OPEN SOURCE IS NOT PUBLIC DOMAIN: EVOLVING LICENSING PHILOSOPHIES

BRAD FRAZER*

TABLE OF CONTENTS

I. INTRODUCTION: "OPEN SOURCE" DOES NOT MEAN "PUBLIC DOMAIN" 2
A. Introduction 2
B. Copyright Basics 3
C. Who can Claim Copyright 4
D. What is not Protected by Copyright? 5
E. Copyright Secured Automatically upon Creation 6
F. Copyright Registration 7
G. Copyright Infringement 7
II. OPEN SOURCE SOFTWARE 9
A. Introduction 9
B. Definitions 11
C. Examples of Open Source Licenses 14
III. OPEN SOURCE LICENSING 17
A. Introduction 17
B. Licensing Methods 17
C. The Katzer Case 18
D. The GNU General Public License Version Three 23
IV. BUSINESS ISSUES 26
A. Mergers, Acquisitions, and Due Diligence 26

[*] Boise native Brad Frazer is a 1985 *cum laude* graduate of Brigham Young University (Provo) and a 1988 graduate of the University of California, Hastings College of the Law, where he earned a position on the College's law review. After law school, he received his Masters of Business Administration degree from the University of Utah.

From 1989 to 1997, Brad worked at the Boise law firm Elam & Burke, becoming a shareholder in 1997. In 1997, Brad accepted an in-house position as Senior Intellectual Property Counsel with Fortune 1000 company Micron Electronics, Inc. and remained with Micron Electronics through its merger with Interland, Inc. (now Web.com), the nation's largest web hosting company. Brad eventually became Deputy General Counsel to Interland.

From June 2003 to February 2006, Brad served as Deputy General Counsel to MPC Computers, LLC, and he is now a partner at Hawley Troxell Ennis & Hawley LLP, where his practice areas include Internet law, e-commerce, technology and software licensing, trademarks and domain names, copyright, media law, computer law, trade secrets, and related transactional work and litigation. He blogs about these issues at http://internetlawyer.blogspot.com/.

B. Business Planning 26
C. Product Engineering 27
V. CONCLUSION 27

I. INTRODUCTION: "OPEN SOURCE" DOES NOT MEAN "PUBLIC DOMAIN"

Much attention has been garnered recently by so-called "open source software" and the ostensible mechanism by which it is distributed, "open source licensing." Accordingly, there is much misunderstanding about what exactly is meant by "open source software" and "open source licensing." The terms are not necessarily synonymous with either shareware or freeware, and open source software is almost certainly not in the public domain (another misunderstood and misused term of art in copyright law). Yet, these misconceptions tend to propagate rapidly as new versions and types of open source software become available to the public and as more organizations proselytize the use of open source in contravention of established "closed" proprietary software architectures, such as those promoted by companies like Microsoft.

There is nothing magical about open source software. "Open source" does not mean that it contains special, secret computer code, that it has been written in a particular "open" way, or that it is devoid of copyright protection. Indeed, this article argues that "open source," as a construct, inherently connotes nothing about whether copyright has attached to the licensed software, nor does the term suggest that anarchistic abandonment of downstream control over the underlying computer code has occurred. Rather, this article argues that "open source" is a licensing philosophy to be employed by the owner of the copyright in the software in question in recognition of the axiom that collaboration is better than insular behavior. Said another way, the concept of "open source" teaches that two heads are better than one.

A. Introduction

Software—including open source software—is copyrightable subject matter,[1] and therefore, any discussion concerning its use and licensing must be prefaced with an introduction to key copyright law concepts. A software license—including open source software license—is at its core a copyright license. To be clear, a copyright attaches to open source software absent a finding that the underlying code is in the public domain or that there has been an express abandonment of the copyright by the

[1] *See, e.g.*, SCO Group, Inc., v. Novell, Inc., No. 2:04CV139DAK, 2007 WL 2327587, at 56* (D. Utah Aug. 10, 2007) (concluding that Novell is the owner of the UNIX and UnixWare copyrights).

author of the code. It is therefore grossly incorrect to conclude, for example, that the following syllogism is true: "All Linux is open source, and all open source is in the public domain; therefore, Linux is in the public domain, and I may use it, and copy it, and modify it with complete impunity." Many lawsuits have been spawned by such flawed logic.

"Copyright" is not a verb; it is a noun. It is technically incorrect to state that one is going to "copyright" something. For example, it is common for laypersons to make a statement akin to the following: "I copyrighted my website by placing the © symbol on the bottom of the page." From a copyright law perspective, that statement is nonsensical because "copyright" is a noun; the author of the website owns a copyright in the website by virtue of the act of authorship. A copyright is an incorporeal property right that springs into existence automatically when a sufficiently creative idea is reduced by the author into or onto a tangible medium, such as paper, film, clay, or, in the case of computer software, magnetic media, such as a hard drive or CD-ROM.[2] Important rights and statutory protections are gained by registering the copyright an author creates, but no registration is needed, nor is the ubiquitous © symbol required to create a copyright.[3]

B. Copyright Basics

Copyright is a form of protection provided by the laws of the United States to the authors of "original works of authorship," including literary[4], dramatic, musical, artistic, and certain other intellectual works.[5] This protection is available to both published and unpublished works.[6] The 1976 Copyright Act generally gives the owner of a copyright the exclusive right to do, and to authorize others to do, the following:

> to reproduce the copyrighted work in copies or phonorecords;
>
> to prepare derivative copyrighted works based upon the work;
>
> to distribute copies or phonorecords of the work to the public by sale or other transfer of ownership, or by rental, lease, or lending;
>
> in the case of literary, musical, dramatic, and choreographic works, pantomimes, and motion pictures and other audiovisual works to perform the work publicly;

[2] 17 U.S.C. § 102(a) (2007).

[3] *Id.*

[4] In general, computer programs are considered to be literary works for purposes of copyright law analysis. *See* Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 702 (2d Cir. 1992) ("[L]egislative history leaves no doubt that Congress intended [computer programs] to be considered literary works.").

[5] 17 U.S.C. § 102(a).

[6] For copyright law definitions, see 17 U.S.C. § 101.

> in the case of literary, musical, dramatic, and choreographic works, pantomimes, and pictorial, graphic, or sculptural works, including the individual images of a motion picture or other audiovisual work to display the copyrighted work publicly; and
>
> in the case of sound recordings, to perform the copyrighted work publicly by means of a digital audio transmission.[7]

In addition, certain authors of works of visual art have the rights of attribution and integrity as described in section 106A of the 1976 Copyright Act,[8] also known as the "Visual Artists Rights Act."

