The Trademark Reporter®

International Trademark Association

The Law Journal of the International Trademark Association

July–August, 2012 Vol. 102 No. 4

THE UNEASY ROLE OF TRADEMARKS IN FREE AND OPEN SOURCE SOFTWARE: YOU CAN SHARE MY CODE, BUT YOU CAN'T SHARE MY BRAND*

*By Pamela S. Chestek***

INTRODUCTION

At first blush, a trademark for free and open source software is no different from any other trademark. A trademark for open source software can be misused like any other trademark and enforcement against infringers is a necessary task.[1] What is different, however, is the reaction of those against whom enforcement might be sought. Because of the collaborative nature of much open source software development and its culture of sharing, efforts to restrict how a trademark may be used can meet with resistance, sometimes even challenged on the basis that trademark enforcement is contrary to the open source license under which the software is distributed.[2] This article will discuss enforcement of free and open source software trademarks, examining both open source copyright licenses and trademark law principles.

In furtherance of the collaborative nature of open source code development and use, however, a trademark owner for an open source software project may also want others to use its mark. This article will also discuss trademark licensing law principles including when the sharing goes too far, putting trademark rights at risk of loss. Part I of the article will give a short background on the open source development model, both technical and community aspects. Part II of the article will review the legal aspects of trademark use for FOSS projects.

** Pamela S. Chestek is Senior IP Attorney for Red Hat, Inc. and is trademark counsel for the Red Hat, Fedora and JBoss brands, all open source software brands. The author would like to thank Tom Callaway, Paul Frields, Richard Fontana and Jared Smith for their assistance with the article, particularly the description of the technical aspects of software. Any errors, however, are entirely those of the author. All opinions expressed in this article are those of the author and not her employer.

1. Hja's Blog, Trademarks—the Good, the Bad and the Ugly, http://lockshot.wordpress.com/2009/04/20/trademarks-%E2%80%93-the-good-the-bad-and-the-ugly/ (April 20, 2009).

2. *See* note 24, *infra*, and accompanying text.

I. THE OPEN SOURCE DEVELOPMENT MODEL

In broad terms, free and open source software (FOSS) is software where the copyright in the code is licensed in a way that allows fairly unfettered use, reproduction and modification.[3] A necessary predicate to implementing these principles is that the "source code," the language in which software programs are written, is made available for others to review and improve upon, in contrast to the typical practice of proprietary software companies which protect their source code as trade secrets.[4] But free and open source software is much more than simply a matter of whether the source code is made publicly available or not; rather, it is a philosophy about the nature of software, its development, and the role of law—copyright in particular—in advancing the interests of the free software community. Section A of Part I will talk about the mechanics of turning source code into a product, a program that runs on a computer, and Section B will give some brief background on the open source development community.

A. The Technical Aspects of Creating Open Source Software

Free and open source software programs, like most software programs, originate as "source code." Source code is the human-readable form of a computer program,[5] typically written in high-

3. Some call the software "open source," others "free software." *See, e.g.,* Richard Stallman, Why "Free Software" is Better than "Open Source," http://www.gnu.org/philosophy/free-software-for-freedom.html (last visited Nov. 8, 2011). Some also refer to the software as "FLOSS," standing for "free/libre open source," to clarify that "free" in the software context means free as in freedom, not free as in without cost. For purposes of this article, whether described as "open source," "free" or "free and open source" software, in all cases it refers to software where permission is granted to reproduce, modify, and distribute the software and derivative works thereof.

This article will not delve into the various types of licenses that are considered "open source" or "free software" licenses. For more information on licensing, resources are Lawrence Rosen, Open Source Licensing Software Freedom and Intellectual Property Law (2004); Open Source Initiative, Open Source Licenses, http://www.opensource.org/licenses (last visited Nov. 8, 2011); and Free Software Foundation Various Licenses and Comments About Them, http://www.gnu.org/licenses/license-list.html (last visited Nov. 8, 2011).

4. See Free Software Foundation GNU Operating System, The Free Software Definition rev. 1.92, http://www.gnu.org/philosophy/free-sw.html (last visited Nov. 8, 2011) (stating that part of the definition of free software is that one have access to the source code); Open Source Initiative, The Open Source Definition (Annotated) Version 1.9, http://www.opensource.org/docs/definition.php (last visited Nov. 8, 2011) (including in the definition of open source software a requirement that source code must be included).

5. Denis Howe, The Free On-line Dictionary of Computing, http://dictionary.reference.com/browse/source code (last visited Nov. 8, 2011).

level language like C++, Java or Perl.[6] However, a computer cannot directly run the high-level language of source code, so for the computer to perform the functions described in the program the source code must be converted to "object" or "executable" code.[7] This act is performed by a "compiler."[8] Typically when purchasing non-open source software, one gets only a copy of the executable code, not source code.[9] The executable code is the program the computer actually runs. In commercial terms, source code is perhaps more properly understood as the instructions for creating a functional product, not the product itself.[10]

Compiling source code does not necessarily generate identical object code files. The version of a compiler and the "target" hardware for the compilation software will affect the final executable code.[11] Further, a compiler has a number of variables

---

6. Rackspace Cloud Blog, [INFOGRAPHIC] Evolution of Computer Languages, http://www.rackspace.com/cloud/blog/2011/05/17/infographic-evolution-of-computer-languages/ (May 17, 2011).

7. This is a simplified explanation for how computers perform instructions. There may be different levels of translation and some languages are "interpreted" rather than compiled into object code. *See, e.g.*, Linuxdoc.org, The Unix and Internet Fundamentals HOWTO, How Do Computer Languages Work?, http://www.linuxdoc.org/HOWTO/Unix-and-Internet-Fundamentals-HOWTO/languages.html (last visited Nov. 8, 2011); Ed Felton, Freedom to Tinker, *Source Code and Object Code*, https://freedom-to-tinker.com/blog/felten/source-code-and-object-code (Sept. 4, 2002); David S. Touretzky, *Source vs. Object Code: A False Dichotomy*, http://www-2.cs.cmu.edu/~dst/DeCSS/object-code.txt (July 12, 2000). Dr. Touretzky was an expert witness in Universal City Studios, Inc. v. Reimerdes, Civ. No. 00-9185 (S.D.N.Y.). The transcript of his testimony referencing his essay and discussing the compilation of code is available at http://cyber.law.harvard.edu/openlaw/DVD/NY/trial/0725.html, pp. 1086-1089 (last visited Nov. 8, 2011). This article discusses only the case where source code has been compiled into executable code, not interpreted code.

