

3-1-2006

Syntax Errors: Why Version 3 of the GNU General Public License Needs Debugging

Doug Ferguson

Follow this and additional works at: <http://scholarship.law.unc.edu/ncjolt>



Part of the [Law Commons](#)

Recommended Citation

Doug Ferguson, *Syntax Errors: Why Version 3 of the GNU General Public License Needs Debugging*, 7 N.C. J.L. & TECH. 397 (2006).
Available at: <http://scholarship.law.unc.edu/ncjolt/vol7/iss2/4>

This Comments is brought to you for free and open access by Carolina Law Scholarship Repository. It has been accepted for inclusion in North Carolina Journal of Law & Technology by an authorized administrator of Carolina Law Scholarship Repository. For more information, please contact law_repository@unc.edu.

**SYNTAX ERRORS: WHY VERSION 3 OF THE GNU GENERAL
PUBLIC LICENSE NEEDS DEBUGGING**

Doug Ferguson¹

The GNU General Public License ("GPL") is the most popular license in use for free and open-source software projects. Now in its fifteenth year, the GPL has endured both practical and legal challenges and today enjoys widespread use and a reputation as a legitimate legal instrument. However, recently proposed changes to the GPL will harm this reputation and may prove counterproductive to the GPL's continued acceptance. This Recent Development will briefly introduce the GPL and examine some of the more problematic changes of the proposed GPL. Finally, this Recent Development will propose revisions which will allow the GPL to continue enjoying the legitimacy it has earned, while maintaining consistency with both its past and future intended effects.

I. INTRODUCTION

Most software is sold subject to a software license, which specifies what you may or may not do with the software that you have purchased. For example, the End-User License Agreement ("EULA") for Microsoft Windows XP Professional, which governs use of the product, stipulates that the purchaser may install one copy of Windows XP on a single computer.² While a provision limiting installations to one computer per copy may seem reasonable, the license contains additional terms that are more

¹ J.D. Candidate, University of North Carolina School of Law, 2007.

² MICROSOFT CORP., MICROSOFT WINDOWS XP PROFESSIONAL END-USER LICENSE AGREEMENT at 1 (2001), <http://www.microsoft.com/legal/useterms/> (select "Windows XP" under "Product Name"; then select "Professional" under "Version"; then select "English" under "Language"; then click "Go"; then follow "Windows XP_Professional_English.pdf" hyperlink) (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

arbitrary. For instance, the EULA permits users to connect no more than ten “computers or other electronic devices” to the computer on which Windows is installed, and then “solely for File and Print services, Internet Information Services, and remote access”³ In addition, “[t]he ten connection maximum includes any indirect connections made through ‘multiplexing’ or other software or hardware which pools or aggregates connections.”⁴ Microsoft further provides that the licensee “may not reverse engineer,⁵ decompile,⁶ or disassemble⁷ the Product [Windows XP], except and only to the extent that it is expressly permitted by applicable law notwithstanding this limitation.”⁸ Whether or not most purchasers of Windows XP Professional are

³ *Id.*

⁴ *Id.* “Multiplexing” refers generally to “[t]he combining of two or more information channels onto a common transmission medium.” Alliance for Telecommunications Industry Solutions, ATIS Telecom Glossary 2000, http://www.atis.org/tg2k/_multiplexing.html (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). The practical meaning for the purpose of this license is that if the user has combined five individual connections to the computer into one larger connection, the user is using five connections, not one.

⁵ Under the “reverse engineering” doctrine, “[c]ustomers who buy a product on the open market are entitled to break it apart to see how it works.” ROBERT P. MERGES ET AL., *INTELLECTUAL PROPERTY IN THE NEW TECHNOLOGICAL AGE* 866 (3d ed. 2003). Reverse engineering of software “typically involves two phases: (1) disassembly or decompilation of the program in order to create human-readable source code that may be analyzed, and (2) using the results of this analysis to create a commercially viable program.” *Id.* at 917-18.