It is illegal for anyone to violate any of these exclusive rights provided by copyright law to the copyright owner (the Section 106 Rights) without a license (permission) from the owner.[9] These rights, however, are not unlimited in scope. Sections 107 through 121 of the 1976 Copyright Act establish limitations on these rights.[10] In some cases, these limitations are specified exemptions from copyright liability. One major limitation is the doctrine of fair use, which is given a statutory basis in section 107 of the 1976 Copyright Act.[11] In other instances, the limitation takes the form of a "compulsory license" under which certain limited uses of copyrighted works are permitted upon payment of specified royalties and compliance with statutory conditions.[12] Similarly, there can be no copyright infringement of a work that has fallen into the public domain. To be in the public domain means the copyright is not perpetual,[13] that the original copyright for the work has expired, or the copyright, by its nature, is not subject to copyright protection *ab initio* (for example, works created by the United States government).[14] It is difficult, at best, to determine definitively that a work is in the public domain or that a particular use is fair, and, thus, it is always preferable to err on the side of caution and obtain a copyright license from the owner of the copyright.[15]

C. Who can Claim a Copyright

Copyright protection begins when the work is created in fixed form and persists for the statutory duration.[16] The copyright in the work of authorship immediately becomes the property of the author who created

[7] *Id.* § 106(1)–(6).

[8] *Id.* § 106A.

[9] *Id.* § 501(a).

[10] *Id.* §§ 107–121.

[11] *Id.* § 107.

[12] *See, e.g.*, *id.* § 115.

[13] *See generally id.* §§ 301–305.

[14] *See id.* § 105.

[15] Another useful copyright law maxim to remember: "Attribution is not permission." It is common for a layperson to conclude that he may use a work as long as attribution to the source or author is given. This is incorrect.

[16] 17 U.S.C. §§ 302–303.

the work. Only the author or those deriving their rights through the author can rightfully claim copyright, although the authors of a joint work are co-owners of the copyright in the work, unless there is an agreement to the contrary.[17] Moreover, a copyright in each separate contribution to a periodical or other collective work is distinct from a copyright in the collective work as a whole and vests initially with the author of the particular contribution.[18]

In the case of works made for hire, the employer, not the employee, is considered the author.[19] Section 101 of the Copyright Act[20] defines a "work made for hire" as:

> (1) a work prepared by an employee within the scope of his or her employment; or
>
> (2) a work specially ordered or commissioned for use as:
>
> - a contribution to a collective work,
> - a part of a motion picture or other audiovisual work,
> - a translation,
> - a supplementary work,
> - a compilation,
> - an instructional text,
> - a test,
> - answer material for a test, or
> - an atlas
>
> *if* the parties expressly agree in a written instrument signed by them that the work shall be considered a work made for hire.[21]

Thus, the copyright in a work created by an independent contractor, absent either an employee-employer relationship or a signed copyright assignment agreement, will belong to the independent contractor. This is contrary to the common layperson's misconception that, "If I paid for it, I own it." This is an extraordinarily relevant consideration when one is negotiating a software (copyright) license, for if the putative licensor does not own the copyright, the licensee may be subject to claims of copyright infringement by an unknown third-party author who later arises and objects to the putative licensee's use of his or her work.

D. What is not Protected by a Copyright?

Several categories of material are generally not eligible for federal copyright protection. These include:

[17] *Id.* § 201(a).
[18] *Id.* § 201(c).
[19] *Id.* § 201(b).
[20] *Id.* § 101.
[21] *Id.* (emphasis added) (original structure altered).

- Works that have not been fixed in a tangible form of expression (for example, choreographic works that have not been notated or recorded, or improvisational speeches or performances that have not been written or recorded).
- Titles, names, short phrases, and slogans; familiar symbols or designs; mere variations of typographic ornamentation, lettering, or coloring; and mere listings of ingredients or contents.[22]
- Ideas, procedures, methods, systems, processes, concepts, principles, discoveries, or devices, as distinguished from a description, explanation, or illustration.[23]
- Works consisting entirely of information that is common property and containing no original authorship (for example, standard calendars, height and weight charts, tape measures and rulers, and lists or tables taken from public documents or other common sources).[24]

E. Copyright Secured Automatically upon Creation

The way in which a copyright protection is secured is frequently misunderstood. A copyright is secured automatically when the work is created.[25] "[A] work is 'created' when it is fixed in a copy or phonorecord for the first time; . . ."[26] "Copies" are material objects from which a work can be read or visually perceived, "either directly or with the aid of a machine or device"[27] (for example, books, manuscripts, sheet music, film, videotape, or magnetic media, such as CD-ROM). "Phonorecords" are material objects embodying fixations of sounds, excluding, by statutory definition, motion picture soundtracks[28] (for example, cassette tapes, CDs, or LPs). Thus, for example, a song (the work) can be fixed in sheet music (copies), in phonograph disks (phonorecords), or both. If a work is prepared over a period of time, the part of the work that is fixed on a particular date constitutes the created work as of that date.

[22] U.S. COPYRIGHT OFFICE, CIRCULAR NO. 34. COPYRIGHT PROTECTION NOT AVAILABLE FOR NAMES, TITLES, OR SHORT PHRASES (2006), *available at* http://www.copyright.gov/circs/ circ34.pdf.

[23] *See* 17 U.S.C. § 102(b). Interestingly, some commentators have argued that a computer program, as a "process," should not be protectable in copyright under this code section. But the case law makes it clear that a computer program is copyrightable subject matter, as long as what is sought to be protected is the literal expression of the code into a tangible medium and not the underlying functionality. The functionality is protected using a different tool—patent law.

[24] *See* U.S. COPYRIGHT OFFICE, *supra* note 22.

[25] *See* 17 U.S.C. § 102.

[26] *See id.* § 101.

[27] *Id.*

[28] *Id.*

F. Copyright Registration

No publication, registration, or other action in the Copyright Office is required to secure ownership of a copyright.[29] Creation of a copyright occurs when a sufficiently creative idea is embodied in a tangible medium.[30] The author of that creation owns the copyright—and its associated bundle of exclusive Section 106 Rights—in and to the work.[31] There are, however, definite advantages to registration, and a copyright is best protected and enforced through registration.[32] Some of the advantages of registration are:

- Registration establishes a public record of the copyright claim.
- Before an infringement suit may be filed in court, registration is necessary for works of U.S. origin.
- If made before or within [five] years of publication, registration will establish prima facie evidence in court of the validity of the copyright and of the facts stated in the certificate.
- If registration is made within [three] months after publication of the work or prior to an infringement of the work, statutory damages and attorney's fees will be available to the copyright owner in court actions. Otherwise, only an award of actual damages and profits is available to the copyright owner.[33]

Registration is accomplished by sending one or more copies (deposit copies) of the work, along with a properly completed registration application and the correct nonrefundable filing fee to the Copyright Office at the Library of Congress.[34] Registration is accomplished most expeditiously and inexpensively through online filing at the Electronic Copyright Office.[35]

G. Copyright Infringement

> To prove copyright infringement, a plaintiff must demonstrate (1) ownership of the allegedly infringed work and (2) copying of the protected elements of the work by the defendant. Because

[29] *See id.* § 412.