8. David S. Touretzky, *Source vs. Object Code: A False Dichotomy*, http://www-2.cs.cmu.edu/~dst/DeCSS/object-code.txt (July 12, 2000).

9. Karl Fogel, *Producing Open Source Software* 5-6 (2006), *available at* http://producingoss.com [hereinafter *Producing OSS*] (describing the evolution of the portability of software, which led to the practice of denying access to source code); Chris DiBona, Sam Ockman & Mark Stone, Open Source, Voices From the Revolution 54 (1999), *available at* http://oreilly.com/catalog/opensources/book/stallman.html (last visited Nov. 8, 2011); Copyright Office Circular 61, Copyright Registration for Computer Programs (May 2011) (permitting registration of object code instead of source code in order to protect trade secrets in source code).

10. *C.f.*, *e.g.*, Free Software Foundation GNU Operating System, The Free Software Definition rev. 1.92, http://www.gnu.org/philosophy/free-sw.html (last visited Nov. 8, 2011) (describing one of the four freedoms the freedom to change how a program works by giving access to the source code); GNU General Public License, version 2, http://www.gnu.org/licenses/gpl-2.0.html (June 1991) (duty to provide source code arises at distribution of object or executable code); Sun Microsystems, Inc. Trademark License, http://contracts.onecle.com/opentv/sun.tm.1998.03.20.shtml (March 20, 1998) (allowing use of "Java" trademarks only on "Product," which must be executable code because it must pass the Java Test Suites, which are tests run on executable code.).

11. Email from Paul Frields, former Fedora Project Leader, to the author (Aug. 22, 2011, 16:00:34 -0400 GMT) (on file with author); email from Richard Fontana, former legal counsel at the Software Freedom Law Center, to the author (Sept. 8, 2011, 11:39 AM) (on file with author).

that may also affect the object code, such as options,[12] switches,[13] and environment variables.[14] One creating an executable file from source code may also include selected changes to the source code that differ from the official repository when compiling, in an effort to provide subsequent bug fixes or improvements.[15]

It is common for projects or organizations that provide installable executable code packages to digitally sign those packages.[16] These signatures, when authenticated upon the installation of the program, are an assurance that the digital data came from the trusted source and has not been tampered with or altered.[17] FOSS has a risk not found with code that is distributed only in executable form: someone can take the freely available

---

12. *See, e.g.*, Free Software Foundation, GCC 4.6.1 Manual, 9 Binary Compatibility, http://gcc.gnu.org/onlinedocs/gcc-4.6.1/gcc/Compatibility.html#Compatibility (last visited Nov. 8, 2011) ("Some GCC compilation options cause the compiler to generate code that does not conform to the platform's default ABI. Other options cause different program behavior for implementation-defined features that are not covered by an ABI. These options are provided for consistency with other compilers that do not follow the platform's default ABI or the usual behavior of implementation-defined features for the platform. Be very careful about using such options."); Free Software Foundation, GCC 4.6.1 Manual, 3.1 Option Summary, http://gcc.gnu.org/onlinedocs/gcc-4.6.1/gcc/Option-Summary.html#Option-Summary (last visited Nov. 8, 2011).

13. *See, e.g.*, Free Software Foundation, GCC 4.6.1 Manual, 3.15 Specifying Subprocesses and the Switches to Pass to Them, http://gcc.gnu.org/onlinedocs/gcc-4.6.1/gcc/Spec-Files.html#Spec-Files (last visited Nov. 8, 2011).

14. *See, e.g.*, Free Software Foundation, GCC 4.6.1 Manual, 3.19 Environment Variables Affecting GCC, http://gcc.gnu.org/onlinedocs/gcc-4.6.1/gcc/Environment-Variables.html#Environment-Variables (last visited Nov. 8, 2011).

15. *Producing OSS*, at 184.

16. *See, e.g.*, Red Hat Enterprise Linux 6 Security Guide, Verifying Signed Packages, http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/sect-Security_Guide-Updating_Packages-Verifying_Signed_Packages.html (last visited Nov. 8, 2011) ("All Red Hat Enterprise Linux packages are signed with the Red Hat GPG key."); Debian on CDs, Verifying Authenticity of Debian CDs, http://www.debian.org/CD/verify (last visited Nov. 8, 2011) ("Official releases of Debian CDs come with signed checksum files"); Android Developers, Signing Your Applications, http://developer.android.com/guide/publishing/app-signing.html (last visited Nov. 8, 2011) ("The Android system requires that all installed applications be digitally signed with a certificate whose private key is held by the application's developer."); *but see* IgnorantGuru's Blog, Arch's Dirty Little Not-So-Secret, http://igurublog.wordpress.com/2011/02/19/archs-dirty-little-notso-secret/ (Feb. 19, 2011) (criticizing Arch Linux for not signing packages).

17. *See, e.g.*, Debian on CDs, Verifying Authenticity of Debian CDs, http://www.debian.org/CD/verify (last visited Nov. 8, 2011) (stating that signing ensures that the files have not been tampered with); Red Hat Enterprise Linux 6 Security Guide, Verifying Signed Packages, http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Security_Guide/sect-Security_Guide-Updating_Packages-Verifying_Signed_Packages.html (last visited Nov. 8, 2011) (describing the signing process and that it is used to ensure trustworthiness). This is also common practice for proprietary software distribution. *See, e.g.*, Oracle, Application Packaging Developer's Guide, http://download.oracle.com/docs/cd/E19082-01/817-0406/ch6advtech-108/index.html (last visited Nov. 8, 2011) (stating that signature ensures package has not been modified); Microsoft | Technet, Windows 2000 Server, Digitally Signed Software, http://technet.microsoft.com/en-us/library/cc962053.aspx (last visited Nov. 8, 2011) (suggesting developers sign software to ensure its integrity).

source code and when, distributing it in binary format, introduce malicious code or include hidden malware.[18] Thus, knowing that the source of a binary file is trustworthy is an important characteristic of the software.

