

# LICENSE AND REGISTRATION, PLEASE: USING COPYRIGHT “CONDITIONS” TO PROTECT FREE/OPEN SOURCE SOFTWARE

VICTORIA NEMIAH\*

*FOSS (free/open source software) is a growing player in the end-user market, as evidenced by the popularity of everything from Wordpress to Firefox. One of its key appeals to developers is the egalitarian nature of its “access for everyone” model, which allows everyday users to develop and share tweaks that make the program more useful to them. In executing these goals, however, developers can often get lost in the world of legalese surrounding available FOSS licenses. Often, developers choose a license without a full understanding of its contents, or even attempt to draft their own. One of the biggest risks in the wording of a license is that of the “condition”/“covenant” dichotomy. Incorrectly worded, a covenant-based license runs the risk of foreclosing copyright-based remedies and limiting the scope of arguments available when a user breaches a program’s license. In approaching the choice of which license to apply to a new program, then, both developers and their attorneys should be aware of the critical importance of its choice of words.*

INTRODUCTION.....	359
I. HISTORY OF FOSS .....	361
A. <i>First Applications of Open Source Licenses</i> .....	361
B. <i>Subsequent Growth of the FOSS Model</i> .....	365
C. <i>Development of the FOSS Community</i> .....	367
II. IMPORTANT FOSS CASES .....	368
A. <i>Graham v. James</i> .....	369
B. <i>Sun v. Microsoft</i> .....	373
C. <i>Jacobsen v. Katzer</i> .....	376
D. <i>MDY v. Blizzard</i> .....	381
III. DRAFTING AND ARGUING FOR FOSS LICENSES .....	385
A. <i>First Considerations</i> .....	385
B. <i>Best Practices</i> .....	387
C. <i>Arguing a License Violation</i> .....	388

---

\* Victoria is a 2014 J.D. candidate at NYU School of Law. Her professional interests are in the fields of software licensing and software-as-a-service agreements for Web 2.0 companies. She received a B.A. in Sociology from Boston University in 2011. She would like to thank Christopher Sprigman, the JIPEL Editorial Staff, and her family for their help throughout the editing process.

CONCLUSION.....	389
-----------------	-----

## INTRODUCTION

There are three types of people who should care about free and open-source software (FOSS<sup>1</sup>): FOSS software developers, the attorneys who help them license their creations, and absolutely everyone else. Why everyone else? While FOSS has historically been billed as “too difficult” for the average consumer to use and “too technical to understand,”<sup>2</sup> it is fast becoming a popular option, even for technophobes, as evidenced by Firefox’s position as the second most popular browser by market share.<sup>3</sup> FOSS has also started to gain a foothold in other areas, from blog publishing platforms to image-editing software. Consumers are catching onto FOSS offerings like OpenOffice, a word-processing and productivity suite; GIMP, an image-editing Photoshop alternative; Linux, a completely open-source operating system; VLC Media Player, a cross-platform multimedia player; and 7-Zip, a file archiver.

This Note is primarily targeted towards the first two categories of people: developers and their attorneys. FOSS has been a novel and exciting model for developers. It provides several key advantages over the traditional route for those looking to release software they’ve finished (or even just started) writing: once the source code has been publicly released, other developers who choose to use the software themselves can contribute fixes and improvements back to the project.

---

<sup>1</sup> There are a myriad of terms used to refer to software that has any of the qualities of being made collaboratively, being shared freely, or being shared pursuant to a promise to “share alike” (to use an identical license on derivative works). Since this paper is not meant to address the intricacies or politics of the software community’s preferences for different terminology (though I do cover the issue briefly in Part II.C.), I choose to refer to these forms of software simply as “FOSS”, for both clarity and space’s sake. I would include in my “FOSS” umbrella software termed “free software”, “open source software”, “free/open source software” (“FOSS”), or “free/libre/open-source software” (“FLOSS”), and I invite the reader to impose upon my use of the term any definition that helps make the paper relevant to his or her projects.

<sup>2</sup> Matt Kohut, *Lenovo’s Worldwide Competitive Analyst*, argued, “Linux, even if you’ve got a great distribution and you can argue which one is better or not, still requires a lot more hands-on than somebody who is using Windows . . . . You have to know how to decompile codes and upload data, stuff that the average person, well, they just want a computer.” John Pospisil, *Lenovo analyst: Linux on netbooks is doomed*, TECH.BLORGE (Apr. 21, 2009), <http://tech.blorge.com/Structure:%20/2009/04/21/lenovo-analyst-linux-on-netbooks-is-doomed>.

<sup>3</sup> *Top Browser Share Trend: June, 2010 to April, 2011*, NETMARKETSHARE, <http://www.netmarketshare.com/report.aspx?qprid=1&qpct=2&qptimeframe=M&qpsp=137&qnp=11> (last visited Feb. 28, 2014).

People have become accustomed to finding free options for most of their computing needs, so no-fee software is likely to gain a foothold in these markets. The project is more likely to survive where outsiders are actively participating in its continued maintenance. Where there is a disagreement on what slate of features or overall design approach to choose, forking<sup>4</sup> can occur. Opening the source code provides the opportunity for volunteer programming support, which can take the place of hiring paid developers (though one blogger notes that choosing to “open” one’s source code is akin to opening a door, and that developers should not assume that volunteers will be waiting on the other side of the door to come in and help with the project<sup>5</sup>). Making a project open source allows for the use of previously released FOSS or open source code snippets within the project. Lastly, releasing a project’s code may be aligned with the developer’s personal convictions that his software should be free and available to all.

With so many advantages to making a project open source, it has become an easy choice for many developers. The legal process that follows, however, has not been nearly as simple. As a general matter, the open source model is one that stymies many of contract law’s core tenets, from the “exchange of promises” to the issue of consideration. Further, open source licensing serves as a voluntary renunciation of several rights the copyright system grants to software creators. Lawyers and developers have made many attempts to word a license so as to restrict certain behavior while allowing otherwise unrestricted distribution of the software, but to date there have been few cases on the subject, so much of this wording is untested.

In fact, while the first FOSS licenses were created in the late 1980s (with the GPL, MIT License, and BSD license all published between 1988 and 1989<sup>6</sup>), it was not until 2004 that a court issued an opinion regarding the viability of FOSS licenses. In a federal case in the Northern District of Illinois, the court explained the GNU General Public License (“GPL”) as a release of the content into the public domain, with restrictions in place solely because “once the programs are

---

<sup>4</sup> See generally *Fork (software development)*, WIKIPEDIA, [http://en.wikipedia.org/wiki/Fork\\_\(software\\_development\)](http://en.wikipedia.org/wiki/Fork_(software_development)) (last visited Feb. 13, 2014).

<sup>5</sup> Eric Wilhelm, *Open Source Benefits for Developers*, SCRATCHCOMPUTING.COM, [http://scratchcomputing.com/articles/open-source\\_benefits\\_developers.html](http://scratchcomputing.com/articles/open-source_benefits_developers.html) (last visited Feb. 28, 2014).

<sup>6</sup> GOTTFRIED HOFMANN ET AL., A DUAL MODEL OF OPEN SOURCE LICENSE GROWTH 2 (2013), available at <http://dirkriehle.com/wp-content/uploads/2013/04/oss2013.hofmann.pdf>.

freely released into the public domain, the creators intend for them to stay free.”<sup>7</sup> Fortunately for embracers of FOSS, judicial attitudes towards FOSS have changed significantly since this opinion, and “the court's comments should be viewed with skepticism.”<sup>8</sup>

A few relevant cases in the last five years help to clarify the approach lawyers and others working with licensing language should take. I use these as the backbone of this paper to ensure that their takeaways are fully incorporated into licensors’ approaches. In particular, I highlight the construction of a license provision as either a “condition” or a “covenant,” a designation that will determine what remedies are appropriate, an issue recently brought to light in a non-FOSS case in the Ninth Circuit.<sup>9</sup> Making sure that attorneys understand the state of the law surrounding FOSS licenses is important because a future court decision against a FOSS license has the potential to cripple a burgeoning industry whose community-based development principles depend on protection from commercial exploitation of that openness. Thus, I start in Part II by examining the history of the development of the FOSS community. In Part III, I break down several cases relevant to issues of FOSS licensing, highlighting important conclusions by courts that are buried within the sometimes dense language of the opinions. Finally, in Part IV, I summarize several best practices for lawyers and other license drafters. Additionally, I provide an analysis of legal arguments regarding license violations, based on criteria such as copyright registration and wording of license terms. By doing so, I highlight the importance for developers of taking care to follow all the necessary steps to protect their projects, including harnessing the power of conditions.

## II HISTORY OF FOSS

### *A. First Applications of Open Source Licenses*

Software developers have been sharing code with each other for as long as software development has been a pursuit, but it was not until at least 1969 that the beginnings of what would become the FOSS movement started to emerge. In 1965,

---

<sup>7</sup> *Computer Associates Int'l v. Quest Software, Inc.*, 333 F. Supp. 2d 688, 698 (N.D. Ill. 2004). The court refers to GPL’s requirement that subsequent derivative works based on a GPL-licensed program must then themselves be GPL-licensed.

<sup>8</sup> Heather J. Meeker, *Open Source and the Age of Enforcement*, 4 HASTINGS SCI. & TECH. L.J. 267, 270 (2012).

<sup>9</sup> *MDY Indus., LLC v. Blizzard Entm't, Inc.*, 629 F.3d 928 (9th Cir. 2010).