[30] *Id.* § 102.

[31] *Id.* § 106.

[32] *See id.* § 412.

[33] Electronic Copyright Office, Copyright Registration, http://www.copyright.gov/circs/circ1.html#cr (last visited Feb. 23, 2009).

[34] Electronic Copyright Office, Frequently Asked Questions, http://www.copyright.gov/eco/index.html (last visited Feb. 23, 2009). The number of deposit copies one must submit depends on certain factors.

[35] *Id.*

> direct copying is difficult to prove, a plaintiff can satisfy the second element by demonstrating that (a) the defendant had access to the allegedly infringed work and (b) the two works are substantially similar in both idea and expression of that idea.[36]

Proof of access requires "an opportunity to view or to copy plaintiff's work."[37] This is often described as providing a reasonable opportunity or reasonable possibility of viewing the plaintiff's work.[38] Of course, reasonable opportunity, in this context, does not encompass any bare possibility; access "may not be inferred through [mere] speculation or conjecture."[39] Circumstantial evidence of reasonable access is proven in one of two ways: (1) a particular chain of events is established between the plaintiff's work and the defendant's access to that work, or (2) the plaintiff's work has been widely disseminated.[40]

Proof of substantial similarity in a software context is a complex topic beyond the scope of this article, but in general, courts facing the issue will apply the "abstraction—filtration—comparison" test adopted by the Second Circuit in *Computer Associates International, Inc. v. Altai, Inc.*[41] Essentially, the court will look at the protectable expression contained within the plaintiff's source code, compare it with the defendant's source code, and determine, by using the *Altai* test, if the two are substantially similar.[42] Upon a finding of infringement, a prevailing plaintiff may be entitled to remedies under the United States Code.[43] These remedies may include injunctive relief, monetary damages (either actual or statutory), an impoundment and destruction order, and attorney fees and costs to the prevailing party.[44]

The following hypothetical will help coalesce each of these concepts into a whole. Assume that Company X hires an independent contractor to write computer code. Company X pays the programmer, Jim, $5,000, and Jim delivers a CD-ROM with the commissioned program on it. Company X thereafter begins to license the software to third parties. A short time later, Jim's former employer sues Company X for copyright infringement asserting that it owns the copyright, not Company X, because Jim created it while he was their employee, and Company X did not obtain a written copyright assignment from either Jim or Jim's former employer. Company X settles that lawsuit and buys the copyright to

[36] Pasillas v. McDonald's Corp., 927 F.2d 440, 442 (9th Cir. 1991) (citation omitted).

[37] Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp., 562 F.2d 1157, 1172 (9th Cir. 1977).

[38] Jorgensen v. Epic/Sony Records, 351 F.3d 46, 51 (2d Cir. 2003).

[39] Gaste v. Kaiserman, 863 F.2d 1061, 1066 (2d Cir. 1988).

[40] Tomasini v. Walt Disney Co., 84 F. Supp. 2d 516, 519 (S.D.N.Y. 2000).

[41] Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 706 (2d Cir. 1992).

[42] *See id.*

[43] 17 U.S.C. § 501 (2007).

[44] *Id.* §§ 502–505.

the code from Jim's former employer.[45] Company X again begins licensing the code to third parties. One of those licensees begins selling copies of the software without permission, and Company X sues them for copyright infringement. The case is dismissed, however, because Company X did not register its copyright in the software before filing the complaint. Company X obtains registration and refiles the complaint, but because it waited for more than three months after the date of first publication to register the copyright, Company X may not obtain either statutory damages or attorney fees in the infringement litigation.

II. OPEN SOURCE SOFTWARE

A. Introduction

Open source does not connote an abandonment of a copyright. If one infringes an extant, viable copyright in open source software, *Katzer* teaches that one is just as culpable and exposed to possible infringement liability as if one had pirated a copy of a Microsoft Windows operating system, the obvious antithesis to open source.[46] A wise user of open source software knows that the computer code at issue is still very likely subject to copyright protection and will abide by the license terms offered by the licensor to avoid claims of copyright infringement.

Understanding open source software and open source licensing requires an understanding of certain fundamental computer concepts. As a preliminary matter, the thing that is "open," in this context, is the computer source code; hence, "open source." Computers run software to perform operations, and software falls into two very broad categories called source code and object code.[47] Object code is binary code comprised of a series of bits and bytes, of ones and zeroes that are not generally perceivable by humans without machine intervention.[48] Source code, on the other hand, is written in language that is perceivable by humans, and in order for it to be understood at the machine level so that it will run on a computer, the source code is compiled, by which it is turned into object or machine-language code.[49]

Thus, if an end user were able to look at software while it was running inside a computer or while it reposed on the DVD or CD-ROM media on which it was licensed, the end user would see merely the object code—binary data—ones and zeroes, and would not be able to divine the

[45] Note that Jim's former employer may also, on these facts, sue each of Company X's putative licensees.

[46] Jacobsen v. Katzer, 535 F.3d 1373 (Fed. Cir. 2008).

[47] *See* Wikipedia.org, Computer Program, http://en.wikipedia.org/wiki/computer_program (last visited Feb. 23, 2009).

[48] *Id.*

[49] *Id.*

mechanisms behind the operability of the software. Because source code, on the other hand, is perceivable and understandable by humans, great care is typically taken by the owners of software to ensure that end users of the software cannot actually view the source code because those end users could then understand the human thought and processes that went into developing the perhaps extraordinarily complex algorithms that cause the software do to what it does. Typically, many hundreds or thousands of hours of programmer labor are expended writing source code that, when compiled and run, will produce the intended functionality (for example, a video game or a spreadsheet program). An unscrupulous competitor or end user could circumvent those many hours of needed programmer labor by viewing the source code through reverse engineering or decompiling the code, seeing how the human authors (programmers) made the software perform as it does. The unscrupulous competitor could then copy those processes and steps, thereby creating a version of source code that, when compiled into object code and run on a computer, will provide the same basic functionality as the original software without having had to expend thousands of programmer hours writing native source code from the ground up. For example, Microsoft would not want Apple to see the source code for the Microsoft Windows operating system because Apple could see the human processes that went into making the operating system function as it does when compiled and run and could thus more easily replicate those functions without investing many additional hours into writing native source code.