B. The Community Aspects of Open Source

FOSS, in its most normative case, is created through an iterative, collaborative, and community-based process—a "project"—by people who simply have an interest in what the software does.[19] Free and open source software would not exist in its present form without the Internet, which allows for the cooperation of people all around the world. The participants in an open source project use collaborative public tools like source code repositories, mailing lists, real time chatting, bug trackers, and wikis to cooperatively develop and modify not only software,[20] but all facets of the project, like documentation, artwork, translations, and marketing.[21]

The nature of the open source community development model fosters a feeling of commitment and participation.[22] As individuals participate, they get invested in the work and begin to feel a sense of ownership about the project, including about their perceived right to use the trademark associated with it.[23]

---

18. Brian Proffitt, *FLOSS: Accept No Substitutes*, ITWorld Beta, http://www.itworld.com/security/182757/floss-accept-no-substitutes (July 13, 2011) (reporting on sites that wrongfully claim ownership of VideoLAN software and add installers that install malware); Etix's Weblog, These Companies Mislead Our Users, http://blog.l0cal.com/2011/07/07/these-companies-that-mislead-our-users/ (July 7, 2011) (VideoLAN listing companies distributing VLC Media Player with malware).

19. *Producing OSS* at xiv.

20. *Id.* at 45-86 (describing technical requirements for setting up an open source project).

21. The Fedora Project at http://www.fedoraproject.org (for which the author is trademark counsel) is an example of a robust open source community project that has subprojects for design, marketing, documentation, internationalization, localization, the website, and many other functions.

22. *See generally* The Open Source Way, Creating and Nurturing Communities of Contributors, http://www.theopensourceway.org/book/ (last visited Nov. 8, 2011) ("This is the group of people who form intentionally and spontaneously around something important to them. It includes the people who use or benefit from the project, those who participate and share the project to wider audiences, and the contributors who are essential to growth and survival.").

23. *See, e.g.*, Linux-to-go.org nils' blog, Trademark Issue, June 7, 2007, http://www.linuxtogo.org/trademark-issue (upon learning that another individual was registering the open source project trademark "GPE," commenting "Do you know that feeling when you feel treated unjust? When someone has done wrong to you and you simply cannot do anything against it? This is exactly what I am feeling again and again these days ever since I heard about the trademark registrations of handhelds.org especially done by George France.").

II. THE LEGAL PRINCIPLES IN PLAY

Because of the communal nature of an open source project, trademark owners of FOSS projects sometimes take heat for trying to restrict who may use a trademark and how.[24] Meanwhile, open source project participants may have only a limited understanding of the different roles of trademarks and copyright, assume that trademarks can be licensed under sharing principles in the same way that copyrights can, and resist efforts to restrict the use of the trademark.[25] Further, given the community aspect of FOSS projects, there may be a significant benefit in allowing participants to use the trademark fairly freely. This leaves the trademark owner in a quandary: how much control can, and how much control does the owner of an open source project trademark have to, exercise?

Part A of this section will discuss what kind of limitations the trademark owner of a FOSS project trademark can put on the use of its trademarks by others, considering both limitations in the copyright licenses and trademark law principles. Part B will discuss to what extent the trademark owner may license its mark, including when the license may have gone too far, exposing the trademark to a validity challenge.

---

24. One of the more notorious conflicts was the Debian distribution's use of the mark FIREFOX for a web browser it had altered. *See* bug report string at Debian Bug Report Logs - #354622, Uses Mozilla Firefox Trademark Without Permission, http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=354622 (last visited Nov. 8, 2011) for a lengthy and detailed discussion of the dispute. *See also* bug report string at Debian Report Logs - #258918, Abiword: Debian Appears to be Violating AbiWord's License, http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=258918 (last visited Nov. 8, 2011).

25. *See, e.g.*, Timothy MacIntyre, Re: [Fwd: FW: For Approval: Generic Attribution Provision], License-Discuss mailing list of Open Source Initiative, http://www.crynwr.com/cgi-bin/ezmlm-cgi?3:msp:12027:oiccemjkkoffgnlmoebm (Dec. 13, 2006, 12:48:57 -800 GMT) (stating that his project solved the problem of granting a trademark license by granting a license to the *copyright* in the phrase "Powered by Terracotta." "We don't require any redistributions to include the Terracotta logo (i.e., trademark), and instead just ask for inclusion of text."); Debian Bug Report Logs - #354622, Uses Mozilla Firefox Trademark Without Permission, msg. 364, http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=354622 (last visited Nov. 8, 2011) (claiming that permitting modification of the logo does not affect enforceability of the trademark; note the author appears to believe that only words can be trademarks). Brazil has formalized this position, offering an "Open Trademark License" that provides that others may use the trademark as long as they do not suggest that the brand owner endorses the product. Portal do Software Público Brasileiro: Licença Pública de Marca, http://www.softwarepublico.gov.br/lpm (last visited Nov. 8, 2011).

A. Limitations on Trademark Enforcement of Open Source Project Trademarks

1. Trademark Enforcement and FOSS Copyright Licenses

The fundamental principle of free and open source software is that the code may be modified and those modifications redistributed. The question arises whether prohibiting use of the trademark is contrary to those principles. There are two organizations that are generally considered arbiters of free and open source licensing principles, the Open Source Initiative (OSI) and the Free Software Foundation.[26] The Open Source Initiative has propounded its definition of "open source" software, called the "Open Source Definition" (OSD),[27] and the Free Software Foundation has its "Free Software Definition."[28]

Some open source software community participants consider the restriction of use of a trademark inconsistent with the Open Source Initiative's OSD.[29] The argument is that the OSD states that open source licenses may not be specific to a type of use or product,[30] and that this extends to the trademark,[31] or at the least the logo form of a mark.[32] If this standard applies to a trademark, however—that anyone may use the trademark completely unmoored from an association with a particular product—it is simply not a trademark:

---

26. Alan Hicks, et al., Slackware Linux Essentials, 2d edition (FreeBSD Mall 2005), Chapter 1.3, *available at* http://www.slackbook.org/html/introduction-opensource.html.