Bell Labs joined forces with the Massachusetts Institute of Technology (MIT) and General Electric (GE) to work on the development of MULTICS (the MULTiplexed Information and Computing Service), which was intended to develop a “dynamic, modular [computer] system capable of supporting hundreds of users.”<sup>10</sup> By 1969, however, Bell Labs, displeased by the project’s lack of advances and continued drain on monetary capital, pulled out of the MULTICS project.<sup>11</sup> Bell Labs “had a million dollars worth of equipment in the attic that was sitting there being played with by three people. It became clear that we were a drag on the computer center's budget.”<sup>12</sup>

This abandonment of the MULTICS project did not quash the interest of the Bell Labs researchers in the construction of such a system, so four of the most heavily involved researchers in the Lab’s participation started to work towards the construction of a different system.<sup>13</sup> One of the four has explained that their motivation in working on this new system was simple; they did not want to give up the computing power they had personally been able to exploit via the MULTICS machine:

We didn't want to lose the pleasant niche we occupied, because no similar ones were available; even the time-sharing service that would later be offered under GE's operating system did not exist. What we wanted to preserve was not just a good environment in which to do programming, but a system around which a fellowship could form.<sup>14</sup>

The group lobbied hard for the purchase of a new machine.<sup>15</sup> By 1970, they had put together a pitch based on a proposed new operating system and Bell Labs agreed to purchase a new machine for about \$65,000.<sup>16</sup> A year later, they had the

---

<sup>10</sup> Gordon M. Brown, *Unix: An Oral History*, PRINCETON U., <http://www.princeton.edu/~hos/frs122/unixhist/finalhis.htm> (last visited Feb. 28, 2014).

<sup>11</sup> *Id.*

<sup>12</sup> Brown, *supra* note 10.

<sup>13</sup> *Id.*

<sup>14</sup> Dennis M. Ritchie, *The Evolution of the Unix Time-sharing System*, BELL LABORATORIES, <http://cm.bell-labs.com/cm/cs/who/dmr/hist.html> (last visited Feb. 28, 2014).

<sup>15</sup> Brown, *supra* note 10.

<sup>16</sup> Ritchie, *supra* note 14.

machine running their new operating system, which they called UNIX.<sup>17</sup> Due to a consent order signed by AT&T in 1956, Bell Labs could not legally sell any products on the software market, and thus agreed to share UNIX (including its source code) with academic institutions for the price of media.<sup>18</sup>

Ken Thompson, one of the original authors of the UNIX system, returned in 1976 to teach UNIX in the Computer Science department at his alma mater, the University of California at Berkeley (UCB).<sup>19</sup> His classes, coupled with further work by students, led to a system called the Berkeley Standard Distribution (BSD), which is considered by some to be UNIX's first fork.<sup>20</sup> A graduate student named Bill Joy spearheaded the work, and in 1978 the first BSD was released to the public.<sup>21</sup> In his history of BSD, Bruce Montague explains, "UCB made BSD available for the cost of media, under what became known as 'the BSD license'. A customer purchased Unix from AT&T and then ordered a BSD tape from UCB."<sup>22</sup> As the cost of this license from AT&T continued to increase, users started to request that Berkeley break out their own code and provide it under terms that did not require concurrent acquisition of an AT&T source license.<sup>23</sup> In 1989, Berkeley responded to these requests by releasing Networking Release 1, the first freely distributable Berkeley code. The terms of the license for this release were liberal:

A licensee could release the code modified or unmodified in source or binary form with no accounting or royalties to Berkeley. The only requirements were that the copyright notices in the source file be left intact and that products that incorporated the code indicate in their documentation that the product contained code from the University of California and its contributors. . . . [A]nyone was free to get a copy

---

<sup>17</sup> *Id.* The name "UNIX" was (an altered spelling of "UNICS", which was a pun on MULTICS). *Id.*

<sup>18</sup> Bruce Montague, *Unix from a BSD Licensing Perspective*, FREEBSD, <http://www.freebsd.org/doc/en/articles/bsd-gpl/unix-license.html> (last visited Mar. 21, 2014).

<sup>19</sup> *Id.* See also *UNIX/Linux History*, DIGITAL-DOMAIN, <http://digital-domain.net/lug/unix-linux-history.html> (last visited Feb. 28, 2014).

<sup>20</sup> See sources cited *supra* note 19. See generally WIKIPEDIA, *supra* note 4 (for an explanation on forking).

<sup>21</sup> Peter H. Salus, A Quarter Century of UNIX 142 (1994).

<sup>22</sup> Montague, *supra* note 18.

<sup>23</sup> Marshall Kirk McKusick, *Twenty Years of Berkeley Unix*, in *OPEN SOURCES: VOICES FROM THE OPEN SOURCE REVOLUTION* 21, 25 (Chris DiBona & Sam Ockman eds., 1999).

from anyone who already had received it. Indeed, several large sites put it up for anonymous ftp shortly after it was released.<sup>24</sup>

This BSD release and accompanying license formed the basis for today's BSD License, which remains one of the most popular "academic" open source licenses in use.<sup>25</sup>

In 1984, a researcher at MIT's Artificial Intelligence Lab named Richard Stallman quit his job to start writing his own software, determined that unlike UNIX and other fully contained options available at the time, his would be open to all to freely use and modify.<sup>26</sup> Due to drama surrounding the commercialization of a derivative of one of his first programs,<sup>27</sup> Stallman was intent upon ensuring that his newest program, a text editor GNU Emacs, would be perpetually available to the public, and thus decided to use a "copyleft" license that he called the Emacs General Public License.<sup>28</sup>

The license's preamble explains that the software is not in the public domain, but that the restrictions placed on it are "designed to permit everything that a good cooperating citizen would want to do".<sup>29</sup>

Specifically, we want to make sure that you have the right to give away copies of Emacs, that you receive source code or else can get it if you want it, that you can change Emacs or use pieces of it in new free programs, and that you know you can do these things. . . . To

---

<sup>24</sup> *Id.*

<sup>25</sup> Bennett M. Sigmond, *Free/open Source Software Licensing-Too Big to Ignore*, COLO. LAW., Dec. 2005, at 89, 90.

<sup>26</sup> Richard Stallman, *The GNU Project*, GNU.ORG, <http://www.gnu.org/gnu/thegnuproject.html> (last visited Feb. 28, 2014).

<sup>27</sup> Stallman's legal battle with James Gosling, creator of a C-based version of Stallman's Emacs program, is a highly entertaining read. It is detailed in an article by Li-Cheng Tai, and vivified by comments given by Stallman at a lecture in Sweden. Li-Cheng Tai, *The History of the GPL*, FREE-SOFT.ORG (July 4, 2001), [http://www.free-soft.org/gpl\\_history](http://www.free-soft.org/gpl_history); Richard Stallman, *RMS Lecture at KTH (Sweden), 30 October 1986*, GNU.ORG, <http://www.gnu.org/philosophy/stallman-kth.html> (last visited Feb. 28, 2014).

<sup>28</sup> Tai, *supra* note 27.

<sup>29</sup> *Emacs General Public License*, FREE-SOFT.ORG, [http://www.free-soft.org/gpl\\_history/emacs\\_gpl.html](http://www.free-soft.org/gpl_history/emacs_gpl.html) (last visited Feb. 28, 2014).

make sure that everyone has such rights, we have to forbid you to deprive anyone else of these rights. For example, if you distribute copies of Emacs, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.<sup>30</sup>

While distributing GNU Emacs, Stallman continued working on building an operating system. The result was a free operating system that was compatible with UNIX, which Stallman named “GNU” (a recursive acronym for “GNU's Not Unix”).<sup>31</sup> Those developing GNU decided to share it under the principle of “copyleft” touched on above.<sup>32</sup> In an effort to encourage others to follow his lead, Stallman started the Free Software Foundation, a group that now publishes the second version of the GNU GPL, currently the most popular open source software license available.<sup>33</sup>

### *B. Subsequent Growth of the FOSS Model*

Since the creation and success of FOSS like BSD and GNU,<sup>34</sup> such software has gone on to see remarkable success. As mentioned above, many popular software options today are FOSS, from Firefox to Wordpress. Contributions to FOSS have grown exponentially over the last fifteen to twenty years, as shown in a chart by researchers Amit Deshpande & Dirk Riehle.<sup>35</sup>

---

<sup>30</sup> *Id.*

<sup>31</sup> Stallman, *supra* note 26.

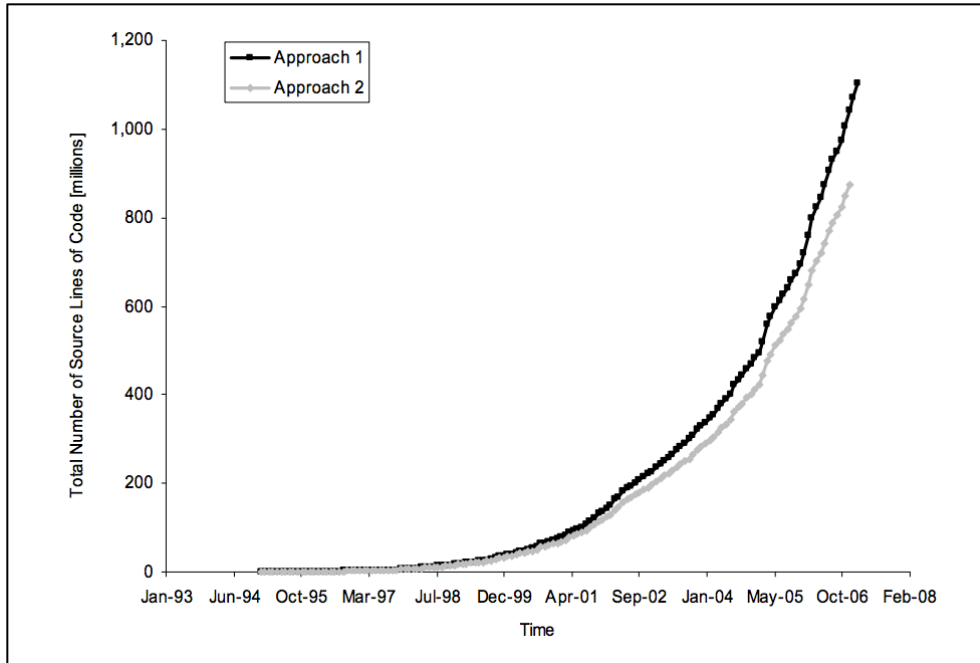
<sup>32</sup> 1 GNU'S BULLETIN, no. 5, June 1988, <http://www.gnu.org/bulletins/bull5.html>.