The notion of open source code is counterintuitive because if an end user has access to the source code, the software's functionality may be more readily replicated. That is why this article takes the position that open source is a licensing philosophy: the programmers (who arguably are also the authors and, thus, the copyright owners) are granting access to the underlying source code specifically because they wish third persons to see how the software operates. Then those third persons may perhaps find ways to improve the code, make it run faster or on more operating system platforms, develop new applications for it, and in general, make the software more usable by more people through a spirit of cooperation. The licenses that contractually govern how the programmers/authors/copyright owners distribute and grant use rights to the open source software contemplate and, in fact, many times mandate, this collaborative spirit. Indeed, this open philosophy may be inferred from many of the most common definitions used within the open source community, as set forth below. There is no central repository of agreed-upon definitions in this arena, so there may be some arbitrariness and variance in nomenclature depending to which authority one appeals, but the following are in generally accepted usage within the open source lexicon. Some come from the Free Software Foundation, the entity responsible for the granddaddy of all open source licenses, the GNU General Public License, and, thus, are likely authoritative.

B. Definitions

Open Source Software: One of the best definitions of "open source" is found in the recently decided Federal Circuit case *Jacobsen v. Katzer*:[50]

> Open Source software projects invite computer programmers from around the world to view software code and make changes and improvements to it. Through such collaboration, software programs can often be written and debugged faster and at lower cost than if the copyright holder were required to do all of the work independently. In exchange and in consideration for this collaborative work, the copyright holder permits users to copy, modify and distribute the software code subject to conditions that serve to protect downstream users and to keep the code accessible.[51]

Freeware: "The term 'freeware' has no clear accepted definition, but it is commonly used for packages which permit redistribution but not modification (and their source code is not available). These packages are *not* free software . . . ."[52]

Free Software: "[S]oftware that comes with permission for anyone to use, copy, and distribute, either verbatim or with modifications, either gratis or for a fee. In particular, this means that source code must be available."[53]

Shareware:

> Shareware is software which comes with permission for people to redistribute copies, but says that anyone who continues to use a copy is *required* to pay a license fee.
>
> Shareware is not free software [as defined by the Free Software Foundation], or even semi-free. There are two reasons it is not:
>
> - For most shareware, source code is not available; thus, you cannot modify the program at all.

[50] Jacobsen v. Katzer, 535 F.3d 1373 (Fed. Cir. 2008).

[51] *Id.* at 1378–79.

[52] Free Software Foundation, http://www.fsf.org/licensing/essays/categories.html#freeware (last visited Feb. 23, 2009).

[53] Free Software Foundation, http://www.fsf.org/licensing/essays/categories.html#FreeSoftware (last visited Feb. 23, 2009).

- Shareware does not come with permission to make a copy and install it without paying a license fee, not even for individuals engaging in nonprofit activity.[54]

Open Source License: A license is the contract by which the copyright owner grants use rights to the software to end users. Such licenses may take many forms, such as shrink-wrap, click-wrap, and actual paper contracts signed with pen and ink.[55] There is nothing magic about an open source license in terms of the actual licensing mechanism, as the open source nature of the license is imbued within and conveyed by the contractual terms and covenants imposed by the licensor and agreed to by the licensee. A very comprehensive definition of an open source license is the one promulgated by the Open Source Initiative (OSI), a California 501(c)(3) corporation that fosters and champions various open source programs and activities.[56] The OSI's definition has ten components:

> 1. *Free Redistribution*: The license shall 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 shall not require a royalty or other fee for such sale.
>
> 2. *Source Code*: The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.
>
> 3. *Derived Works*: The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
>
> 4. *Integrity of The Author's Source* Code: The license may restrict source-code from being distributed in modified form *only* if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software

[54] Free Software Foundation, http://www.fsf.org/licensing/essays/categories.html#shareware (last visited Feb. 23, 2009).

[55] *See generally*, Open Source.org, Open Source Initiative http://opensource.org/licenses/alphabetical (last visited Feb. 23, 2009).

[56] Open Source.org, About the Open Soure Initiative, http://www.opensource.org/about (last visited Feb. 23, 2009).

> built from modified source code. The license may require derived works to carry a different name or version number from the original software.
>
> 5. *No Discrimination Against Persons or Groups*: The license must not discriminate against any person or group of persons.
>
> 6. *No Discrimination Against Fields of Endeavor*: The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.
>
> 7. *Distribution of License*: The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
>
> 8. *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.
>
> 9. *License Must Not Restrict Other Software*: The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.
>
> 10. *License Must Be Technology-Neutral*: No provision of the license may be predicated on any individual technology or style of interface.[57]

Any licensor of purportedly open source software may impose any contractual terms the licensor wishes on downstream end users via the license agreement, but in order to be a license approved in form by the OSI, the above ten elements must be present.[58] Note that in contrast to the OSI, other organizations, such as the Free Software Foundation ("FSF"), favor an even less restrictive licensing scheme. To that end, the FSF itself refers to its GNU General Public License as a free software

[57] Open Source.org, The Open Source Definition, http://opensource.org/docs/osd (last visited Feb. 23, 2009) (emphasis added).

[58] Open Source.org, Frequently Answered Questions, http://opensource.org/faq (last visited Feb. 23, 2009).

license.[59] A detailed discussion of the differences between the FSF and the OSI in terms of their respective licensing and copyright philosophies is beyond the scope of this article. Accordingly, use of the term open source, herein, includes, by reference, free software.[60]

Proprietary Software: "Proprietary software is software that is not free or semi-free. Its use, redistribution or modification is prohibited, or requires you to ask for permission, or is restricted so much that you effectively can't do it freely."[61] For example, the Microsoft Windows Operating System would be considered proprietary, both because Microsoft closely guards the source code as a trade secret (the antithesis to open source code) and because Microsoft's license terms prohibit use of the code that might otherwise be permitted under an open source license.[62]

C. Examples of Open Source Licenses

OSI's website lists the names of a number of current open source licenses that accord with its proffered definition.[63] The list illustrates the scope and variety of open source licenses an end user may encounter. Each of these likely has different specific contractual terms, but at their core they must each comport with the open aspects of the OSI definition:

- Academic Free License 3.0 (AFL 3.0)
- Affero GNU Public License
- Adaptive Public License
- Apache Software License
- Apache License, 2.0
- Apple Public Source License
- Artistic license
- Artistic license 2.0
- Attribution Assurance Licenses
- New and Simplified BSD licenses
- Boost Software License (BSL1.0)
- Computer Associates Trusted Open Source License 1.1

[59] FSF.org, Categories of Free and Non-Free Software, http://www.fsf.org/licensing/essays/categories.html# (last visited Feb. 23, 2009). The FSF observes: "The term 'open source' software is used by some people to mean more or less the same category as free software. [However, their criteria are somewhat lax;] they accept some licenses that we consider too restrictive . . . we prefer the term 'free software' because it refers to freedom—something that the term 'open source' does not do." *Id.*

[60] FSF.org, The Free Software Definition, http://www.fsf. org/licensing/essays/free-sw.html (last visited Feb. 23, 2009).