27. Open Source Initiative, The Open Source Definition (Annotated) Version 1.9, http://www.opensource.org/docs/definition.php (last visited Nov. 8, 2011).

28. Free Software Foundation GNU Operating System, The Free Software Definition rev. 1.92, http://www.gnu.org/philosophy/free-sw.html (last visited Nov. 8, 2011).

29. *See* Posting of Brandon Robinson, AbiWord, trademarks and DFSG-freeness, http://lists.debian.org/debian-legal/2004/10/msg00236.html (Oct. 15, 2004, 02:12:41 -500 GMT) (outlining perceived inconsistency between AbiWord's trademark policy and the Open Source Definition).

30. See Open Source Initiative, The Open Source Definition (Annotated) Version 1.9, http://www.opensource.org/docs/definition.php (last visited Nov. 8, 2011) ("License Must Not Be Specific to a Product. The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.").

31. Debian Bug Report Logs - #258918, abiword: Debian Appears to Be Violating AbiWord's License, msg 20, http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=258918 (last visited Nov. 8, 2011) (proposing language for granting a license to downstream users).

32. Debian Bug Report Logs - #354622, Uses Mozilla Firefox Trademark Without Permission, msg. 25, http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=354622 (last visited Nov. 8, 2011) ("I had to break the switch, because I need to call it Firefox, but I can't include the official graphics . . . [b]ecause it uses graphics which have a non-free copyright license.").

> [T]here is no property whatever in a trademark, as such. . . . [A] man . . . has no property in that mark per se, any more than in any other fanciful denomination he may assume for his own private use, otherwise than with reference to his trade. If he does not carry on a trade in iron, but carries on a trade in linen, and stamps a lion on his linen, another person may stamp a lion on iron. . . . In short, the trademark is treated as merely a protection for the good will, and not the subject of property except in connection with an existing business.[33]

Thus, despite some opinions to the contrary, the Open Source Initiative principles are meaningless in the context of trademark law and therefore cannot be stating a trademark licensing principle, other than perhaps one may not have trademarks at all for open source software.

Some also argue a restriction on the use of a trademark downstream is inconsistent with Section 6 of the GNU General Public License, version 2 ("GPLv2") propounded by the Free Software Foundation.[34] Section 6 of GPLv2 says that "[e]ach time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein."[35]

This, however, is not intended as a trademark license. The Free Software Foundation defines the principles that the GPL was intended to protect thusly:

> Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software. More precisely, it means that the program's users have the four essential freedoms:

---

33. Hanover Star Milling Co. v. Metcalf, 240 U.S. 403, 413-14, 36 S. Ct. 357, 360-61 (1916); *see also* American Steel Foundries v. Robertson, 269 U.S. 372, 380, 46 S. Ct. 160, 162 (1926) ("There is no property in a trade-mark apart from the business or trade in connection with which it is employed."). There are also FOSS proponents who understand this concept. *See, e.g.*, Gervase Markham, Re: Educational Community License 1.0, http://lists.debian.org/debian-legal/2011/08/msg00025.html (29 Aug. 2011 11:37:24 +0100 GMT) ("The purpose of trademarks in law is as a determinant of origin. If putting a trademark into Debian requires the trademark holder to allow software from any origin and with arbitrary changes to bear the trademark, then the trademark is no longer valid or defendable. And such a 'trademark' is no longer a trademark (literally: a mark used in trade)").

34. Posting of Bill Allombert, Re: AbiWord, Trademarks and DFSG-freeness, http://lists.debian.org/debian-legal/2004/10/msg00276.html (Oct. 18, 2004, 12:41:45 +0200 GMT) ("The trademark condition impose [sic] a restriction on modification in direct conflicts [sic] with GPL6.").

35. GNU General Public License, version 2, http://www.gnu.org/licenses/gpl-2.0.html (June 1991).

- The freedom to run the program, for any purpose (freedom 0).
- The freedom to study how the program works, and change it to make it do what you wish (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbor (freedom 2).
- The freedom to distribute copies of your modified versions to others (freedom 3). . . .[36]

This licensing philosophy is specifically directed at freedom under copyright law, not trademark law.[37] Note that these freedoms are all directed at improving the functionality of the software, not the freedom to represent the software as something that it might not be, e.g., the same as the code from which it was derived.

The Free Software Foundation does not find trademark rights inconsistent with free software. According to Richard Stallman,[38] if it is easy to find and remove the trademarks, restrictions on the re-use of trademarks is not inconsistent with the GPLv2.[39] A newer version of the GNU General Public License, version 3 ("GPLv3"), specifically allows a trademark owner to decline to grant rights for the use of trade names, trademarks, and service marks.[40] GPLv3 is seen largely as a clarification of GPLv2,[41] so the same holds true

36. Free Software Foundation GNU Operating System, The Free Software Definition rev. 1.92, http://www.gnu.org/philosophy/free-sw.html (last visited Nov. 8, 2011).

37. *See*, *e.g.*, Free Software Foundation GNU Operating System, What is Copyleft, http://www.gnu.org/copyleft/ (last visited Nov. 8, 2011) ("To copyleft a program, we first state that it is copyrighted; then we add distribution terms, which are a legal instrument that gives everyone the rights to use, modify, and redistribute the program's code, or any program derived from it, but only if the distribution terms are unchanged.")

38. Richard Stallman is the founder and president of the Free Software Foundation. Free Software Foundation, About, http://www.fsf.org/about/board (last visited Nov. 8, 2011).

