] owners can retain their . . . rights . . . in the . . . code”). “Software that is placed in the public domain can be snapped up and put into non-free programs, and be free no more. To stay free, software must be copyrighted and licensed.” Debian GNU/Linux, *supra* note 79.

221. See Gomulkiewicz, *supra* note 8, at 187-88.

222. See *supra* Parts I-II.

223. See *supra* Part IV.

224. See *supra* Part II.D.

225. See *supra* Part III.D-E.