[61] FSF.org, *supra* note 59.

[62] *See* Wikipedia.org, Comparison of Open Source and Closed Source, http://en.wikipedia.org/wiki/Comparison_of_open_source_and_closed_source (last visited Feb. 23, 2009).

[63] Open Source.org, Licenses by Name, http://opensource.org/licenses/alphabetical (last visited February 19, 2009)

- Common Development and Distribution License
- Common Public Attribution License 1.0 (CPAL)
- Common Public License 1.0
- CUA Office Public License Version 1.0
- EU DataGrid Software License
- Eclipse Public License
- Educational Community License, Version 2.0
- Eiffel Forum License
- Eiffel Forum License V2.0
- Entessa Public License
- Fair License
- Frameworx License
- GNU General Public License (GPL)
- GNU General Public License version 3.0 (GPLv3)
- GNU Library or "Lesser" General Public License (LGPL)
- GNU Library or "Lesser" General Public License version 3.0 (LGPLv3)
- Historical Permission Notice and Disclaimer
- IBM Public License
- Intel Open Source License
- ISC License
- Jabber Open Source License
- Lucent Public License (Plan9)
- Lucent Public License Version 1.02
- Microsoft Public License (Ms-PL)
- Microsoft Reciprocal License (Ms-RL)
- MIT license
- MITRE Collaborative Virtual Workspace License (CVW License)
- Motosoto License
- Mozilla Public License 1.0 (MPL)
- Mozilla Public License 1.1 (MPL)
- Multics License
- NASA Open Source Agreement 1.3
- NTP License
- Naumen Public License
- Nethack General Public License
- Nokia Open Source License
- Non-Profit Open Software License 3.0 (Non-Profit OSL 3.0)
- OCLC Research Public License 2.0
- Open Group Test Suite License
- Open Software License 3.0 (OSL 3.0)
- PHP License
- Python license (CNRI Python License)
- Python Software Foundation License
- Qt Public License (QPL)

- RealNetworks Public Source License V1.0
- Reciprocal Public License
- Reciprocal Public License 1.5 (RPL1.5)
- Ricoh Source Code Public License
- Simple Public License 2.0
- Sleepycat License
- Sun Industry Standards Source License (SISSL)
- Sun Public License
- Sybase Open Watcom Public License 1.0
- University of Illinois/NCSA Open Source License
- Vovida Software License v. 1.0
- W3C License
- wxWindows Library License
- X.Net License
- Zope Public License
- zlib/libpng license[64]

Notably absent from the OSI list is Linux, perhaps the most well-known open source software. This explanation from the Wikipedia entry on Linux is helpful:

> 'Linux' . . . is a generic term referring to Unix-like computer operating systems based on the Linux kernel. Their development is one of the most prominent examples of free and open source software collaboration; typically all the underlying source code can be used, freely modified, and redistributed by anyone under the terms of the GNU GPL and other free licenses.[65]

Thus, there is no separate, specific Linux open source license. The underlying kernel is licensed under the GNU General Public License promulgated by the FSF or a variant thereof. Modifications and different incarnations of the kernel, such as that version of Linux offered by licensors such as RedHat, are likely distributed subject to those licensors' particular license terms.

Also missing from the OSI list is any reference to one or more of the licenses promulgated by Creative Commons.[66] The various types of open

[64] Open Source.org, The License Review Process, http://opensource.org/licenses/alphabetical (last visited Feb. 23, 2009).

[65] Wikipedia.org, Linux, http://en.wikipedia.org/wiki/Linux (last visited Feb. 23, 2009); *see also* Richard Stallman, *GNU/Linux FAQ*, http://www.gnu.org/gnu/gnu-linux-faq.html#why (last visited Feb. 23, 2009).

[66] Creative Commons, About, http://creativecommons.org/about (last visited Feb. 23, 2009) ("Creative Commons is a nonprofit corporation dedicated to making it easier for people to share and build upon the work of others, consistent with the rules of copyright. [It] provide[s] free licenses and other legal tools to mark creative work with the freedom the creator wants it to carry, so others can share, remix, use commercially, or any combination thereof.").

licenses promulgated by Creative Commons and instructions on how to use them are available on the Creative Commons website.[67]

III. OPEN SOURCE LICENSING

A. Introduction

By definition, a license is a contract granting permission to another party to perform some act.[68] In a classic copyright context, a license may specifically be looked upon as a covenant not to sue for infringement of a right, and thus, a software license is a covenant not to sue for copyright infringement. The licensor is ostensibly the owner of the copyright and is permitting the downstream end-user licensee to violate one or more of the Section 106 Rights granted to the owner of a copyright. For example, the licensor is granting a right to the licensee to copy the work by installing a copy of the code or pieces of the code on a computer's hard drive and agrees not to sue the licensee for copyright infringement as long as the licensee observes the contractual covenants of the license agreement.

B. Licensing Methods

The means by which software is licensed are as varied as the colors of the rainbow and run the gamut from an implied verbal license to a full-blown, heavily negotiated, multi-page, pen-and-ink-signatures contract. At one end of the spectrum, if the owner of the copyright in software knowingly permits a third party to use the software in contravention of one or more of the owner's rights without an express contractual understanding governing that use, an implied license may be said to arise from that course of conduct. Next, in the Internet space, a variant known as a "browse-wrap" license has arisen, in which a visitor to a website will confront terms of use that contain a provision stating in essence that if the visitor continues past that point in the site, the visitor will be deemed to have accepted the terms of use (the license) granted by the owner of the copyright in the content on the site to use it in the manner prescribed. A more overt but similar licensing scheme is the "click-wrap" license, in which a fact pattern similar to the browse-wrap license arises, but the site visitor is caused to actively review a license agreement and click on an "I Accept" button before being allowed to proceed and view the site or download content.[69] Even more licensee involvement is required by the "shrink-wrap" license,[70] which arises

[67] Creative Commons, About Licenses, http://creativecommons.org/about/licenses/ (last visited Feb. 23, 2009). Open licenses may not be "open" for purposes of the OSI definition, but are certainly a form of open source licenses for purposes of this article.