⁶ “Decompilation” is the process by which “object code” (machine-readable instructions) are “transmogrified in a manner that represents human-friendly source code.” *Texas Instruments, Inc. v. Hyundai Elecs. Indus. Co.*, 190 F.R.D. 413, 414 (1990).

⁷ “Disassembly” refers to “translat[ion] [of machine-readable instructions] into assembly code, which a person [who is] knowledgeable in assembly code can read.” *Id.* at 415. While decompilation and disassembly both refer to transforming machine instructions into something more readable by humans, they are distinct processes. *See id.* at 417 (“Decompilation and disassembly are not the same thing.”). The lesson to take from this license provision is that attempting to learn the inner workings of Windows XP Professional, at least by examining a copy of it, is prohibited by the EULA. *See MICROSOFT, supra* note 2, at 1.

⁸ MICROSOFT, *supra* note 2, at 2.

interested in doing any of these things, it is clear that Microsoft intends to exert restrictions not only on *how* one may acquire its software, but also on *what* one may do with the software after it is acquired. In fact, Microsoft cautions that one does not acquire their software, but only a right to use it: "The Product is licensed, not sold."⁹

Free and open-source software ("FOSS") projects take a much different approach. FOSS projects are not "owned" by any entity in the sense that Microsoft owns Windows XP Professional. Rather, FOSS projects are collaborative efforts which welcome programming expertise, and sometimes financial contributions, from volunteers. The resulting software may be used, free of charge, by anyone.¹⁰ There are no restrictions on using techniques like reverse-engineering, decompiling and disassembling to understand how the software works, although such techniques are rarely needed, since the source code (the human readable instructions written by programmers) to FOSS projects is freely available.¹¹ Most FOSS projects do not apply any restrictions to what the software may be used for, or on how many computers users may install the software. While these projects are usually subject to some license restrictions regarding how they may be distributed, as discussed below, these restrictions are designed to ensure that subsequent recipients of the software enjoy the same freedoms.¹² Thus, licenses commonly used for FOSS projects are considerably more permissive than licenses used for commercial software, such as the Windows XP EULA.

⁹ *Id.* at 4.

¹⁰ Open Source Initiative ("OSI"), *The Open Source Definition*, <http://www.opensource.org/docs/definition.php> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

¹¹ OSI, *supra* note 10.

¹² See OSI, *supra* note 10. "[An open source] 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 By constraining the license to require free redistribution, we eliminate the temptation to throw away many long-term gains in order to make a few short-term sales dollars." *Id.*

The GNU¹³ General Public License (“GPL”) is the most popular software license for FOSS projects.¹⁴ Although its legal enforceability has been the subject of speculation, it nonetheless has proved to be a remarkably effective instrument for achieving its purpose: making free software widely available for use and modification, all while ensuring that the software remains free. However, a recently proposed update to the license, GPL Version 3 (“GPLv3”), contains provisions that are both ambiguous and poorly suited to a software license and, consequently, may compromise GPLv3’s viability as an enforceable legal instrument. This Recent Development will provide a brief introduction to the philosophy behind the GPL, and its successes in upholding that philosophy. This article will also discuss some problems with the proposed revision, and offer a solution aimed at helping GPLv3 meet its intended goals without compromising its legal viability.