<sup>33</sup> Sigmond, *supra* note 25, at 90.

<sup>34</sup> GNU has gone on to become one of the two contributions that make up the Linux operating system.

<sup>35</sup> Amit Deshpande & Dirk Riehle, *The Total Growth of Open Source*, in OPEN SOURCE DEVELOPMENT, COMMUNITIES AND QUALITY 197, 202 (Barbara Russo et al. eds., 2008).





One of the many forks<sup>36</sup> of Linux, the commercial venture RedHat, saw its revenues grow by 25% in 2012, finally crossing the \$1 billion mark at the end of its fiscal year on February 29 of that year, which made it the first FOSS company to reach that point.<sup>37</sup> In the market for web servers, FOSS favorite Apache holds 54.37% of the market share, with its next closest competitor holding only 12.91%.<sup>38</sup> Most promising, however, are reports of the public's increased acceptance of the FOSS model. A 2010 survey of three hundred large organizations in the public and private sectors found that 50% of respondents are "fully committed to open source in their business", while another 28% say they are "experimenting with open source and keeping an open mind to using it."<sup>39</sup>

<sup>36</sup> See *Linux Distribution*, WIKIPEDIA, [http://en.wikipedia.org/wiki/Linux\\_distribution](http://en.wikipedia.org/wiki/Linux_distribution) (for a beautiful graphic illustration of the various forked distributions of Linux) (last visited Feb. 17, 2014).

<sup>37</sup> Charles Babcock, *Red Hat: First \$1 Billion Open Source Company*, INFORMATIONWEEK (Mar. 29, 2012, 9:49 AM), <http://www.informationweek.com/development/open-source/red-hat-first-1-billion-open-source-comp/232700454>.

<sup>38</sup> *May 2013 Web Server Survey*, NETCRAFT (May 3, 2013), <http://news.netcraft.com/archives/2013/05/03/may-2013-web-server-survey.html>.

<sup>39</sup> *Investment in Open Source Software Set to Rise, Accenture Survey Finds*, ACCENTURE (Aug. 5, 2010), [http://newsroom.accenture.com/article\\_display.cfm?article\\_id=5045](http://newsroom.accenture.com/article_display.cfm?article_id=5045).

### *C. Development of the FOSS Community*

Of course, along with any movement espousing such a political message (that software should be fundamentally free) comes spirited debate. Within the FOSS community, this has mainly taken the form of differing definitions of, and terms for, “free” sharing of software. The two terms most often at odds are “free software” and “open source software”, each with different origins and definitions. From this divergence have come the clunkier terms FOSS (free/open source software) and FLOSS (free/libre/open source software).

When Richard Stallman decided to take his “copyleft” approach in distributing the projects he worked on, he used the term “free software.” The Free Software Foundation (FSF) defines “free software” as “software that respects users' freedom and community. Roughly, the users have the freedom to run, copy, distribute, study, change and improve the software. With these freedoms, the users (both individually and collectively) control the program and what it does for them.”<sup>40</sup> Thus, the FSF explains, “you should think of ‘free’ as in ‘free speech,’ not as in ‘free beer’.”<sup>41</sup>

Conversely, the Open Source Initiative (OSI) prefers the term “open source software”, and explains that “[o]pen source doesn't just mean access to the source code.”<sup>42</sup> The OSI requires compliance with ten criteria: free redistribution, distribution of the source code (not simply the compiled software), permission to create derivative works, limited restrictions for the purpose of protecting the original “integrity,” no discrimination against people or groups, no discrimination against fields of endeavor, automatic application of the same license terms to redistributed copies, lack of reference within the license to a specific product, lack of restrictions on other software, and technological neutrality.<sup>43</sup>

While the term “free open source software” was first mentioned in a newsgroup in March 1998,<sup>44</sup> the term developed primarily as an option for those who did not wish to pick a side in the battle between the FSF and OSI definitions. In an early use of the term, in 2002 a Google Groups user wrote, “99% of the

---

<sup>40</sup> *The Free Software Definition*, GNU (Feb. 28, 2013), <http://www.gnu.org/philosophy/free-sw.html>.

<sup>41</sup> *Id.*

<sup>42</sup> *The Open Source Definition*, OPEN SOURCE INITIATIVE, <http://opensource.org/osd> (last visited Mar. 21, 2014).

<sup>43</sup> *Id.*

<sup>44</sup> *History*, FREE OPEN SOURCE SOFTWARE (last updated Sept. 14, 2012), <http://freeopensourcesoftware.org/index.php?title=History>.

software on my computer is F/OSS (Free/Open Source Software).”<sup>45</sup> The term FLOSS was likewise created by Rishab Aiyer Ghosh in an attempt to “somehow unify ‘Free Software’ and ‘Open Source’ concept into software.”<sup>46</sup> By adding the term “libre,” Ghosh hoped to highlight both the “free” (no cost) and “libre” (free from restrictions) aspects of the software.<sup>47</sup> While those representing developers should be aware of the differences in implications of the various terms, both “FOSS” and “FLOSS” have become umbrella terms that cover many variants of freely shared software.

### III IMPORTANT FOSS CASES

The field of FOSS law is a bit like an undisturbed minefield: much ground has yet to be covered, so it’s unclear where the problem areas are or how big their scope will be. Thus, the best thing a lawyer or license drafter can do is use an understanding of the existing cases to make sure that their license terms protect the software to the fullest extent possible.

One of the most important elements of a secure FOSS license is the phrasing of clauses that restrict the licensee’s behavior. For instance, a FOSS licensor might want to ensure that his licensee follows such FOSS precepts as attribution, sharing alike, and non-discrimination. To enforce a violation of these demands, he may be able to sue under two theories: copyright violation and breach of contract. Since a license is a contract in the traditional sense, a breach of contract claim is easily reached. However, to succeed on a claim of copyright infringement, the licensor must demonstrate either that no license was granted in the first place, or that the license was *conditional* upon the licensee not undertaking such behavior, and thus that due to a violation of the conditions the license effectively did not exist. To make this distinction easier to understand, consider the following scenario.<sup>48</sup>

---

<sup>45</sup> Heikki Orsila, *using m\$ products is supporting them*, GOOGLE GROUPS (Feb. 18, 2002, 6:37 AM), <https://groups.google.com/forum/#!msg/comp.sys.amiga.games/Z8Kh3jciTHA/VQzhmaguGUo>.

<sup>46</sup> Ricardo García Fernández, *Floss Developers*, UN AÑO CON EL MÁSTER DE SOFTWARE LIBRE EN LA URJC (Oct. 24, 2012), <http://mastersfwlurjc.blogspot.com/2012/10/floss-developers.html>.

<sup>47</sup> Frederick Noronha, *Book Review: Free/Open Source Software: Network Infrastructure and Security by Gaurab Raj Upadhyaya*, FREE SOFTWARE MAGAZINE (Aug. 16, 2007), <http://fsmsh.com/2331>.

<sup>48</sup> Given the intricacies of modern property law, we will assume for the purposes of this example that such a property-based contract would behave identically to a license agreement for intellectual property.

Amy and her family own a house with a large field behind it. Bob wishes to cut through Amy's field to make his commute to work shorter, so he approaches Amy. She is fine with the proposal, so long as Bob does not engage in any lewd activities in the field that might be visible to her children. She and Bob agree to the following language: "Amy grants Bob the right to traverse Amy's field whenever he pleases. Bob agrees that he will not engage in any lewd activities while in Amy's field." If Bob later streaks through Amy's field on his way home from the bar one night while Amy's children watch out the window, Amy may sue him only for breach of contract, because the grant of a right to traverse her field was not made *conditional* upon Bob conducting himself with decorum, and thus would be considered a *covenant*. Now consider an identical scenario where the agreement reads as follows: "So long as Bob does not engage in any lewd activities, Amy grants Bob the right to traverse her field whenever Bob pleases." Here, because Bob was engaging in lewd conduct (nudity) when he tried to traverse Amy's field, he effectively had no permission to do so; Amy's grant of permission contained a *condition*. This was a key distinction in several of the cases discussed below, as well as more nuanced questions about the permitted scope of conditions (e.g., should Amy be allowed to condition Bob's easement on something irrelevant to the land, like Bob not wearing a hat while crossing her field?).

#### A. *Graham v. James*

In the 1998 case of *Graham v. James*, the Second Circuit dealt with claims of copyright ownership and infringement from both parties and laid out several fundamental principles of contract law as they relate to licenses. In the late 1980s and early 1990s, Richard Graham sold CD-ROMs containing compilations of software programs that were free due to their status as shareware, freeware, or public domain software.<sup>49</sup> In 1991, he got in touch with defendant Larry James, a programmer, to commission a file-retrieval program to go on the discs. Graham described "in general terms" the type of program he wanted.<sup>50</sup> James agreed that in exchange for credit on the final project and a CD-ROM disk drive, he would write a file-retrieval program. Graham included this program in his next release.<sup>51</sup> Soon after, James wrote a new version in C++ ("the C++ version"). This version was an entirely new program, and Graham's only contributions were communicating the program's general requirements and collaborating on the organization of the files retrieved by the program. The C++ version also included authorship and copyright

---

<sup>49</sup> *Graham v. James*, 144 F.3d 229, 232–33 (2d Cir. 1998).