[68] *See*, *e.g.*, Law.com License, License http://dictionary.law.com/default2.asp?selected=1156& bold (last visited Feb. 23, 2009).

[69] *See* Specht v. Netscape Commc'ns Corp., 306 F.3d 17, 22 (2d Cir. 2002).

[70] *Id.*

when a purchaser of media on which software code is licensed must physically open the packaging—typically shrink-wrap packaging—that encases the media and on which is printed the terms of the license. For example, a shrink-wrap license may state, "By opening this package you agree to be bound by the license terms hereon. If you do not wish to so agree, do not open this package and return it to the place of purchase for a full refund." Lastly, for large corporate or enterprise software transactions that might contemplate millions of dollars in license fees, it is typical to negotiate and execute lengthy physical contracts that contain detailed provisions governing use of the licensed code.

Regardless of the method used, the key point to remember is that each is a merely a mechanism by which the copyright owners place limitations on the downstream end user's ability to utilize the software code under 17 U.S.C. § 106.[71] In the case of open source licensing, it is very likely that the contractual limitations, if any, will be quite broad and will permit copying, modification, publication, redistribution, and the creation of derivative works, all as long as the contractual covenants of the license are observed by the licensee. Make no mistake, most licensors of open source software do not view their product as being in the public domain and will actively assert their copyrights through actions in federal court for copyright infringement. The recent *Katzer* case[72] is an excellent example.

C. The Katzer Case

One of the most recent and highly publicized cases dealing with open source code is *Jacobsen v. Katzer*.[73] The Federal Circuit's August 13, 2008, decision clarifies many issues regarding licensing, copyrights, and contract law as they impact open source software. The introductory paragraphs of the opinion sets the stage well:

> Appellant Robert Jacobsen ("Jacobsen") appeals from an order denying a motion for preliminary injunction. . . . Jacobsen holds a copyright to computer programming code. He makes that code available for public download from a website without a financial fee pursuant to the Artistic License, an "open source" or public license. Appellees Matthew Katzer and Kamind Associates, Inc. (collectively "Katzer/Kamind") develop commercial software products for the model train industry and hobbyists. Jacobsen accused Katzer/Kamind of copying certain materials from Jacobsen's website and incorporating them into one of Katzer/Kamind's software packages without following the terms of the Artistic License. Jacobsen brought an action for copyright infringement and moved for a preliminary injunction.

[71] 17 U.S.C. § 106 (2007).

[72] Jacobsen v. Katzer, 535 F.3d 1373, 1375–76 (Fed. Cir. 2008).

[73] *Id.*

> . . . .
>
> Jacobsen manage[d] an open source software group called Java Model Railroad Interface ("JMRI"). Through the collective work of many participants, JMRI created a computer programming application called DecoderPro, which allows model railroad enthusiasts to use their computers to program the decoder chips that control model trains. DecoderPro files are available for download and use by the public free of charge from an open source incubator website called SourceForge; Jacobsen maintains the JMRI site on SourceForge. The downloadable files contain copyright notices and refer the user to a "COPYING" file, which clearly sets forth the terms of the Artistic License.
>
> Katzer/Kamind offers a competing software product, Decoder Commander, which is also used to program decoder chips. During development of Decoder Commander, one of Katzer/Kamind's predecessors or employees is alleged to have downloaded the decoder definition files from DecoderPro and used portions of these files as part of the Decoder Commander software. The Decoder Commander software files that used DecoderPro definition files did not comply with the terms of the Artistic License. Specifically, the Decoder Commander software did not include (1) the author' [sic] names, (2) JMRI copyright notices, (3) references to the COPYING file, (4) an identification of SourceForge or JMRI as the original source of the definition files, and (5) a description of how the files or computer code had been changed from the original source code. The Decoder Commander software also changed various computer file names of DecoderPro files without providing a reference to the original JMRI files or information on where to get the Standard Version.
>
> Jacobsen moved for a preliminary injunction, arguing that the violation of the terms of the Artistic License constituted copyright infringement and that, under Ninth Circuit law, irreparable harm could be presumed in a copyright infringement case. The District Court reviewed the Artistic License and determined that "Defendants' alleged violation of the conditions of the license may have constituted a breach of the nonexclusive license, but does not create liability for copyright infringement where it would not otherwise exist." The District Court found that Jacobsen had a cause of action only for breach of contract, rather than an action for copyright infringement based on a breach of the conditions of the Artistic License. Because a breach

> of contract creates no presumption of irreparable harm, the District Court denied the motion for a preliminary injunction.[74]

Jacobsen appealed the finding that he did not have a cause of action for copyright infringement.[75] On appeal, the Federal Circuit made several interesting and relevant pronouncements. First, the court made observations relative to open source licensing in general:

> Public licenses, often referred to as "open source" licenses, are used by artists, authors, educators, software developers, and scientists who wish to create collaborative projects and to dedicate certain works to the public. Several types of public licenses have been designed to provide creators of copyrighted materials a means to protect and control their copyrights. Creative Commons, one of the amici curiae, provides free copyright licenses to allow parties to dedicate their works to the public or to license certain uses of their works while keeping some rights reserved.
>
> Open source licensing has become a widely used method of creative collaboration that serves to advance the arts and sciences in a manner and at a pace that few could have imagined just a few decades ago. For example, the Massachusetts Institute of Technology ("MIT") uses a Creative Commons public license for an OpenCourseWare project that licenses all 1800 MIT courses. Other public licenses support the GNU/Linux operating system, the Perl programming language, the Apache web server programs, the Firefox web browser, and a collaborative web-based encyclopedia called Wikipedia. Creative Commons notes that, by some estimates, there are close to 100,000,000 works licensed under various Creative Commons licenses. The Wikimedia Foundation, another of the amici curiae, estimates that the Wikipedia website has more than 75,000 active contributors working on some 9,000,000 articles in more than 250 languages.[76]

Next, the court opined on the issue of whether or not the fact Katzer's license was royalty free impacted the analysis and concluded that it did not:

> Traditionally, copyright owners sold their copyrighted material in exchange for money. The lack of money changing hands in open source licensing should not be presumed to mean that there is no economic consideration, however. There are substantial benefits, including economic benefits, to the creation and distribution of copyrighted works under public licenses that

[74] *Id.* at 1376–77 (citations omitted) (footnotes omitted).
[75] *Id.* at 1377.
[76] *Id.* at 1378.