II. BACKGROUND OF THE GPL

A. *Philosophy*

The core philosophy of the GNU Project (“GNU”) and the Free Software Foundation (“FSF”)¹⁵ can be roughly stated as the idea

¹³ “GNU” stands for “GNU’s Not Unix,” a so-called recursive acronym. *See* The GNU Operating System, <http://www.gnu.org/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

¹⁴ *See, e.g.,* Freshmeat.net, *Statistics and Top 20*, <http://freshmeat.net/stats/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

¹⁵ Strictly speaking, the GNU Project refers to a software project, and the Free Software Foundation refers to that project’s “principal organizational sponsor.” *See* The GNU Operating System, <http://www.gnu.org/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). According to FSF’s home page, FSF provides services for developers to contribute to the GNU project. The Free Software Foundation, <http://www.fsf.org/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). The relationship between the two organizations is symbiotic, and the distinction between them is not always obvious. The FSF holds the copyright to the GNU Public License, while the GNU Project’s website contains a considerable amount of material promoting the free software philosophy. While this article attributes statements to their respective issuing organizations, one should keep in

that software “should be free.”¹⁶ This summary, however, falls far short of capturing the essence of the idea. In order to qualify as free software, a software package must: (1) be available at no charge; (2) be usable for any purpose, also at no charge; and (3) be open to anyone (at least, anyone with some programming skill) to study and change.¹⁷ GNU makes clear that the term free software goes beyond monetary cost. Rather, “[f]ree software’ is a matter of liberty, not price. To understand the concept, you should think of ‘free’ as in ‘free speech,’ not as in ‘free beer.’”¹⁸ Further separating free software from the notion of price, GNU stipulates that money may indeed change hands when free software is distributed – as long as the payment is in the form of a fee for the distribution, and not for the software itself.¹⁹ “[I]f you are redistributing copies of free software, you might as well charge a substantial fee and *make some money*. Redistributing free software is a good and legitimate activity; if you do it, you might as well make a profit from it.”²⁰ Thus, the true meaning of free software advanced by GNU is both more and less rigorous than its name suggests.

GNU spends a considerable amount of time establishing specific semantics to describe the characteristics of free software. For example, “commercial” does not mean non-free software, but rather any software developed as a business activity,²¹ “freeware”

mind that both organizations promote a single, identical “free software” philosophy.

¹⁶ Richard Stallman, *Why Software Should Be Free*, <http://www.gnu.org/philosophy/shouldbefree.html>, Apr. 24, 1992 (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

¹⁷ The GNU Project, *The Free Software Definition*, <http://www.gnu.org/philosophy/free-sw.html> (last visited Mar. 7, 2006) [hereinafter *Free Software Definition*] (on file with the North Carolina Journal of Law & Technology).

¹⁸ Free Software Definition, *supra* note 17.

¹⁹ The GNU Project, *Selling Free Software*, <http://www.gnu.org/philosophy/selling.html> (last visited Mar. 7, 2006) [hereinafter *Selling Free Software*] (on file with the North Carolina Journal of Law & Technology).

²⁰ *Selling Free Software*, *supra* note 19 (emphasis in original).

²¹ The GNU Project, *Some Confusing or Loaded Words and Phrases that are Worth Avoiding*,

does not mean free software, but rather software available at no charge that is in other respects not “free;”²² and there really is no such thing as a “software industry,” because software development is not something that occurs in a factory.²³

Another example sheds particular light on the true reasoning behind the GNU philosophy. By definition, free software must be “open-source,” meaning that the source code to the software must be available so that the program’s components can be studied and modified.²⁴ But to GNU, simply making the source code available misses the deeper meaning behind free software.²⁵ That is, while free software is by definition open-source and open-source software shares several critical requirements with free software,²⁶ the motivations behind the open-source philosophy and the free software philosophy are different. It has been stated that “[o]pen source is a development methodology; free software is a social movement.’ For the Open Source movement, non-free software is a suboptimal solution. For the Free Software movement, non-free software is a social problem and free software is the solution.”²⁷

The Free Software movement thus teaches not only that free software is preferable for practical and development reasons, but also that non-free software is inherently immoral. In particular, the movement teaches that copyrights on software are unethical,²⁸ that

<http://www.gnu.org/philosophy/words-to-avoid.html> (last visited Mar. 7, 2006) [hereinafter *Confusing or Loaded Words and Phrases*] (on file with the North Carolina Journal of Law & Technology).