<sup>50</sup> *Id.* at 233.

<sup>51</sup> *Id.*

notices in James' own name.<sup>52</sup> When Graham discovered this notice, he removed it and released a new version of the disc.<sup>53</sup> Graham subsequently released several other compilations containing the C++ version. During this period, James also sold his C++ version to another CD-ROM publisher.<sup>54</sup>

Graham eventually brought suit against James for this sale of the program, claiming ownership of the C++ version's copyright under the work-for-hire doctrine and alleging that James's sale to a publisher constituted an infringement of Graham's copyright. James counterclaimed that he was the rightful owner of the copyright and that Graham's inclusion of the program on later compilations (and removal of James's copyright notice) constituted infringement. He also alleged breach of contract.<sup>55</sup>

Since the issue of who owned the copyright needed to be decided before the court could address the copyright infringement claims, the district court looked at each party's evidence and found that "the parties had orally agreed that James would provide Graham with a file retrieval program written in . . . C++" and that Graham would pay James for each version.<sup>56</sup> The district court granted a preliminary injunction barring James from "publishing, copying, selling, marketing or otherwise disposing of" programs that were substantially similar to the C++ version.<sup>57</sup> However, at trial the district court held that as an independent contractor James owned the copyright to the C++ version, and that Graham's releases with substantially similar programs infringed this copyright. The court awarded various damages and permanently enjoined Graham from using the program.<sup>58</sup>

Graham appealed to the Second Circuit, claiming that he had a license to use the copyrighted work.<sup>59</sup> The court acknowledged that "[u]nder federal law, 'nonexclusive licenses may . . . be granted orally, or may even be implied from conduct,'"<sup>60</sup> and pointed to the fact that the district court had found an implied license for the amount of Graham's payments between the two parties. They also

---

<sup>52</sup> *Id.*

<sup>53</sup> *Id.* at 234.

<sup>54</sup> *Id.*

<sup>55</sup> *Graham*, 144 F.3d 229 at 234.

<sup>56</sup> *Id.* at 233.

<sup>57</sup> *Id.* at 234.

<sup>58</sup> *Id.*

<sup>59</sup> *Id.* at 235.

<sup>60</sup> *Id.* (quoting 3 MELVILLE B. NIMMER & DAVID NIMMER, NIMMER ON COPYRIGHT § 10.03[A][7], at 10-43); *see also* I.A.E., Inc. v. Shaver, 74 F.3d 768, 775-76 (7th Cir. 1996).

expressed concern over the awarded copyright damages, since “a copyright owner who grants a nonexclusive license . . . waives his right to sue the licensee for copyright infringement.”<sup>61</sup> Regarding Graham removing the attribution notice mentioning James, the court noted that the prevailing view under copyright law is that an author who sells or licenses her work “does not have an inherent right to be credited as author of the work,” and explained that when a licensee omits such a notice, it has been held not to be infringing.<sup>62</sup> James claimed that even if there had been a license, it had been rescinded for nonpayment or non-attribution, but the court felt the existence of a license was “essentially uncontested.”<sup>63</sup> Instead, the court found the question at hand to be one of the *scope* and *duration* of the license. In situations where the question is simply the scope of a license, the court pointed out that the copyright owner has the burden of proof that the use in question is outside the activity authorized by the license.<sup>64</sup>

Ultimately, the court felt the issue of whether Graham had infringed James’s copyright rested on one question: were the payment of royalties and inclusion of a notice “crediting James’s authorship” considered *covenants* or *conditions*?<sup>65</sup> First, the court noted that under New York state law, terms of a contract are assumed to be covenants, not conditions.<sup>66</sup> Second, the court pointed to the fact that James tendered a copy of the C++ version to Graham for use before any royalties were paid and that the first version of the compilation with the C++ version was published with proper attribution. Both of these facts were relevant because New York does not treat as conditions those contractual obligations that are to be performed after partial performance by the other party.<sup>67</sup> Presented with these precedents, the Court of Appeals found that the attribution and royalty payment obligations were *covenants*.

---

<sup>61</sup> Graham, 144 F.3d at 236.

<sup>62</sup> *Id.*

<sup>63</sup> *Id.*

<sup>64</sup> *Id.*

<sup>65</sup> *Id.* at 237.

<sup>66</sup> *Id.*; *see also* Grand Union Co. v. Cord Meyer Dev. Co., 761 F.2d 141, 147 (2d Cir. 1985) (“In the absence of more compelling evidence that the parties intended to create a condition, the negotiation provision must be construed as a promise or covenant.”).

<sup>67</sup> 22 N.Y. JUR.2D CONTRACTS § 265 (1996); *see also* Jacob Maxwell, Inc. v. Veeck, 110 F.3d 749, 754 (11th Cir. 1997) (holding that royalty payments and author attribution were covenants because “[the song’s composer] expressly granted [the licensee] permission to play the song before payment was tendered or recognition received”).

James advanced a final argument that even if the obligations were covenants, Graham's breach of the two covenants had terminated the license.<sup>68</sup> In previous cases, both the Ninth and D.C. Circuits had held that a "material breach" could give rise to a right of rescission: "even if the counterclaims asserted merely constitute a breach of contract, an action for copyright infringement would lie if the breach is so material that it allows the grantor power to recapture the rights granted so that any further use of the work was without authority."<sup>69</sup> New York law generally allows rescission of a contract where the breach is "material and willful, or, if not willful, so substantial and fundamental as to strongly tend to defeat the object of the parties in making the contract."<sup>70</sup> However, in order to find that a contract has been rescinded, the non-breaching party (here, James) must have taken some affirmative steps towards rescission.<sup>71</sup> Ultimately, the court held that James had not sufficiently demonstrated that the contract had been successfully rescinded or otherwise terminated and remanded the case for further determination.

*Graham* was notable for a few reasons: first, it highlighted the important distinction between covenants and conditions in both awarding damages and succeeding in a case against a licensee run amok. Second, it laid out the difference between questions of the *existence* of a license and questions of a license's *scope* or *duration*. Third, it raised questions of whether a failure of attribution could ever constitute a violation of a condition, though there has been speculation that this holding was influenced by the lack of a written instrument.<sup>72</sup> Thus, *Graham's* primary lessons were the importance of putting agreements regarding software in writing and of framing important provisions as explicit conditions to the license.

---

<sup>68</sup> *Graham*, 144 F.3d at 237.

<sup>69</sup> *Costello Publ'g Co. v. Rotelle*, 670 F.2d 1035, 1045 (D.C. Cir. 1981); *see also* *Rano v. Sipa Press, Inc.*, 987 F.2d 580, 586 (9th Cir. 1993).

<sup>70</sup> *Callanan v. Powers*, 92 N.E. 747, 752 (N.Y. 1910).

<sup>71</sup> 22A N.Y. JUR.2D CONTRACTS § 497 (1996) ("The failure of a party to perform his part of a contract does not per se rescind it. The other party must manifest his intention to rescind within a reasonable time.").

<sup>72</sup> Evan Brown, *Open Source Software and the Covenant-Condition Dichotomy*, INTERNETCASES (Feb. 20, 2009), <http://blog.internetcases.com/2009/02/20/open-source-software-and-the-covenant-condition-dichotomy> ("It is difficult to ascertain how the licensor in *Graham* would have, in reality, viewed the use of his software without a copyright notice as being authorized (and therefore within the scope of the license). Perhaps the most plausible explanation for the contrary holding in *Graham* was the absence of a written agreement, and a presumption arising under New York law that the parties intend a covenant and not a condition.").

*B. Sun v. Microsoft*

The case of *Sun Microsystems v. Microsoft Corp.* posed somewhat similar questions to those in *Graham*, with one particularly notable change: there was a written agreement. In 1996, Microsoft was looking to license the right to use Sun's Java programming language in other products that it would create and distribute.<sup>73</sup> Sun<sup>74</sup> originally created Java "so that programmers could write a single program that would work on any operating system," and was determined to ensure that Microsoft distributed only versions of Java that maintained this cross-platform compatibility.<sup>75</sup> This demand became especially important when, midway through negotiations, talks broadened from use only in Microsoft's internet browsers and software development tools to a broader license that would allow Microsoft to include Java in its operating systems.<sup>76</sup> Sun's license agreements with other Java partners required products developed by licensees to pass Sun's compatibility tests, which Sun termed the JCK (Java Compatibility Kit) test suite.<sup>77</sup> If a product passed the test suite, the licensee was granted the right to display Sun's "JAVA Compatible" trademark on its product.

The terms of Sun's new agreement with Microsoft were recorded in a "Technology License and Distribution Agreement" (the "TLDA").<sup>78</sup> The TLDA granted Microsoft the right to a non-exclusive development license to use the source code of Sun's technology "for the purposes of developing, compiling to binary form and supporting [Microsoft's own] Products."<sup>79</sup> In addition, Sun granted Microsoft a non-exclusive distribution license to "make, use, import, reproduce, license, rent, lease, offer to sell, sell or otherwise distribute to end users as part of a Product or an upgrade to a Product, the Technology and Derivative

---

<sup>73</sup> *Sun Microsystems, Inc. v. Microsoft Corp.*, 81 F. Supp. 2d 1026, 1028 (N.D. Cal. 2000) [hereinafter *Sun V*].

<sup>74</sup> In 2010, Sun Microsystems was acquired by Oracle Corporation, another large computer technology corporation, who continued to market Sun's software portfolio. *Oracle and Sun Microsystems*, ORACLE, <http://www.oracle.com/us/sun/index.html> (last visited May 22, 2013).

<sup>75</sup> *Sun Microsystems, Inc. v. Microsoft Corp.*, 188 F.3d 1115, 1118 (9th Cir. 1999) [hereinafter *Sun III*].