> range far beyond traditional license royalties. For example, program creators may generate market share for their programs by providing certain components free of charge. Similarly, a programmer or company may increase its national or international reputation by incubating open source projects. Improvement to a product can come rapidly and free of charge from an expert not even known to the copyright holder. The Eleventh Circuit has recognized the economic motives inherent in public licenses, even where profit is not immediate. *See Planetary Motion, Inc. v. Techsplosion, Inc.*, 261 F.3d 1188, 1200 (11th Cir. 2001) (Program creator "derived value from the distribution [under a public license] because he was able to improve his Software based on suggestions sent by end-users. . . . It is logical that as the Software improved, more end-users used his Software, thereby increasing [the programmer's] recognition in his profession and the likelihood that the Software would be improved even further.").[77]

The core issue in *Katzer* was whether the use by Katzer/Kamind was outside the scope of the Artistic License, given that Jacobsen had made out a prima facie case of copyright infringement, and Katzer/Kamind had argued that they could not be liable for copyright infringement because they had a license to use the material.[78] With respect to the Artistic License, the court noted that "[t]he copyrighted materials in this case are downloadable by any user and are labeled to include a copyright notification and a COPYING file that includes the text of the Artistic License. The Artistic License grants users the right to copy, modify, and distribute the software",[79] subject to certain specific conditions, including significantly the obligation that "changes to the computer code be tracked so that downstream users know what part of the computer code is the original code created by the copyright holder and what part has been newly added or altered by another collaborator."[80]

In reversing the District Court and finding that the Artistic License was enforceable as a means of protecting Jacobsen's copyrights in the open source code, the Federal Circuit held:

> The conditions set forth in the Artistic License are vital to enable the copyright holder to retain the ability to benefit from the work of downstream users. By requiring that users who modify or distribute the copyrighted material retain the reference to the original source files, downstream users are directed to Jacobsen's website. Thus, downstream users know about the

[77] *Id.* at 1379 (alteration in original).

[78] *Id.*

[79] *Id.* at 1379–80.

[80] *Id.* at 1379.

> collaborative effort to improve and expand the SourceForge project once they learn of the "upstream" project from a "downstream" distribution, and they may join in that effort.
>
> . . . .
>
> The copyright holder here expressly stated the terms upon which the right to modify and distribute the material depended and invited direct contact if a downloader wished to negotiate other terms. These restrictions were both clear and necessary to accomplish the objectives of the open source licensing collaboration, including economic benefit.[81]

The court concluded with a clear pronouncement affirming the rights of the licensors of open source software:

> Copyright holders who engage in open source licensing have the right to control the modification and distribution of copyrighted material.
>
> The Artistic License, like many other common copyright licenses, requires that any copies that are distributed contain the copyright notices and the COPYING file.
>
> It is outside the scope of the Artistic License to modify and distribute the copyrighted materials without copyright notices and a tracking of modifications from the original computer files. If a down loader [sic] does not assent to these conditions stated in the COPYING file, he is instructed to "make other arrangements with the Copyright Holder." Katzer/Kamind did not make any such "other arrangements."[82]

The *Katzer* case was much ballyhooed in the press as "a victory for supporters of free software,"[83] but there had never been a formal pronouncement before *Katzer* by anyone conversant with the open source movement that open source software was anything other than copyrightable subject matter that was able to be licensed successfully. Thus, many were left wondering what it was *Katzer* actually did, except to perhaps formally clarify that "[c]opyright holders who engage in open source licensing have the right to control the modification and distribution of copyrighted material."[84] But that is a truism that did not really need clarifying, for it had never been suggested anywhere that such was not the case. Nonetheless, *Katzer* will likely serve to embolden licensors of open source software and cause them to feel empowered that their

[81] *Id.* at 1381.

[82] *Id.* at 1381–82 (citations omitted).

[83] John Markoff, *Ruling Is a Victory for Supporters of Free Software*, N.Y. TIMES, Aug. 13, 2008, at C7, *available at* http://www.nytimes.com/2008/08/14/technology/14commons.html.

[84] *Katzer*, 535 F.3d at 1381.

licenses, if written within the framework approved by *Katzer*, will be a viable means of enforcing their copyrights.

D. The GNU General Public License Version Three

On June 29, 2007, the FSF released version three of the GNU General Public License (the GNU GPL). The FSF offered this brief historical overview:

> The GNU GPL is the most widely used free software license worldwide: almost three quarters of all free software packages are distributed under this license. It is not, however, the only free software license.
>
> Richard Stallman wrote the version 1 and 2 of the GNU GPL . . . . Version 1 was released in 1989, and version 2 in 1991. Since 1991, free software use has increased tremendously, and computing practices have changed, introducing new opportunities and new threats. In 2005, Stallman began revising the GPL for version 3.[85]

The GNU GPL is significant, then, in terms of the breadth of its use and distribution. An examination of certain key provisions will thus prove profitable in understanding critical aspects of open source or, as FSF prefers, "free," software licensing. For example, the GNU GPL Preamble stated:

> The GNU General Public License is a free, copyleft[86] license for software and other kinds of works.
>
> The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.
>
> When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or

[85] FSF.org, *FSF releases the GNU General Public License, Version 3,* http://www.fsf.org/news/gplv3_launched (last visited Feb. 23, 2009).

[86] "Copyleft is a play on the word copyright to describe the practice of using copyright law to remove restrictions on distributing copies and modified versions of a work for others and requiring that the same freedoms be preserved in modified versions." Wikipedia.org, Copyleft, http://en.wikipedia.org/wiki/Copyleft (last visited Feb. 23, 2009).

> can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.
>
> To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.
>
> For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.[87]

Similarly, the following provision from section three of the GNU GPL contractually avoids the anticircumvention provisions of applicable copyright laws:

> When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.[88]

These excerpts make it clear that when an end user licenses software under the GNU GPL, that end user had best be aware of the aggressive copyleft restrictions the license places on use of the licensed code and the subsequent downstream licensing of any modified or derivative works based on the code licensed under GNU GPL. This is especially true if any downstream licensee asks the licensor to warrant that the licensor has sole title to the licensed code. Any licensor in that situation must consult the terms of the upstream open source license implicated by any such request for a warranty of title.

In contrast to the GNU GPL's "open" philosophy, consider this language from the Microsoft Windows XP Home Edition End User License Agreement published June 1, 2004:

> 1.1 INSTALLATION AND USE. You may install, use, access, display and run one copy of the Software on a single computer, such as a workstation, terminal or other device ("Workstation

[87] GNU.org, GNU General Public License Version 3, 29 June 2007, http://www.gnu.org/licenses/gpl.html (last visited Feb. 23, 2009).