39. Richard Stallman, [Savannah-hackers] Re: Issue of Trademark Logo Images in Source Distribution, http://lists.gnu.org/archive/html/savannah-hackers/2004-11/msg00508.html (18 Nov. 18, 2004 21:25:40 -0500 GMT) ("It is no problem if the program contains trademarked images and names, provided the trademark usage and requirements don't make it difficult in practice to change the program and publish a modified version. In other words, it has to be easy to find and remove the trademarks, if and when the trademark conditions require this.").

40. Free Software Foundation GNU General Public License Version 3, § 7, http://www.gnu.org/licenses/gpl.html (June 29, 2007) ("Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms: . . . Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks").

41. Free Software Foundation GNU Operating System, A Quick Guide to GPLv3 by Brett Smith, http://www.gnu.org/licenses/quick-guide-gplv3.html (last visited Nov. 8, 2011) ("GPLv3 isn't a radical new license; instead it's an evolution of the previous version. Though a lot of text has changed, much of it simply clarifies what GPLv2 said.").

for it. Thus, it should be beyond purview that FOSS licenses do not in any general sense grant a license in the trademark for the software.[42]

2. The Reach of Trademark Law in Enforcement of FOSS Project Trademarks

As a general principle, there is no right to use another's trademark.[43] One may use a trademark to sell the product, but the permitted use of the trademark extends only to stocking, displaying and reselling the product. This result has been variously reached on the basis of the "first sale" doctrine[44] or that such activities with respect to the mark do not constitute trademark use.[45]

As a result, in the case of both open source software and proprietary software, the mark may be used only to distribute the exact form of the software as provided by the project owner, namely executable code only in executable form and source code only in source code form.[46] It therefore would not extend to permit the use of the mark for a different product, e.g., executable code created by a third party from project source code.[47]

---

42. *See* Progress Software Corp. v. MySQL AB, 195 F. Supp. 2d 328, 329 (D. Mass. 2002) (holding that use of trademark for GPL-licensed MySQL software was unauthorized use), *see also* The Nagios Trademark Policy—Why It's Necessary, May 11, 2009, http://community.nagios.org/2009/05/11/the-nagios-trademark-policy-why-its-necessary/ ("Open Source licenses—at least the GPL—do not address trademarks at all."); Oracle, Free and Open Source Java FAQ, http://www.sun.com/software/opensource/java/faq.jsp#g26 (last visited June 13, 2011)("The GPL v2 does not include a trademark license - no OSI-approved open-source licenses do."); Linux Foundation Trademark Usage Guidelines, http://www.linuxfoundation.org/about/linux-foundation-trademark-usage-guidelines (last visited June 10, 2011) ("A copyright license, even an open source copyright license, does not include an implied right or license to use a trademark that may be related to the project or workgroup developing the licensed software or other materials"). The German Oberlandesgericht Düsseldorf has also held that the GPL does not grant a trademark license. Dr. Till Kreutzer, OLG Düsseldorf Entscheidet über das Verhältnis Zwischen GPL und Markenrecht, http://www.ifross.org/artikel/olg-duesseldorf-entscheidet-ueber-verhaeltnis-zwischen-gpl-und-markenrecht (Oct. 10, 2010).

43. A. Bourjois & Co., Inc. v. Katzel, 260 U.S. 689, 692, 43 S. Ct. 244, 245 (1923) ("Ownership of the goods does not carry the right to sell them with a specific mark.").

44. Sebastian Int'l Inc v. Longs Drugs Stores Corp., 53 F.3d 1073, 1076 (9th Cir. 1995) ("It is the essence of the 'first sale' doctrine that a purchaser who does no more than stock, display, and resell a producer's product under the producer's trademark violates no right conferred upon the producer by the Lanham Act").

45. Wilkof and Burkitt, Trade Mark Licensing, 2d edition (Sweet & Maxwell, 2005), para 7-09.

46. The Slackware project adheres to this standard. Slackware Trademark Policy, http://www.slackware.com/trademark/trademark.php (last visited Nov. 8, 2011) ("In order to be called 'Slackware', the distribution may not be altered from the way it appears on the central FTP site").

47. See notes 11-15, *supra*, and accompanying text explaining the differences there might be between different compilations of the same source code.

Further, there is also no argument that executable code compiled by a third party from project source code is equivalent to the project owner's own version of the executable code. There is no right to use a trademark even for new product if the product being sold has been altered, which makes it necessary to examine whether there are "material differences" between the trademark owner's version of the software and the third party's newly-compiled software.[48]

While the question has not been answered in the context of free and open source software, the issue of whether differences are "material" is frequently examined in "parallel import" or "gray market" situations. "Gray market" goods are generally thought of as foreign-manufactured goods sold in the US bearing a valid United States trademark without the consent of the trademark holder,[49] but the legal theory also applies to domestic goods.[50] Under the "quality control" line of cases applied in gray market goods situations, use of a trademark will be considered an infringement if the trademark owner's quality control measures are thwarted.[51] To show there are material differences, "[t]he trademark owner must demonstrate only that (i) it has established legitimate, substantial, and nonpretextual quality control procedures (ii) it abides by these procedures, and (iii) the non-conforming sales will diminish the value of the mark."[52]

The digital signature used for executable software packages[53] is the vehicle used for the quality control of open source software products, much like a UPC code.[54] The signature (i) has established legitimate, substantial, and nonpretextual quality control procedures, i.e., it is a legitimate practice used to confirm

48. Beltronics USA, Inc. v. Midwest Inventory Distrib., LLC, 562 F.3d 1067, 1072 (10th Cir. 2009) ("It logically follows that the first sale doctrine is not applicable "when an alleged infringer sells trademarked goods that are materially different than those sold by the trademark owner.").

49. Quality King Distrib., Inc. v. L'Anza Research Int'l, Inc., 523 U.S. 135, 153, 118 S. Ct. 1125, 1134 (1998).

50. Iberia Foods Corp. v. Romeo, 150 F.3d 298, 302 (3d Cir. 1998) (stating that the scope of the action is not limited to gray goods cases but may be used to enjoin the sale of domestic products in conditions materially different from those offered by the trademark owner), *citing* Warner-Lambert Co. v. Northside Dev. Corp., 86 F.3d 3 (2d Cir. 1996) (where the owner of the Halls cough drops trademark was entitled to an injunction against the sale of Halls cough drops past their expiration date); *see also Beltronics*, 562 F.3d at 1072, n.4 ("Thus, we agree with the Third and Eleventh Circuits that the rule 'is not limited to gray goods cases.'").