<sup>76</sup> *Sun Microsystems, Inc. v. Microsoft Corp.*, 21 F. Supp. 2d 1109, 1113 (N.D. Cal. 1998) [hereinafter *Sun II*], *vacated*, 188 F.3d 1115 (9th Cir. 1999), *reinstated*, 87 F. Supp. 2d 992 (N.D. Cal. 2000).

<sup>77</sup> *Sun Microsystems, Inc. v. Microsoft Corp.*, 999 F. Supp. 1301, 1303 (N.D. Cal. 1998) [hereinafter *Sun I*].

<sup>78</sup> *Sun V*, 81 F. Supp. 2d at 1027.

<sup>79</sup> *Id.* at 1028 (quoting TLDA § 2.1(a)).



Works thereof in binary form.”<sup>80</sup> In a later clause within the same section, the TLDA placed compatibility requirements on Java-containing commercial distributions by Microsoft.<sup>81</sup> The TLDA likewise required that any Java compiler developed and distributed by Microsoft meet compatibility requirements.<sup>82</sup>

On September 30, 1997, Microsoft released the new version of its web browser, Internet Explorer (“IE 4.0”), on its website.<sup>83</sup> Microsoft claimed that IE 4.0 “incorporate[d] and [was] fully compatible with” the required version of Java, and displayed Sun’s “JAVA Compatible” trademark in close proximity to the IE 4.0 logo on both its website and CD-ROM versions. Testing by Sun revealed that IE 4.0 failed to pass the test suite, and on October 7, 1997, Sun notified Microsoft of the failure and requested that they commit to becoming compliant. Microsoft responded, claiming that IE 4.0 fully complied with requirements and that they would therefore continue to use the trademark.<sup>84</sup> Outside testing ultimately confirmed that IE 4.0 did not pass the test suite, a flaw that jeopardized the compatibility of the Java contained within IE 4.0.<sup>85</sup>

Sun took the case to court to request multiple forms of relief, starting with a preliminary injunction to stop Microsoft’s supposedly TLDA-violating conduct. In *Sun II*, the district court granted a preliminary injunction relating to Sun’s copyright infringement and unfair competition claims,<sup>86</sup> and it was this copyright infringement claim that provoked Microsoft’s appeal to the 9<sup>th</sup> Circuit in *Sun III*. The 9<sup>th</sup> Circuit found that the license provisions at issue could be either *covenants* or *conditions*.<sup>87</sup> Thus, the court vacated *Sun II*’s injunction and remanded the case for further determination. In *Sun V*, the court denied Sun’s request to reinstate the preliminary injunction, holding that Sun had not “definitively established” that it had a case for copyright infringement, rather than breach of contract.<sup>88</sup>

---

<sup>80</sup> *Id.* (quoting TLDA § 2.2(a)(iii)).

<sup>81</sup> *Id.* (quoting TLDA § 2.6(a)(vi)).

<sup>82</sup> *Id.* (TLDA § 2.6(b)(iv) reads “[A]ny new version of a Product that includes the Java Language compilation function that Licensee makes commercially available to the public after the most recent Compatibility Date shall include a mode which a Tool Customer may use to permit such Product to pass the Java Language Test Suite that accompanied the Significant Upgrade.”).

<sup>83</sup> *Sun I*, 999 F. Supp. 1301, 1305 (N.D. Cal. 1998).

<sup>84</sup> *Id.*

<sup>85</sup> *Id.*

<sup>86</sup> *Sun II*, 21 F. Supp. 2d 1109, 1127–28 (N.D. Cal. 1998)

<sup>87</sup> *Sun III*, 188 F.3d 1115, 1119–24 (9th Cir. 1999).

<sup>88</sup> *Sun V*, 81 F. Supp. 2d 1026, 1033 (N.D. Cal. 2000).

However, it was not until *Sun VI* that the issue of whether the violated license agreement provisions were covenants or conditions (referred to by the court as “license restrictions”) was addressed. Quoting *Graham v. James*, the court pointed out that a copyright owner who grants a nonexclusive license generally waives his right to sue the licensee for copyright infringement.<sup>89</sup> As in *Graham*, the court then moved on to principles found in state law regarding the interpretation of contracts. California law requires that in reviewing the intent suggested by the language of the TLDA, the court examine the “four corners of the instrument.”<sup>90</sup> The court held that “[t]he language and structure of the TLDA suggest that the compatibility obligations are separate covenants and not conditions of, or restrictions on, the license grants.”<sup>91</sup> Specifically, the court highlighted the fact that the language actually granting the software license made no reference to meeting compatibility requirements. The court contrasted this with Sun’s use of conditions in the TLDA’s “Trademark License” section, where the grant was limited to goods “that fully meet the certification requirements” and the license was later specified as only applying to versions that passed the JCK test suite.<sup>92</sup> In addition, the court read the remedies section to suggest a scheme where compatibility breaches, rather than immediately triggering a copyright violation, could be remedied within thirty days without penalty. Given these pieces of evidence, the court concluded, “the only reasonable interpretation of the TLDA is that the compatibility provisions of Section 2.6 are independent covenants, rather than limitations on the scope of Microsoft’s pre-termination license.” Thus, the court granted Microsoft’s motion for summary judgment and dismissed Sun’s claim for relief based on the allegation of copyright infringement.

*Sun VI* follows *Graham*’s precedent, but serves as an important case because it exemplifies the dangers still at hand for a licensor when there is a written agreement. When drafting clauses that grant any rights to the licensee, it is crucial that the sentence or clause containing such a grant include any restrictions that are fundamental to the licensor’s willingness to provide a license. The *Sun* cases also serve as a warning to those drafting clauses regarding available remedies, particularly when a licensor enters negotiations with a licensee before the terms of a license are agreed upon (as contrasted with situations where software is provided with a pre-attached license). First, drafters should write damages clauses that avoid

---

<sup>89</sup> *Sun Microsystems, Inc. v. Microsoft Corp.*, No. C 97-20884RMWPVT, 2000 WL 33223397, at \*3 (N.D. Cal. May 8, 2000) [hereinafter *Sun VI*].

<sup>90</sup> *Machado v. S. Pac. Transp. Co.*, 284 Cal. Rptr. 560, 562 (Ct. App. 1991).

<sup>91</sup> *Sun VI*, 2000 WL 33223397, at \*3.

<sup>92</sup> *Id.*

suggesting either that a violation of a condition (or other important covenant) is insignificant or that such a violation does not qualify as a material breach of the agreement. Additionally, licensors should carefully consider the ways in which licensees might violate the terms of a license agreement before allowing the inclusion of a clause restricting or barring the ability to obtain an injunction against a licensee. Ultimately, the *Sun* cases serve as a warning to developers and lawyers alike: even the big guys can mess up.

### *C. Jacobsen v. Katzer*

While a 2001 case in the 11<sup>th</sup> Circuit acknowledged that FOSS licensed under the GPL was not simply released into the public domain,<sup>93</sup> it was not until 2007 that a case progressed far enough<sup>94</sup> to fully discuss the merits of a FOSS license and whether its requirements could constitute conditions. In *Jacobsen v. Katzer*, plaintiff Robert Jacobsen worked at the Lawrence Berkeley National Laboratory and taught physics as a professor at the University of California, Berkeley.<sup>95</sup> A model train hobbyist, he was a leading member of the “Java Model Railroad Interface” (JMRI) Project, a SourceForge-based community that builds model railroad computer controls.<sup>96</sup> Through contributions by many JMRI participants, the project created an application called DecoderPro that “allows model railroad enthusiasts to use their computers to program the decoder chips that

---

<sup>93</sup> *Planetary Motion, Inc. v. Techsplosion, Inc.*, 261 F.3d 1188, 1198 (11th Cir. 2001) (“That the Software had been distributed pursuant to a GNU General Public License does not defeat trademark ownership, nor does this in any way compel a finding that [plaintiff] abandoned his rights in trademark. Appellants misconstrue the function of a GNU General Public License. Software distributed pursuant to such a license is not necessarily ceded to the public domain and the licensor purports to retain ownership rights, which may or may not include rights to a mark.”).

<sup>94</sup> Around the same time, the Software Freedom Law Center filed the first of several suits challenging violations of the “share-alike” provision of the GPL (mandating that the source code for derivative works be released) in regards to the FOSS product BusyBox. *On Behalf of BusyBox Developers, SFLC Files First Ever U.S. GPL Violation Lawsuit*, SOFTWARE FREEDOM LAW CENTER (Sept. 20, 2007), <http://www.softwarefreedom.org/news/2007/sep/20/busybox>. However, most of these suits were eventually settled. Of the four relevant decisions available on the Westlaw legal research service, none discuss the terms of the GPL.

<sup>95</sup> *Jacobsen v. Katzer*, No. C 06-01905 JSW, 2007 WL 2358628, at \*1 (N.D. Cal. Aug. 17, 2007) [hereinafter *Jacobsen I*], *vacated in part*, 535 F.3d 1373 (Fed. Cir. 2008).