²² See *Confusing or Loaded Words and Phrases*, *supra* note 21.

²³ See *Confusing or Loaded Words and Phrases*, *supra* note 21.

²⁴ See *Free Software Definition*, *supra* note 17.

²⁵ The GNU Project, *Why ‘Free Software’ is better than ‘Open Source,’* <http://www.gnu.org/philosophy/free-software-for-freedom.html> (last visited Mar. 7, 2006) [hereinafter *Why Free Software is Better*] (on file with the North Carolina Journal of Law & Technology).

²⁶ Open Source Initiative (“OSI”), *The Open Source Definition*, <http://www.opensource.org/docs/definition.php> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). Key shared requirements include free redistribution, access to source code, and no restrictions on the use of the software. *Id.*

²⁷ *Why Free Software is Better*, *supra* note 25.

²⁸ See Richard Stallman, *Why Software Should Not Have Owners*, <http://www.gnu.org/philosophy/why-free.html> (last visited Mar. 7, 2006)

the terms “piracy” and “theft” are misleading when applied to software,²⁹ that “sharing information with your neighbor” is a more appropriate term than “theft,”³⁰ and that software publishers use “draconian measures” to enforce their copyrights and have instituted police-state style tactics to protect their “property.”³¹ The view that software belongs to some class of things that cannot legitimately be owned is obviously incompatible with the prevailing practices of the “software industry” (as that term is generally understood).

Intent upon bringing free software into the world, but mindful of the need for a legally viable instrument for enforcing the philosophy, the GNU Project developed the GPL, first applied in 1989.³² The license embodies the philosophy of free software, entitling anyone to download, use, and redistribute a GPL-covered software package at no charge—and most importantly, with GNU-style freedom.³³ All subsequent recipients of the software enjoy the same rights.³⁴ One may use the original source code to create new programs based on the software.³⁵

To preserve the freedom of the software, however, the license imposes a condition: if GPL-covered software is modified or used as a basis for some new project, and subsequently redistributed, no additional restrictions may be placed on the resulting work.³⁶ This

[hereinafter *Why Software Should Not Have Owners*] (on file with the North Carolina Journal of Law & Technology). “If your friend asks to make a copy [of proprietary software], it would be wrong to refuse. Cooperation is more important than copyright.” *Id.*

²⁹ *Why Software Should Not Have Owners*, *supra* note 28.

³⁰ *Confusing or Loaded Words and Phrases*, *supra* note 21.

³¹ *Why Software Should Not Have Owners*, *supra* note 28.

³² The GNU Project, *General Public License, Version 1*, <http://www.gnu.org/copyleft/copying-1.0.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). *See also* The GNU Project, *GNU General Public License*, <http://www.gnu.org/licenses/gpl.html> (last visited Mar. 7, 2006) [hereinafter *GNU GPL: Version 2*] (on file with the North Carolina Journal of Law & Technology) (reciting the current recommended version of the GPL, Version 2).

³³ *GNU GPL: Version 2*, *supra* note 32.

³⁴ *GNU GPL: Version 2*, *supra* note 32.

³⁵ *GNU GPL: Version 2*, *supra* note 32.

³⁶ *GNU GPL: Version 2*, *supra* note 32.

applies whether or not the distribution is for a fee and irrespective of whether the distribution occurs in source code or some other form.³⁷ The practical effect of this requirement is that where, for example, a company creates a program based on GPL-covered software which it uses only internally, the company is not required to share the new program.³⁸ However, if a company uses a modified version of a GPL-covered work in a product that it *sells*, then that changed version has been “distributed” and must be made available to the world under the terms of the GPL.³⁹ GNU uses the broad term “copyleft” to describe this requirement that “anyone who redistributes the software, with or without changes, must pass along the freedom to further copy and change it.”⁴⁰

B. The Successes of Free Software

Although in some respects at odds with typical business thinking, free software has had undeniable successes in the real-life marketplace. Freshmeat.net, a web directory of software projects, states that about two-thirds of the roughly 40,000 projects listed are distributed under the terms of the GPL.⁴¹ Noteworthy examples include Linux, the popular operating system;⁴² MySQL, a database server;⁴³ and Perl, a programming language.⁴⁴ The

³⁷ GNU GPL: Version 2, *supra* note 32.