[88] *Id.*

> Computer”). The Software may not be used by more than one processor at any one time on any single Workstation Computer.
>
> . . . .
>
> 3. RESERVATION OF RIGHTS AND OWNERSHIP. Microsoft reserves all rights not expressly granted to you in this EULA. The Software is protected by copyright and other intellectual property laws and treaties. Microsoft or its suppliers own the title, copyright, and other intellectual property rights in the Software. The Software is licensed, not sold.
>
> 4. LIMITATIONS ON REVERSE ENGINEERING, DECOMPILATION, AND DISASSEMBLY. You may not reverse engineer, decompile, or disassemble the Software, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation. [89]

The conceptual differences are indeed stark, but do not let the copyleft philosophy of the GNU GPL suggest that the FSF is lax about enforcing its copyrights. To the contrary, on December 11, 2008, the FSF announced that it had filed a copyright infringement lawsuit against Cisco.[90] The FSF press release provides further details:

> The FSF’s complaint alleges that in the course of distributing various products under the Linksys brand Cisco has violated the licenses of many programs on which the FSF holds copyright, including GCC, binutils, and the GNU C Library. In doing so, Cisco has denied its users their right to share and modify the software.
>
> Most of these programs are licensed under the GNU General Public License (GPL), and the rest are under the GNU Lesser General Public License (LGPL). Both these licenses encourage everyone, including companies like Cisco, to modify the software as they see fit and then share it with others, under certain conditions. One of those conditions says that anyone who redistributes the software must also provide their recipients with the source code to that program. The FSF has documented many instances where Cisco has distributed licensed software but failed to provide its customers with the corresponding source code.[91]

[89] Microsoft.com, Microsoft Windows XP Home Edition (Retail) End-User License Agreement for Microsoft Software, http://www.microsoft.com/windowsxp/eula/home.mspx (last visited Feb. 23, 2009).

[90] FSF.org, Free Software Foundation Files Suit Against Cisco for GPL Violations, http://www.fsf.org/news/2008-12-cisco-suit (last visited Feb. 23, 2009).

[91] *Id.*

Such suits evidence that "open" source does not mean "in the public domain" and support the premise that open source is a licensing philosophy as opposed to an abandonment of any rights.

IV. BUSINESS ISSUES

The foregoing discussion illustrates that open source software creates usage implications for both licensors and licensees, both from a copyright and a contract perspective. For example, in the areas of mergers, acquisitions, due diligence, business planning, and product engineering, the presence of open source software and open source licenses creates a variety of business issues.

A. Mergers, Acquisitions, and Due Diligence

In any setting in which one business is considering merging with or acquiring another, robust due diligence mandates that all software and other intellectual property (IP) licenses (click-wrap, browse-wrap, shrink-wrap and paper) be gathered and reviewed by the acquirer with an eye toward determining, for example, contract duration, royalty fees due, license restrictions, assignability, and indemnification provisions. Moreover, to the extent any such licenses cover open source software, the acquirer must determine if that impacts the value of the underlying transaction. For example, if a large percentage of the negotiated purchase price for a business derives from the seller's representation that it has good, sole, and marketable title to all the IP that the seller used in developing its products and business infrastructure, and the buyer determines that a certain amount of that infrastructure or a certain amount of those products were built instead on open source platforms and licenses, that may impact the buyer's ability to extract full value for the acquired assets downstream because that the buyer may adversely affect royalties it may reasonably charge to its future downstream end users. Furthermore, it may also accordingly diminish the purchase price that the buyer is willing to pay.

B. Business Planning

As businesses make adjustments in their information technology business planning, in addition to considerations about new hardware purchases and implementations, consideration must be given to what types of new software will be purchased and implemented. For example, if an enterprise is considering purchasing several hundred new desktop computers, a question arises as to whether it is advisable to have them shipped from the manufacturer with an open source operating system preinstalled,[92] as opposed to the default Microsoft Windows operating

[92] For example, the Vietnamese Ministry of Information and Communications has issued an instruction on using open source software products at state agencies. Accordingly,

system.[93] If so, will the enterprise's business end users be able to adapt to an open source operating system environment, which may be different from Microsoft's or less feature-rich? Open source implementations typically portend great cost savings, but businesses must carefully plan in order to ensure a smooth transition.

C. Product Engineering

As new products are developed, businesses must consider whether any of them have been developed on open source platforms of any kind, as this will impact whether the business may treat the new product as fully proprietary. For example, if a software company is rolling out a new video game and certain aspects of the code have been built on an open source software platform, the company may be precluded from registering its own copyrights in the full code or otherwise placing license restrictions on its downstream end users' use of or modifications to that software.

V. CONCLUSION

As more and more of the world runs on software via desktop applications, server applications, enterprise operations, and the Internet, the philosophical and practical divide between open (collaborative) and proprietary (closed) software and licenses will likely become greater. Microsoft has done an excellent job of marketing and gaining consumer acceptance of a closed philosophy, touting things such as maintenance, technical support, service, upgrades, the latest technology, and feature-rich environments—but at a high retail price. Conversely, open source typically comes with a very moderate price tag (in accordance with its collaborative philosophy), but consumers and business customers might be expected to make certain sacrifices in terms of technical support, customer service, and features. Current trends suggest this is not an insurmountable impediment to acceptance; business analyst Gartner Group recently reported that eighty-five percent of companies are now

by June 30, 2009, 100% of servers of IT divisions of government agencies must be installed with open source software. One-hundred percent of staffs at these IT divisions must be trained in the use of these software products and at least 50% must be able to use them proficiently. *See* VietNamNet.com, *Vietnam to Widely Use Open-Source Software* (Jan. 6, 2009), http://english.vietnamnet.vn/tech/2009/01/822425/.

[93] Recently, an Israeli man successfully obtained a refund for the price of the Windows Vista operating system that came bundled with his Dell laptop, based on terms in the Vista end-user license agreement, which instructed users to return the software for a refund if they did not agree to its terms. *Israeli Linux User Gets $137 Refund for Unused Windows Software*, [Jan. 7, 2009] Andrews Litig. Reporter: Computer & Internet (West) at 4.

using open source in some fashion.[94] If the trend continues, businesses will be well advised to be familiar with the license terms that accompany proffered open source software, abide scrupulously by those terms, and not be lulled into accepting the conventional wisdom that open source software falls outside the realm of copyright law, which, as explained above, is certainly not the case.

[94] David Meyer, *Gartner: 85 Percent of Companies Using Open Source*, CNET.COM, Nov. 17, 2008, http://news.cnet.com/8301-1001_3-10098624-92.html (last visited Feb. 23, 2009).