<sup>96</sup> *What is JMRI?*, JMRI, <http://jmri.sourceforge.net> (last visited Mar. 21, 2014). Sourceforge is a website that provides software developers a place to collaborate on development, and offers both developers and companies a place to distribute their software. *About*, SOURCEFORGE, <http://sourceforge.net/about> (last visited Mar. 21, 2014).

control model trains.”<sup>97</sup> JMRI made DecoderPro available through SourceForge for free, but included a copyright notice and a reference to the included “COPYING” file, which contained the terms of the Artistic License, another popular FOSS license, in each of the package’s files.<sup>98</sup>

The defendant, Matthew Katzer, was the CEO and chairman of the board of directors for Kamind Associates, Inc. (KAM), a Portland, Oregon-based software company that developed software for model railroad enthusiasts.<sup>99</sup> KAM offered a competing software product called Decoder Commander, which is also used in the programming of decoder chips.<sup>100</sup> During the development of Decoder Commander, a KAM employee allegedly downloaded “decoder definition files” from DecoderPro and used portions of them in the code for Decoder Commander.<sup>101</sup> The files containing DecoderPro definitions violated the terms of the Artistic License in five ways, neglecting to include “(1) the authors’ 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.”<sup>102</sup> In addition, KAM changed the file names of various DecoderPro files without providing a reference to the original JMRI files or information about where to obtain the original versions.<sup>103</sup>

Jacobsen brought suit against Katzer and KAM, alleging that they “fraudulently secured patents for their software and, despite knowing the patents were invalid and unenforceable, sought to enforce the patents and collect patent royalties, and threatened litigation.”<sup>104</sup> In addition to seeking declaratory judgment of the unenforceability and invalidity of the patents and non-infringement of DecoderPro, Jacobsen brought claims against the defendants for antitrust, unfair competition, cybersquatting, libel, copyright infringement, trademark claims under 15 U.S.C. § 1125, and unjust enrichment.<sup>105</sup> While the court threw out many of his

---

<sup>97</sup> *Jacobsen v. Katzer*, 535 F.3d 1373, 1376 (Fed. Cir. 2008) [hereinafter *Jacobsen II*].

<sup>98</sup> *Id.*

<sup>99</sup> *Jacobsen I*, 2007 WL 2358628, at \*1.

<sup>100</sup> *Jacobsen II*, 535 F.3d at 1376.

<sup>101</sup> *Id.*

<sup>102</sup> *Id.*

<sup>103</sup> *Id.* at 1376–77.

<sup>104</sup> *Jacobsen I*, 2007 WL 2358628, at \*1.

<sup>105</sup> *Id.*

other claims, Jacobsen's claim of copyright infringement and related motion for a preliminary injunction became major issues in the case.<sup>106</sup>

In *Jacobsen I*, the district court for the Northern District of California addressed Jacobsen's motion for a preliminary injunction enjoining the defendants from "willfully infringing [his] copyrighted material."<sup>107</sup> The court noted that Jacobsen alleged copyright infringement, arguing that the defendants "without permission or consent, has [sic] made copies, distributed copies to the public, or created derivative works in violation of the exclusive rights."<sup>108</sup> The court then cited the three relevant provisions of the Artistic License which together state that those with access to the files may make copies, distribute and create derivative works from the software, provided that they give proper credit to the JMRI project: (a) that potential licensees may "make or give away verbatim copies of the source form . . . without restriction provided that [the licensee] duplicate all of the original copyright notices and assorted disclaimers," (b) that the licensee can distribute the copyrighted work "in a more-or-less customary fashion, plus [have] the right to make reasonable modifications," and (c) that the licensee may "distribute [the material] in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that [the licensee] not advertise [the material] as a product of [his] own."<sup>109</sup>

The court found that the license represented a choice by Jacobsen to distribute the decoder definition files "by granting the public a nonexclusive license to use, distribute, and copy the files."<sup>110</sup> It noted that the license was subject to several requirements (such as attribution), but held that implicit in a nonexclusive license is "the promise not to sue for copyright infringement."<sup>111</sup> In a major blow to FOSS goals, the court held that "[t]he condition that the user insert a prominent notice of attribution does not limit the scope of the license."<sup>112</sup> The court then acknowledged that actions exceeding the *scope* of the license could still give rise to a copyright infringement claim, but held that in granting rights to "any member of the public" to "use and distribute the [material] in a more-or-less customary fashion, plus the right to make reasonable accommodations," Jacobsen

---

<sup>106</sup> *Jacobsen II*, 535 F.3d at 1373.

<sup>107</sup> *Jacobsen I*, 2007 WL 2358628, at \*5.

<sup>108</sup> *Id.* at \*6 (quoting Amended Complaint at ¶ 100).

<sup>109</sup> *Id.* (quoting Amended Complaint at ¶ 100 and Suppl. Jacobsen Del. at ¶ 2, Ex. A).

<sup>110</sup> *Id.*

<sup>111</sup> *Id.*

<sup>112</sup> *Jacobsen I*, 2007 WL 2358628, at \*7.

had made the scope of the nonexclusive license “intentionally broad.”<sup>113</sup> Thus, the court found that Jacobsen only had a case for breach of the Artistic License, not copyright infringement, and denied Jacobsen’s motion for a preliminary injunction.<sup>114</sup>

Jacobsen appealed to the Federal Circuit (*Jacobsen II*). The Federal Circuit considered the history of “public licenses” (“often referred to as ‘open source’ licenses”).<sup>115</sup> It noted that one of the amici curiae, Creative Commons, provided free copyright licenses to allow parties to either release works into the public domain or license certain uses while keeping other rights reserved, and pointed to MIT’s use of a Creative Commons license for their course system, as well as uses of FOSS licenses by Apache, Firefox, and Wikipedia, as evidence that “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.”<sup>116</sup>

The court also noted several advantages of FOSS licensing and collaborations, such as faster writing and debugging of programs and lower development costs.<sup>117</sup> While it acknowledged that traditionally copyright holders have sold their copyrighted material “in exchange for money”, the court argued that FOSS licensing’s lack of monetary compensation “should not be presumed to mean that there is no economic consideration.”<sup>118</sup> To support this statement, the court cited the generation of market share, reputational gains, and improvements by outside contributors as gains created by FOSS licensing.

The court established that there was no dispute regarding the fact that Jacobsen held the copyright to certain materials throughout the website (a presumption established by his copyright registration), nor that portions of the DecoderPro software’s code were copied, modified, and distributed within Decoder Commander.<sup>119</sup> Given these facts, the court found a prima facie case of copyright infringement, subject to a finding that the defendants were acting outside

---

<sup>113</sup> *Id.*

<sup>114</sup> *Id.* Note that unlike infringement of a copyright, a breach of contract creates no presumption of irreparable harm.

<sup>115</sup> *Jacobsen II*, 535 F.3d 1373, 1378 (Fed. Cir. 2008).

<sup>116</sup> *Id.*

<sup>117</sup> *Id.* at 1378–79.

<sup>118</sup> *Id.* at 1379.

<sup>119</sup> *Id.*

the *scope* of the license. The court looked to the language of the Artistic License, which granted rights to copy, modify and distribute the software:

*provided that* [the user] insert a prominent notice in each changed file stating how and when [the user] changed that file, and provided that [the user] do at least ONE of the following:

- a) place [the user's] modifications in the Public Domain or otherwise make them Freely Available . . . .
- b) use the modified Package only within [the user's] corporation or organization.
- c) rename any non-standard executables so the names do not conflict with the standard executables, which must also be provided, and provide a separate manual page for each nonstandard executable that clearly documents how it differs from the Standard Version, or
- d) make other distribution arrangements with the Copyright Holder.<sup>120</sup>

The court highlighted the covenant–condition distinction, explaining that if the terms were found to be conditions, a claim would lie under copyright, while if they were merely covenants, Jacobsen’s only claim would be for breach of contract. It noted that the district court in *Jacobsen I* had not acknowledged this distinction, but had clearly been proceeding under the assumption that the terms were contractual covenants.<sup>121</sup> In his brief, Jacobsen argued that the terms should be read as conditions, while Katzer and KAM argued that they were covenants, citing *Gilliam v. ABC* for the premise that copyright law does not recognize a cause of action for non-economic rights (in the case of *Gilliam*, moral rights<sup>122</sup>).<sup>123</sup>

---

<sup>120</sup> *Jacobsen II*, 535 F.3d at 1380 (emphasis added).

<sup>121</sup> *Id.*

<sup>122</sup> *Gilliam v. Am. Broad. Cos., Inc.*, 538 F.2d 14, 24 (2d Cir. 1976) (“American copyright law, as presently written, does not recognize moral rights or provide a cause of action for their violation, since the law seeks to vindicate the economic, rather than the personal, rights of authors.”).

Ultimately, the Federal Circuit held that the terms were *conditions*, giving rise to a claim of copyright infringement. First, the court pointed to the fact that the Artistic License stated on its face that it created conditions: “The intent of this document is to state the conditions under which a Package may be copied.”<sup>124</sup> Each grant of a copyright-related right (copying, modifying, and distributing) within the License was also granted “provided that” the conditions were met, and California law generally construes “provided that” as creating a condition. The court also noted the importance of the attribution condition in allowing users of derivative works to backtrack to the original software. Further, the court explained its position on FOSS very clearly, writing that “[c]opyright holders who engage in open source licensing have the right to control the modification and distribution of copyrighted material.”<sup>125</sup>

Copyright licenses are designed to support the right to exclude; money damages alone do not support or enforce that right. The choice to exact consideration in the form of compliance with the open source requirements of disclosure and explanation of changes, rather than as a dollar-denominated fee, is entitled to no less legal recognition. Indeed, because a calculation of damages is inherently speculative, these types of license restrictions might well be rendered meaningless absent the ability to enforce through injunctive relief.<sup>126</sup>

Vacating *Jacobsen I* (thus denying a preliminary injunction) and remanding the case, the court explained that the Artistic License contained “clear language” creating conditions to protect the copyright holder’s rights and that those terms were enforceable copyright conditions. *Jacobsen II* serves as a beautifully-written ode to FOSS, clearly laying out the various gains available to a FOSS licensor and demonstrating that where a FOSS licensor takes care to choose a license with sufficiently clear wording of its FOSS conditions, courts are willing to recognize these conditions as restrictions on the license grant.

#### *D. MDY v. Blizzard*

The last case of particular relevance to the condition-covenant licensing distinction is one that did not involve FOSS, and perhaps due to that fact threatens to open up a dangerous line of defense for FOSS license violators. In the 2010 case

---

<sup>123</sup> *Jacobsen II*, 535 F.3d at 1381.