³⁸ Free Software Definition, *supra* note 17.

³⁹ Free Software Definition, *supra* note 17.

⁴⁰ The GNU Project, *What is Copyleft?*, <http://www.gnu.org/copyleft/copyleft.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁴¹ Freshmeat.net, *supra* note 14.

⁴² Strictly speaking, the name “Linux” refers only to the core (“kernel”) of the operating system. See, e.g., Richard Stallman, *Linux and the GNU Project*, <http://www.gnu.org/gnu/linux-and-gnu.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). To be useful, an operating system requires external programs (“utilities”) in addition to the kernel. *Id.* GNU has urged that “GNU/Linux” is a more accurate name for such a system. *Id.* Generally, these systems are simply referred to as “Linux” in common usage. *Id.*

⁴³ See MySQL, *MySQL 5.0 Reference Manual: 1.4 Overview of the MySQL Database Management System*, <http://dev.mysql.com/doc/refman/5.0/en/what-is.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

continued growing use of Linux as a corporate server platform suggests that businesses have found the GPL's terms of use acceptable.⁴⁵ IBM has gone so far as to contribute a version of one of its software products, the Journaled File System (JFS), to Linux, also under the GPL.⁴⁶

In addition, the widespread use of free software has opened a new avenue for some hardware makers. These companies have recognized that using a freely-available operating system as a starting point, and making modifications to support specific hardware, can have economic advantages over developing a new operating system from scratch, even if the GPL requires that the modified version be released to the public. Device manufacturers following this route include Linksys, a manufacturer of wireless networking equipment, for some of its wireless routers;⁴⁷ Hewlett-Packard, for one of its network switching devices;⁴⁸ and Gamepark Co., Ltd., for its GP2X Personal Entertainment Player.⁴⁹ This

⁴⁴ See Perl Licensing, <http://dev.perl.org/licenses/index.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). Perl may be used and distributed under either the GPL or the "Artistic License." *Id.*

⁴⁵ "Linux servers generated \$1.6 billion in quarterly revenue, the fourteenth consecutive quarter of double-digit growth, with year-over-year revenue growth of 20.8% [C]ustomers continued to expand the role of Linux servers into an increasingly wider array of commercial and technical workloads." Press Release, IDC, Worldwide Server Market Slows in Fourth Quarter But Grows to \$51.3 Billion in 2005, Highest Revenues in 5 Years, According to IDC (Feb. 22, 2006), <http://www.idc.com/getdoc.jsp?containerId=prUS20074406> (last visited Apr. 3, 2006) (on file with the North Carolina Journal of Law & Technology).

⁴⁶ JFS Project, <http://jfs.sourceforge.net/home.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁴⁷ Linksys, *Support/Technical/GPL Code Center*, <http://www.linksys.com/> (follow "Downloads" hyperlink under "Support"; then follow "GPL Code Center" hyperlink) (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁴⁸ Hewlett-Packard, *Download HP 24-Port 4x Fabric Copper Switch GPL Sourcecode*, <http://h18002.www1.hp.com/support/files/server/us/download/23719.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁴⁹ Gamepark has made the source code for version 1.4.0 of its device available, in a form suitable for the technically-inclined. See

practice marks a shift away from the traditional practice of attempting to increase the value of products by keeping their inner workings a secret.