51. Zino Davidoff SA v. CVS Corp., 571 F.3d 238, 243 (2d Cir. 2009) ("[T]he interference with the trademark holder's legitimate steps to control quality unreasonably subjects the trademark owner to the risk of injury to the reputation of its mark.").

52. *Warner-Lambert Co.*, 86 F.3d at 6.

53. *See* notes 1618, *supra*, and accompanying text.

54. *See Zino Davidoff*, 571 F.3d at 243 (finding that UPC codes on perfume boxes were an adequate quality control measure).

that the software is authentic and of a quality that satisfies the trademark owner's standards.[55] Factor (ii) is satisfied for those FOSS distributions that require signing the software package as part of their standard practices.[56] Finally, under (iii), unsigned packages will diminish the value of the mark; non-conforming goods may not work as intended or may include malware.[57] Thus, a FOSS project is well within its rights to prevent use of the trademark for anything but its own signed files.

A subsequent distributor also cannot claim that its use of the project mark for its own executable file built from project source code is simply to describe the origin of its software. While the doctrine of nominative fair use allows "a defendant to use a plaintiff's trademark to identify the plaintiff's goods so long as there is no likelihood of confusion about the source of the defendant's product or the mark-holder's sponsorship or affiliation,"[58] the doctrine is inapplicable in this context. For the defense to apply, the trademark use must be in reference to the original product, not as a name for the infringing product.[59] A person who builds a new executable file from source code and labels it with the trademark is not using that trademark in a nominative manner to describe the original project's executable

---

55. *See* note 17, *supra*.

56. *See* note 16, *supra*.

57. *See* Etix's Weblog, These Companies Mislead Our Users, http://blog.l0cal.com/2011/07/07/these-companies-that-mislead-our-users/ (July 7, 2011) ("What bothers us the most is that many of them are bundling VLC with various crapware to monetize it in ways that mislead our users by thinking they're downloading an original version. This is not acceptable. The result is a poor product that doesn't work as intended . . . ."); Hja's Blog, Trademarks—the Good, the Bad and the Ugly, http://lockshot.wordpress.com/2009/04/20/trademarks-%E2%80%93-the-good-the-bad-and-the-ugly/ (April 20, 2009) ("The Bad. This category involves people who are intentionally trading on the brand for their own benefit. . . . This is especially offensive because these actors are trading on the value of the Firefox brand built by the community and ripping off users in the process."); *see, e.g.*, Slackware Trademark Policy, http://www.slackware.com/trademark/trademark.php (last visited Nov. 8, 2011) ("Note that you can still redistribute a distribution that doesn't meet these criteria, you just can't call it 'Slackware'. Personally, I hate restricting things in any way, but these restrictions are not designed to make life difficult for anyone. I just want to make sure that bugs are not added to commercial redistributions of Slackware. They have been in the past, and the resulting requests for help have flooded my mailbox! I'm just trying to make sure that I have some recourse when something like that happens.").

58. Tiffany (NJ) Inc. v. eBay Inc., 600 F.3d 93, 102 (2d Cir. 2010) (brackets omitted), quoting Merck & Co. v. Mediplan Health Consulting, Inc., 425 F. Supp. 2d 402, 413 (S.D.N.Y. 2006).

59. *See* New Kids on the Block v. News Am. Publ'g, Inc., 971 F.2d 302, 308 (9th Cir. 1992). (Defining nominative fair use as applicable in situations "where the only word reasonably available to describe a particular thing is pressed into service. . . . If the defendant's use of the plaintiff's trademark refers to something other than the plaintiff's product, the traditional fair use inquiry will continue to govern.").

code goods, but rather is using it as the name for its own newly-created version of the product.[60]

B. Licensing of Open Source Project Trademarks

Thus, the owner of the trademark for free and open source software is well within its rights when it does not allow others to label their newly-created versions with the same mark. But a FOSS project trademark owner may prefer a policy that allows use of its mark for more than unaltered redistribution[61] in order to foster the participation a healthy community requires.[62] It may do so by licensing the use of its trademark in limited ways that protect the significance of the trademark while allowing some use by third parties.

Trademark law allows an owner to license the use of its mark so long as it controls the quality of the goods and services offered by the licensee.[63] A trademark owner is not obliged to provide high quality goods or services, but simply must provide a quality of goods that is consistent and predictable: "the chief function of a trademark is a kind of 'warranty' to purchasers that they will receive, when they purchase goods bearing the mark, goods of the same character and source, anonymous as it may be, as other goods previously purchased bearing the mark that have already given the purchaser satisfaction."[64] A trademark licensor

---

60. Many FOSS trademark policies expressly permit nominative fair use, but the permission granted is not under these circumstances. Instead, the policies acknowledge that one may use the trademark to truthfully refer to the original software, not to label one's own version with the trademark. *See, e.g.*, OpenJDK, OpenJDK Trademark Notice Version 1.1, http://openjdk.java.net/legal/openjdk-trademark-notice.html (Mar. 10, 2008) ("The Name may also be used in connection with descriptions of the Software that constitute "fair use," such as "derived from the OpenJDK code base" or "based on the OpenJDK source code.").

61. *See, e.g.*, Francesco Poli, Re: Educational Community License 1.0 http://lists.debian.org/debian-legal/2011/08/msg00024.html (Aug. 27, 2011 00:25:23 +0200 GMT) ("If you consider such modifications as ground for forcing a name change, well, it may happen that most distros will distribute your piece of software under a different name: I am not sure that this is your goal, or that this is any good for you . . .").