<sup>124</sup> *Id.*

<sup>125</sup> *Id.*

<sup>126</sup> *Id.* at 1381–82.



*MDY v. Blizzard*, the 9<sup>th</sup> Circuit dealt with license restrictions imposed by a video game manufacturer on players, in a case where game manufacturer Blizzard brought claims of copyright infringement against the creator of an automated game-playing program under theories of contributory and vicarious liability.

In 2004, Blizzard Entertainment created World of Warcraft (WoW), a massively multiplayer online role-playing game (MMORPG) in which players interact with a “virtual world” containing other players and interactive experiences, while advancing through the game’s seventy levels.<sup>127</sup> Players role-play as different characters including humans, elves, and dwarves, advancing through the game’s levels by participating in quests and battling monsters.<sup>128</sup> Blizzard requires players to read and accept an End User License Agreement (EULA) and Terms of Use (ToU) on multiple occasions.<sup>129</sup> The EULA pertains to the installed game client itself, and a player must agree to it upon both install and first running the game. The ToU covers the online service, and a player agrees to it both when creating an account and when first connecting to the service.

Declaratory judgment plaintiff MDY Industries was a software company run by Michael Donnelly. In 2005, Donnelly developed a software bot called Glider that automated the gameplay of WoW’s early levels, intending it solely for his personal use. Glider did not alter or copy WoW’s game client software in any way, did not allow users to avoid Blizzard’s monthly fees, and was not initially designed to avoid detection by Blizzard.<sup>130</sup> Later in 2005, Donnelly reviewed Blizzard’s EULA and client-server manipulation policy, reached the conclusion that bots were not prohibited under their terms, and began to sell Glider through MDY’s website for \$15 to \$25 per license.<sup>131</sup> Blizzard soon discovered Glider and in response launched a technology called Warden, which prevented players who used unauthorized third-party software (such as bots) from connecting to the WoW servers. In response, MDY altered Glider to evade detection by Warden, even advertising this alteration on its website.<sup>132</sup>

In 2006, after Blizzard threatened to sue, MDY commenced an action for a declaratory judgment that Glider’s files did not violate Blizzard’s copyright on the

---

<sup>127</sup> *MDY Indus., LLC v. Blizzard Entm't, Inc.*, 629 F.3d 928, 934 (9th Cir. 2010) [hereinafter *MDY II*].

<sup>128</sup> *Id.* at 935.

<sup>129</sup> *Id.*

<sup>130</sup> *Id.*

<sup>131</sup> *Id.* at 936.

<sup>132</sup> *Id.*

intellectual property contained in WoW (*MDY I*).<sup>133</sup> Blizzard counterclaimed against MDY and Donnelly for copyright infringement, asserting that the license it granted players to use the game (and therefore to copy the game to their computer's RAM) was conditioned upon not using bots, and thus that MDY was contributorily and vicariously liable for the unauthorized "copying" of WoW that occurred when Glider users played the game.<sup>134</sup> In response, MDY argued that when players used Glider, they were simply breaching a contract, not infringing a copyright.<sup>135</sup>

The district court cited *MAI Systems, Corp. v. Peak Computer, Inc.*, which held that copying software to RAM qualifies as "copying" under section 106 of the Copyright Act.<sup>136</sup> Thus, the court focused on the threshold issue of whether the "no bots" clauses within the EULA and ToU constituted covenants or conditions. The first paragraph of the EULA stated, "IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, YOU ARE NOT PERMITTED TO INSTALL, COPY, OR USE THE GAME."<sup>137</sup> Later, the EULA granted rights to install and use the game via the following language: "*Subject to your agreement to and continuing compliance with this License Agreement*, Blizzard hereby grants, and you hereby accept, a limited, non-exclusive license to (a) install the Game Client . . . and (b) use the Game Client . . . ." <sup>138</sup> Considered "in their entirety," the district court found that the "subject to" and "limited" language in the EULA and ToU established limitations on the *scope* of the license.<sup>139</sup> The court pointed to a section governing behavior in interactions between users as an example of Blizzard's use of independent covenants, pointing out that "[a] single contract clearly can contain both [covenants and conditions]."<sup>140</sup> Ultimately, the court held that Glider users acted outside the scope of this limited license and granted partial summary judgment to Blizzard based on a finding that Glider sales contributorily and vicariously infringed Blizzard's copyrights.<sup>141</sup>

---

<sup>133</sup> *MDY II*, 629 F.3d at 937.

<sup>134</sup> *MDY Indus., LLC v. Blizzard Entm't, Inc.*, No. CV-06-2555-PHX-DGC, 2008 WL 2757357, at \*3 (D. Ariz. July 14, 2008) [hereinafter *MDY I*].

<sup>135</sup> *Id.*

<sup>136</sup> *Id.* (citing *MAI Sys. Corp. v. Peak Computer, Inc.*, 991 F.2d 511, 518–19 (9th Cir.1993)).

<sup>137</sup> *Id.* at \*4.

<sup>138</sup> *Id.* (emphasis added).

<sup>139</sup> *Id.* at \*6.

<sup>140</sup> *Id.*

<sup>141</sup> *MDY II*, 629 F.3d at 937.

Soon after, MDY appealed to the Ninth Circuit (*MDY II*). In a deviation from previous cases like *Jacobsen*, the court looked to Delaware law (as provided for in the EULA and ToU) and found that “[c]onditions precedent are disfavored because they tend to work forfeitures.”<sup>142</sup> Noting that while the contract should be construed according to its terms if unambiguous, the court stated that where conflicting interpretations are possible, equity construes ambiguous provisions as covenants rather than conditions.<sup>143</sup> Given these standards of interpretation, the court found the bot and third-party software prohibitions to be covenants rather than copyright-enforceable conditions.<sup>144</sup> Specifically, the court claimed, “we have held that the potential for infringement exists only where the licensee’s action (1) exceeds the license’s scope (2) *in a manner that implicates one of the licensor’s exclusive statutory rights.*”<sup>145</sup> While the court believed a derivative works ban was grounded in copyright, it felt that the anti-bot provision was not, and thus that the provision was a covenant.

The court justified its decision by claiming that were it to hold otherwise, a software copyright holder could designate “any disfavored conduct during software use” as copyright infringement, citing RAM copies made during this conduct as a violation of the rights holder’s exclusive right to copy. Ultimately, the court believed that for a term to be a condition whose violation would give rise to a claim of copyright infringement, there needed to be a “nexus between the condition and the licensor’s exclusive rights of copyright.” As a result, the Ninth Circuit reversed the district court’s grant of summary judgment for Blizzard and vacated a related permanent injunction.<sup>146</sup>

The impact of *MDY* on FOSS licenses has yet to be fully felt: of the fourteen cases to cite the decision since it came down, none have dealt with a FOSS license or other similarly uncompensated license. Thus, it remains to be seen whether a court in the Ninth Circuit, following the precedent set by *MDY*, would uphold a FOSS license’s attribution clause (the type of clause at issue in *Graham*), since it does not “implicate[] one of the licensor’s exclusive [copyright] rights.”<sup>147</sup> Since the Copyright Act grants only the rights to reproduce the work, prepare derivative works, distribute copies, perform the work publicly, and display the

---

<sup>142</sup> *Id.* at 939 (citing *AES P.R., L.P. v. Alstom Power, Inc.*, 429 F.Supp.2d 713, 717 (D. Del. 2006)).

<sup>143</sup> *Id.*

<sup>144</sup> *Id.* at 940.

<sup>145</sup> *Id.* (emphasis added).

<sup>146</sup> *Id.* at 941–42.

<sup>147</sup> *Id.* at 940.

work publicly,<sup>148</sup> it remains to be seen whether any of the common FOSS requirements of distribution of derivative works' source code, application of an identical license to derivative works, notation of changes between the original and derivative works, and nondiscrimination would be held unenforceable.

#### IV DRAFTING AND ARGUING FOR FOSS LICENSES

##### *A. First Considerations*

Before reaching the covenant–condition distinction, a licensor must take an important first step: copyright registration. 17 U.S.C. § 411 reads, “no civil action for infringement of the copyright in any United States work shall be instituted until preregistration or registration of the copyright claim has been made.”<sup>149</sup> Further, while a copyright owner can technically bring an action for copyright infringement at any time after registration (regardless of how long after publication the registration happens), § 412(2) limits the availability of certain copyright remedies to those works whose registration “is made within three months after the first publication of the work.”<sup>150</sup> While some jurisdictions allow the *application* for the copyright to satisfy § 411, most jurisdictions require full completion of registration before a suit can be brought.<sup>151</sup> Where a creator wishes to file suit and needs to first complete an expedited registration, the applicable fee can be several hundred dollars more than the usual fee.<sup>152</sup> Thus, prompt registration of a software program's copyright is crucial.

The rewards for having previously registered a work's copyright are important for the FOSS community. Under the copyright statute, the available remedies for copyright infringement are injunctions,<sup>153</sup> impoundment and

---

<sup>148</sup> 17 U.S.C. § 106.

<sup>149</sup> 17 U.S.C. § 411.

<sup>150</sup> § 412(2).

<sup>151</sup> 5 PATRY ON COPYRIGHT § 17:78 (2013) (“Then Solicitor General Elena Kagan found, and every court of appeals to address the issue, as well as over 100 district courts, has held that actual registration, not mere application, is required to satisfy section 411(a), with the exception of the Fifth and the Ninth Circuit which have held that it is sufficient if the Copyright Office receives the application, deposit, and fees. District courts in the District of Columbia have also split on the issue, with most requiring actual registration.”).