In addition, a new industry has developed around free software services: selling not free software itself, but distribution, consulting, and support for those using or wishing to use free software. This is explicitly permitted by the GPL⁵⁰ and encouraged by GNU.⁵¹ This activity can take the form of selling a simple “distribution” of Linux (e.g., Slackware, available on CD for \$39.95),⁵² a full-service package including software and different support channels (e.g., a Premium Subscription to Red Hat Enterprise Linux, at \$2499.00 annually),⁵³ or any conceivable form in between. A remark from the GNU Manifesto, first written in 1985, seems especially prescient given this result: “[U]sers who know nothing about computers need handholding: doing things for them which they could easily do themselves but don’t know how. Such services could be provided by companies that sell just handholding and repair service.”⁵⁴ Though the GPL rejects the idea of software as property, it can hardly be judged *per se* incompatible with commercial activities.

C. Legal Status

The GPL has not been directly tested in an American court, and the enforceability of the license has been a topic of some speculation. Despite the speculation, however, the license has fared well. Eben Moglen, General Counsel for the Free Software

<http://svn.gp2x.com/gp2x/tag/kernel/1.4.0/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁵⁰ *GNU GPL: Version 2*, *supra* note 32.

⁵¹ *Selling Free Software*, *supra* note 19.

⁵² Slackware Linux, *Product Details*, <http://store.slackware.com/cgi-bin/store/slack10.2> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁵³ Red Hat, *Server Support and Pricing*, <http://www.redhat.com/rhel/compare/server/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁵⁴ The GNU Project, *The GNU Manifesto*, <http://www.gnu.org/gnu/manifesto.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

Foundation, declares that “as a copyright license the GPL is absolutely solid. That’s why I’ve been able to enforce it dozens of times over nearly ten years, without ever going to court.”⁵⁵ Moglen also explains that most violations have been unintentional and have been cured by simply bringing the violation to the violator’s attention.⁵⁶ Where parties have instead been unwilling to comply with the GPL—for instance, by distributing GPL-covered software within a commercial product without making that software available to others—Moglen claims that the problem has been solved by going directly to the product’s customers, who can presumably apply pressure to the supplier under threat of taking their business elsewhere.⁵⁷

True challenges to the GPL’s enforceability have been less frequent. A 2005 case alleging that the GPL is an illegal restraint of trade for fixing prices at zero was dismissed for failure to show harm to consumers.⁵⁸ Currently, The SCO Group⁵⁹ is litigating claims that Linux contains portions of SCO’s proprietary source code, alleging that the GPL is “poorly written and unenforceable,”⁶⁰ in part because it violates Article I, Section 8 of

⁵⁵ Eben Moglen, *Enforcing the GPL*, LINUX USER, Sept. 2001, at 66, available at http://www.linuxuser.co.uk/images/stories/pdf/lu14-Free_Speech-Enforcing_the_GPL.pdf (last visited Apr. 3, 2006) (on file with the North Carolina Journal of Law & Technology).

⁵⁶ Eben Moglen, *Enforcing the GPL*, Part Two, LINUX USER, Oct. 2001, at 66, http://www.linuxuser.co.uk/images/stories/pdf/lu15-Free_Speech-Enforcing_the_GPL_part_two.pdf (last visited Apr. 3, 2006) (on file with the North Carolina Journal of Law & Technology).

⁵⁷ *Id.*

⁵⁸ See *Wallace v. Free Software Foundation*, No. 1:05-cv-00618-JDT-TAB, 2005 U.S. Dist. LEXIS 31728 (S.D. Ind. Nov. 28, 2005) (dismissing an antitrust action against FSF for failure to state antitrust injury).