62. *See* Julie Bort, Network World, Linux Foundation Chief: 'You Are an Idiot' if You Don't Give Back to Open Source, http://www.networkworld.com/news/2011/083011-zemlin-250234.html (Aug. 30, 2011) (quoting Jim Zemlin, executive director of the Linux Foundation: "It's not the right thing to do because of some moral issue or because we say you should do it. It's because you are an idiot if you don't. You're an idiot because the whole reason you're using open source is to collectively share in development and collectively maintain the software. Let me tell you, maintaining your own version of Linux ain't cheap, and it ain't easy.").

63. Barcamerica Int'l USA Trust v. Tyfield Importers, Inc., 289 F.3d 589, 595-906 (9th Cir. 2002) ("It is well-established that '[a] trademark owner may grant a license and remain protected provided quality control of the goods and services sold under the trademark by the licensee is maintained.'") (brackets in original), quoting Moore Bus. Forms, Inc. v. Ryu, 960 F.2d 486, 489 (5th Cir. 1992).

64. 1 J. Thomas McCarthy, McCarthy on Trademarks and Unfair Competition § 3:10 (4th ed.) (2011) [hereinafter McCarthy]; *see also* El Greco Leather Prods. Co. v. Shoe World,

necessarily relinquishes some loss of control over product quality by licensing, but as long as there is "reasonable control" over the product's quality, there is no forfeiture of trademark rights.[65]

Therefore, a trademark owner may permit its trademark to be used for executable code that is built from source code as long as there are only minor modifications.[66] By doing so, a project may be taking a risk that third party executable product will have some dissimilarity to its own executable build because of differences that might be introduced in the compilation process.[67] Nevertheless, if it ensures that the consumer experience will be relatively the same, for example, if it only allows use of the mark for software with cosmetic changes, bug fixes, patches, translations, or artwork,[68] consumer expectations will be met. Thus, enforcement of a standard that provides that a consumer will get a software that has generally the same functionality in all circumstances, whether it was installed from the project's original executable files or from another file, is a reasonable exercise of control over use of the trademark by others and preserves the validity of the trademark.

But a trademark license can go too far. Take this example of a provision in an open source trademark license, known as the 'MPL [Mozilla Public License]' with attribution" (and less flatteringly, "badgeware"):[69]

Inc., 806 F.2d 392, 395 (2d Cir. 1986) ("For this purpose the actual quality of the goods is irrelevant; it is the control of quality that a trademark holder is entitled to maintain.")

65. TMT North America, Inc. v. Magic Touch GmbH, 124 F.3d 876, 886 (7th Cir. 1997) ("Admittedly, licensing always entails some loss of control over product quality. If a licensor maintains reasonable control over product quality, however, consumers ultimately do rely upon the licensor's quality control. Absent a significant deviation from the licensor's quality standards, a licensor does not forfeit its trademark rights through licensing agreements."), *citing* Kentucky Fried Chicken Corp. v. Diversified Packaging Corp., 549 F.2d 368, 387 (5th Cir. 1977).

66. *See*, *e.g.*, The Document Foundation TradeMark Policy, http://wiki.documentfoundation.org/TradeMark_Policy (last visited Nov. 8, 2011) (stating that the mark can be used for software in "substantially unmodified form," which is defined as built from The Document Foundation source code with minor modifications, e.g., enabling or disabling features, translation, compatibility, patches, additional fonts, extensions, artwork, and templates).

67. *See* notes 11-15, *supra*, and accompanying text.

68. *See* note 66, *supra*.

69. At one point, twenty open source software companies were reported to be using this licensing language. Rick Moen, When is an Open Source License Open Source? http://www.linuxgazette.net/141/misc/lg/when_is_an_open_source_license_open_source.html (June 27, 2007 14:51:36 -700 GMT)("SugarCRM started the trend, and the other dozen-odd firms (Socialtext, Alfresco, Zimbra, Qlusters, Jitterbit, Scalix, MuleSource, Dimdim, Agnitas AG, Openbravo, Emu Software, Terracotta, Cognizo Technologies, ValueCard, KnowledgeTree, OpenCountry, 1BizCom, MedSphere, vTiger) literally copied their so-called 'MPL-style' licence, with minor variations.").

> This License does not grant any rights to use the trademarks "ABC"[70] and the "ABC" logos even if such marks are included in the Original Code or Modifications.
>
> However, in addition to the other notice obligations, all copies of the Covered Code in Executable and Source Code form distributed must, as a form of attribution of the original author, include on each user interface screen (i) the "ABC" logo . . . .[71]

In the license, "Covered Code" is defined as both the original code and any modifications of it, where modifications include additions, deletions, and new files.[72] Therefore, one can change the code without any limitations and the code will still be "Covered Code," with the resulting requirement that the ABC trademark be displayed.

The requirement that the word mark and logo must be used, without limiting whatsoever how many changes can be made to the code before the trademark must be removed, most likely means this trademark grant amounts to no more than a "naked license" under current trademark jurisprudence.[73] A trademark carries with it a message that the trademark owner is controlling the nature and quality of the goods or services sold under the mark.[74] Trademark law therefore imposes a duty on the trademark owner to control the quality of the goods and services with which the mark is used.[75] If the trademark owner does not, there is a risk

---

70. The actual names and marks are replaced by ABC for the purposes of this article.

71. Link to actual site last visited by the author on Sept. 18, 2011. The license continues: "In addition, the 'ABC' logo must be visible to all users and be in the same position as and at least as large as the ABC logo is within the Original Code. When users click on the original ABC logo it must direct them to http://www.abc.com/."

72. *See* n.71, *supra.*

73. The disclaimer that there is no license granted to the trademark, yet requiring that the trademark be used, is irreconcilably inconsistent. Frequently software trademark guidelines include a statement that one may use a trademark as long as there is no suggestion of affiliation or endorsement. *See*, *e.g.*, WordPress Foundation Trademark Policy, http://wordpressfoundation.org/trademark-policy/ (last visited Nov. 8, 2011) ("All other WordPress-related businesses or projects can use the WordPress name and logo to refer to and explain their services, but they cannot use them as part of a product, project, service, domain, or company name and they cannot use them in any way that suggests an affiliation with or endorsement by the WordPress Foundation or the WordPress open source project."); Debian Logos, http://www.debian.org/logos/ (last visited Nov. 8, 2011) ("This logo or a modified version may be used by anyone to refer to the Debian project, but does not indicate endorsement by the project."). Perhaps ABC's claim that no license is granted while simultaneously requiring use of the trademark is an inartful effort to convey that using the logo is not meant to suggest that there is any affiliation or endorsement by ABC of the subsequent distribution.