<sup>152</sup> United States Copyright Office, *Circular 4: Copyright Office Fees*, COPYRIGHT.GOV, <http://www.copyright.gov/circs/circ04.pdf> (“The fee for expedited searches is \$445 an hour.”).

<sup>153</sup> 17 U.S.C. § 502.

destruction of illegal copies,<sup>154</sup> damages and profits (including statutory damages),<sup>155</sup> and costs and attorney's fees.<sup>156</sup> Of course, under a breach of contract theory, injunctions and monetary damages may be available, but there are two important advantages to requesting remedies under a copyright infringement theory. First, a category of copyright-only remedies becomes available. Assuming timely registration, a rights holder can, at any time during trial, opt to seek statutory damages—an award of “not less than \$750 or more than \$30,000 as the court considers just” (but up to \$150,000 for willful violations)—in lieu of actual damages.<sup>157</sup> In a situation where actual damages are speculative or otherwise difficult to calculate (as is often the case with FOSS infringements), statutory damages provide a mechanism for a rights holder to dispatch with the expensive process of proving damages, instead receiving what the court views to be a “just” sum. The copyright-only remedies category also includes recovery of costs and attorney's fees under § 505. Second, copyright registration invokes a slightly different approach towards injunctions. Where a plaintiff has demonstrated a likelihood of success on the merits, courts have historically relied on a judicially created presumption of irreparable harm, rather than undertaking an analysis to consider whether the plaintiff has demonstrated a likelihood of such harm.<sup>158</sup> However, in the 2010 case *eBay v. MercExchange*, the Supreme Court held that courts should not rely on this presumption and should instead utilize the traditional four-factor test for injunctions.<sup>159</sup> This would seem to suggest that the injunctions available under a copyright theory had been made commensurate with those under a breach-of-contract theory, especially since both the Second and Ninth Circuits have affirmed *eBay*'s holding.<sup>160</sup> However, the Federal Circuit—the first circuit to address *eBay*'s holding (in *Jacobsen*, no less)—“declined to invoke *eBay*'s standards, and instead applied the Ninth Circuit's pre-*eBay* presumption of irreparable harm,” the same presumption later overturned by the Ninth Circuit itself.<sup>161</sup> Likewise, many district courts have continued to apply the presumption.<sup>162</sup>

---

<sup>154</sup> § 503.

<sup>155</sup> § 504.

<sup>156</sup> § 505.

<sup>157</sup> § 504(c)(1)–(2).

<sup>158</sup> Erin V. Klewin, Reconciling Federal Circuit Choice of Law with *Ebay v. MercExchange*'s Abrogation of the Presumption of Irreparable Harm in Copyright Preliminary Injunctions, 80 *FORDHAM L. REV.* 2113, 2115 (2012).

<sup>159</sup> *Id.* at 2116.

<sup>160</sup> *Salinger v. Colting*, 607 F.3d 68 (2d Cir. 2010); *Perfect 10, Inc. v. Google, Inc.*, 653 F.3d 976 (9th Cir. 2011).

<sup>161</sup> Klewin, *supra* note 158, at 2113.

Since courts are clear that the “granting of an injunction to restrain a breach of contract rests largely in the sound discretion of the court,”<sup>163</sup> even post-*eBay* copyright-based injunction requests may be looked at more favorably than ones for breach of contract, and thus still provide advantages for a rights holder.

### *B. Best Practices*

In a perfect world, FOSS licenses and their publication would be so well handled as to avoid review by courts in the first place. Following best practices can provide enough clarity to avoid infringement by confused licensees and can also strengthen the licensor’s position if he ends up in court nonetheless.

Before licensing his software, a FOSS developer’s first step should be to register his copyright. As touched on above, this registration process can open the doors to vastly preferable remedies, such as statutory damages. Beyond the actual availability of statutory damages, their existence as a possibility can keep a licensor out of court in the first place: a licensee who is aware that statutory damages are available may be more willing to negotiate a settlement. Since the advent of the Copyright Office’s Electronic Copyright Office (eCO) system, registering a copyright has become a relatively simple process. A developer must simply go online to access the application form,<sup>164</sup> complete its fields, and pay a \$35 basic application fee.<sup>165</sup>

Second, any developer or lawyer working on a FOSS license should be careful that the license’s language properly defines important provisions of the license as conditions. For instance, references to an attribution requirement or “share-alike” policy for derivative works should be included within the granting sentence, introduced by conditional language such as “provided that” or “on the condition that.” In addition, suggestive words like “limited” should be used in describing the license, as courts have shown a belief that such language hints at a

---

<sup>162</sup> 6 PATRY ON COPYRIGHT § 22:44 n.14 (2013) (citing cases in North Carolina, Oregon, New York, and California).

<sup>163</sup> 43A C.J.S. INJUNCTIONS § 152 (2013).

<sup>164</sup> The Electronic Copyright Office (eCO) form is available at <http://www.copyright.gov/forms>. Instructions for submitting a paper application are also available at <http://www.copyright.gov/circs/circ61.pdf>.

<sup>165</sup> United States Copyright Office, *Fees*, COPYRIGHT.GOV, <http://www.copyright.gov/docs/fees.html>.

license's conditional restrictions.<sup>166</sup> Finally, any conditions should be explicitly referred to as "conditions."

Third, license drafters should keep in mind the possibility of rescission. Drafters may opt to include some type of termination clause (as in the Sun-Microsoft license) or may otherwise mention the licensor's ability to rescind or terminate the license upon material breach. Outside of this, a licensor should, as a safety measure, send notice of license termination where he believes the licensee has materially breached the contract, since this will make copyright remedies available if the court finds the violated term is a covenant and holds that the violation is "material."

Lastly, developers may wish to provide prominent notice to their users that the software is licensed under a FOSS scheme, explaining the purpose and breadth of requirements such as attribution and sharing alike. By doing so, licensors make it clear that the FOSS content is not in the public domain and thus alert users who intended to borrow code, thereby inadvertently violating the license, that such behavior is not condoned. Through careful attention to the act of licensing FOSS, a developer and his attorney can drastically reduce the risk of misuse and resulting damages to the software's FOSS goals.

### *C. Arguing a License Violation*

When a licensee has misused FOSS software, especially where the scope of the license is unclear, it is important to understand the categories that FOSS violations can fall into as well as the licensor's strongest available arguments. A preliminary distinction can be made between cases where there is some form of license and cases where there is none. In FOSS cases, the software is generally distributed with a pre-attached license for all users, which means that the category of cases involving no license is largely irrelevant. Thus, I will only consider approaches for cases where some semblance of a license is involved.

The first path a FOSS owner can take is to argue that while there is a license, the licensee has violated a *condition* of that license, and thus is liable for copyright infringement, not simply breach of contract. A simple example would be a license whose grant of rights read, "Licensor grants licensee a limited license to distribute the work *provided that the licensee distributes the work with all attributions and*

---

<sup>166</sup> See, e.g., *MDY I*, No. CV-06-2555-PHX-DGC, 2008 WL 2757357 at \*4 (D. Ariz. July 14, 2008) ("The title—'Grant of Limited Use License'—makes clear that the license is limited, as does the later reference to a 'limited, non-exclusive license.'")

*notices intact.*” Here, if a licensee distributed the work with attribution to the licensor stripped away, the licensor could argue that he had never provided a license for such conduct. Essentially, the licensor would argue that the licensee had acted outside the *scope* of the license, and was therefore liable for copyright infringement.<sup>167</sup>

If a court holds a FOSS license term to be a *covenant* due to a reading of either the language of the license or the limitations of *MDY*, a licensor can argue that the licensee’s breach constituted a “material breach.” In many jurisdictions, such a breach could give the licensor the ability to rescind the contract.<sup>168</sup> If a licensor can establish rescission of the contract, the license is essentially stripped away, re-exposing the licensee to copyright liability. However, to succeed in making a claim of rescission, the licensor must have undertaken an “affirmative act” to indicate rescission, such as “transmitting a notice of termination.”<sup>169</sup> Thus, it is important that a licensor attempt to notify a licensee of termination where the licensor believes there has been a material breach.

When a licensor cannot succeed in arguing that a condition was violated or that rescission resulted from a material breach, the licensor’s options are more limited. Without the ability to reach the copyright remedies available to a rights holder whose copyright was infringed, a licensor must resort to a breach of contract claim. This means that in order to collect any monetary damages, he will need to establish his actual damages and/or any profits reaped by the defendant. However, where irreparable harm can be demonstrated, injunctions may still be available, thus making such a claim useful in certain situations.

### CONCLUSION

Lawyers as a group are not known for their tech savvy or rapid embrace of emerging technologies. In an interview regarding law firms’ adoption of online tools for lawyers and firms, a partner at the Boston firm Foley & Lardner joked that “law firms must innovate to finally move into the 20th century.”<sup>170</sup> For every lawyer who cannot find the ‘any’ key, however, there is a lawyer working with

---

<sup>167</sup> Note, however, that in a situation where the condition involves attribution (as in the above example), the defendant could argue that such language could only be read as a covenant under *MDY*, since attribution doesn’t “implicate a statutory right.”

<sup>168</sup> See sources cited *supra* note 69.

<sup>169</sup> Jay Dratler, Jr., LICENSING OF INTELLECTUAL PROPERTY § 1.06[1] at 1–50.13 (2003).

<sup>170</sup> Scott Kirsner, *Start-ups Take on Tough Customers: Lawyers*, BOSTON GLOBE (Sep. 1, 2013), <http://www.bostonglobe.com/business/2013/08/31/tech-start-ups-target-tough-customer-law-firms/fyznk5CXkhnCqQzHpGEIGO/story.html>.



software developers, crafting the licenses for tomorrow's killer apps. It is this latter category of attorneys who must make sure that they understand the 'covenant versus condition' distinction, because the next time this issue comes to bear, it might be more than model trains or a video game at stake.