74. 3 McCarthy § 18:48.

75. Dawn Donut Co. v. Hart's Food Stores, Inc., 267 F.2d 358, 367 (2d Cir. 1959) ("Clearly the only effective way to protect the public where a trademark is used by licensees is to place on the licensor the affirmative duty of policing in a reasonable manner the activities of his licensees.").

that the various products with the same mark will be of diverse quality.[76] This practice, known as a "naked license," is "inherently deceptive and constitutes abandonment of any rights to the trademark by the licensor."[77]

The Court of Appeal for the Ninth Circuit has held that a grant similar to ABC's was a naked license. In *Freecyclesunnyvale v. The Freecycle Network*,[78] the defendant-licensor The Freecycle Network (TFN) owned the below trademark:

TFN had a "Freecycle Ethos," which was a democratic leadership structure in which decisions were made through a process of surveys and discussions among volunteer moderators.[79] There was consensus amongst the various local member groups that the guiding rule was "Keep it Free, Legal & Appropriate for All Ages."[80] The local groups participated in decision-making using Yahoo! Groups.[81] Plaintiff FreecycleSunnyvale (FS) came into existence by joining Yahoo! Groups and adopting TFN's etiquette guidelines and instructions. TFN licensed the trademark by email to Freecycle Sunnyvale with the instructions "just don't use it for commercial purposes."[82]

The 9th Circuit held that the email was not a license that gave TFN control over the Freecycle Sunnyvale services because The Freecycle Network had "no ability to terminate FS's license if FS

---

76. *Dawn Donut*, 267 F.2d at 367.

77. Barcamerica Int'l USA Trust v. Tyfield Importers, Inc., 289 F.3d 589, 598 (9th Cir. 2002); Kentucky Fried Chicken Corp. v. Diversified Packaging Corp., 549 F.2d 368, 387 (5th Cir. 1977) ("If a trademark owner allows licensees to depart from its quality standards, the public will be misled, and the trademark will cease to have utility as an informational device. A trademark owner who allows this to occur loses its right to use the mark."); Doeblers' Pennsylvania Hybrids, Inc. v. Doebler, 442 F.3d 812, 823 (3d Cir. 2006) ("Failure to provide quality control may constitute naked licensing leading to abandonment of the mark."); *see* 3 McCarthy § 18:48, n.9 (summarizing cases).

78. 626 F.3d 509 (9th Cir. 2010).

79. Freecyclesunnyvale v. The Freecycle Network, 626 F.3d 509, 512 (9th Cir. 2010). Note how similar this is in description to the management of FOSS projects at Section I.B, *supra*.

80. *Id.* at 513.

81. *Id.* at 512.

82. *Id.* at 513.

used the trademark for commercial purposes."[83] Further, the court held that the rule that the members "Keep it Free, Legal & Appropriate for All Ages," the "Freecycle Ethos," and the terms of use for Yahoo! Groups were also inadequate to exercise actual control over the use of the mark.[84] The court held that (i) there was no requirement that the "Keep it Free, Legal & Appropriate for All Ages" standard be adopted by the groups; (ii) the Yahoo! Groups terms of use were irrelevant; (iii) the non-commercial limitation was not a restriction on the quality of goods and services under the mark; and (iv) the Freecycle Ethos was subject to local control, not control by TFN.[85]

In this context, consider the above requirement that the trademark must be used with software that has been modified without any restriction on how much the software can be changed.[86] This is even less control that in FreecycleSunnyvale; ABCs limitation only refers to when the mark may be used and says nothing whatsoever about the quality of the altered software product it is used with.[87] Query whether ABC might be risking loss of trademark rights in its mark under the Freecyclesunnyvale standard.[88]

CONCLUSION

There are legal tensions between the open source ethos and reasonable standards for ensuring that a trademark remains meaningful. A trademark owner has the obligation to control the use of its mark and, under the law of abandonment and naked licensing, risks loss if the control is not maintained. The trademark owner, therefore, cannot be as liberal in granting rights

83. *Id.* at 516. The author strenuously disagrees with this reasoning. The license was granted on the condition that the trademark could only be used noncommercially. A commercial use would be outside the scope of the license and a material breach, terminating the license. *See* Manpower Inc. v. Mason 405 F. Supp. 2d 959, 968 (E.D. Wis. 2005) (holding that a use outside of the licensed territory was a material breach); E.G.L. Gem Lab Ltd. v. Gem Quality Institute, Inc., 90 F. Supp. 2d 277, 305 (S.D.N.Y. 2000) (same); Restatement (Second) of Contracts § 241, cmt. e (1981) (stating that a material breach excuses performance by both parties).

84. *Id.* at 517.

85. Id. at 517-18.

86. *See* note 71, *supra*, and accompanying text.

87. *See, c.f.*, CNA Financial Corp. v. Brown, 922 F. Supp. 567, 571 (M.D. Fla. 1996), *aff'd on other grounds*, 162 F.3d 1334 (11th Cir. 1998) (distinguishing control of the appearance of the trademark from control over the goods and services: "Although CNAF controls the use of its service marks, it does not control the nature and quality of the insurance services that the CNAF family offers in connection with the service marks.").

88. In the view of the author, the *Freecyclesunnyvale* case has expanded the naked license doctrine beyond any reasonable statutory basis available under the Lanham Act for invalidating trademarks. *See* Property, Intangible, Ninth Circuit Ignores the Law Again, http://www.propertyintangible.com/2010/11/ninth-circuit-ignores-law-again.html (Nov. 29, 2010).

under trademark law as it may under copyright law if it hopes to maintain the validity of the trademark. Nevertheless, there is a fairly wide range of practices available to a trademark owner that allow it to satisfy the community's desire to freely share software code while still protecting consumer expectations about the product that they receive.