⁵⁹ “SCO” refers to “[The] Santa Cruz Operation,” the original name of the company. The SCO Group, Inc., *History of SCO*, <http://www.sco.com/company/history.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁶⁰ The SCO Group, Inc., *SCO Intellectual Property FAQ*, <http://www.sco.com/scosource/ipprotectionfaq.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

the United States Constitution.⁶¹ This particular allegation has been described as “ludicrous” by at least one commentator.⁶² A 2004 German case held that the provisions of the GPL requiring source code availability where a modified work is distributed, as well as providing for a loss of rights upon failure to so redistribute, are indeed enforceable in Germany.⁶³

In sum, the GPL has proved to be a remarkably durable and flexible instrument for achieving its purposes. It is no longer just part of a social movement; rather, a number of companies have recognized the GPL as a valid software license and voluntarily complied with its terms. The trend appears to be toward a presumption that the GPL is indeed enforceable, even where it is raised as part of a legal dispute.⁶⁴ As GPL Version 2 enters its fifteenth year of widespread use, there is little reason to doubt its legitimacy, even in the absence of a concrete American ruling on the issue.

III. LEGAL AND TECHNOLOGICAL CHANGES AND THE RESPONSE: GPLv3

The Free Software Foundation released a proposed Version 3 of the GPL (GPLv3) for public comment on January 16, 2006.⁶⁵ GNU states that the key motivation behind the revised license is

⁶¹ Darl McBride, *Open Letter on Copyrights*, Dec. 4, 2003, <http://www.sco.com/copyright/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁶² Jason Wacha, *Taking the Case: Is the GPL Enforceable?*, 21 SANTA CLARA COMPUTER & HIGH TECH. L.J. 451, 459 (2005).

⁶³ JBB Rechtsanwälte, *Munich Court Confirms GPL Enforceability*, Aug. 13, 2004, <http://www.jbb.de/html/?page=news&id=33> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁶⁴ “The appearance of the GPL within an element of a legal claim, however, does not, in and of itself, bring into question the validity of the GPL Even when the GPL is implicated, the claims typically assume that the GPL is a legal agreement and focus instead on whether the terms of the GPL were violated.” Wacha, *supra* note 62, at 454.

⁶⁵ Free Software Foundation, *GPLv3 Draft* (Discussion Draft 1 of GPL Version 3, dated Jan. 16, 2006), <http://gplv3.fsf.org/draft> (last visited Mar. 7, 2006) [hereinafter *GPLv3 Draft*] (on file with the North Carolina Journal of Law & Technology).

not a philosophical shift, but a perceived need for additional measures to uphold GNU's existing free software philosophy.⁶⁶ It is noteworthy that GPLv3 devotes an entire new section to Digital Rights Management ("DRM").⁶⁷ DRM refers broadly to technological measures which control access to digital information.⁶⁸ Digital information, of course, now includes movies on DVD, music on CD, electronic books, or practically anything else that can be transmitted over the Internet. Given that many such items are works protected by copyright—and that copyright holders are generally interested in controlling the circulation of their works—the number of potential applications for DRM is virtually unlimited. DRM is patently unacceptable to GNU, which has relabeled the concept "Digital Restrictions Management"⁶⁹ and rejected DRM outright as incompatible with free software, stating that "DRM is fundamentally in conflict with the freedoms of users that the GPL is designed to safeguard"⁷⁰ GNU further notes that their "ability to oppose DRM by means of free software licenses is limited."⁷¹ In making an attempt to combat DRM via GPLv3, however, the added provisions are unacceptably vague and will undermine the GPL's legitimacy. The next two sections address GPLv3's DRM provisions in turn.

⁶⁶ Free Software Foundation, *Welcome to GPLv3*, <http://gplv3.fsf.org/> (last visited Mar. 24, 2006) (on file with the North Carolina Journal of Law & Technology).

⁶⁷ *GPLv3 Draft*, *supra* note 65.

⁶⁸ National Institute of Standards and Technology (NIST), *CISD: FY '02 Digital Rights Management Overview*, <http://www.itl.nist.gov/div895/drmmain.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology) (defining DRM as "technologies and services that enable legitimate owners of intellectual property to regulate the right of access to their assets via electronic means").

⁶⁹ *GPLv3 Draft*, *supra* note 65; see also *Confusing or Loaded Words and Phrases*, *supra* note 21 (arguing that the term "Digital Rights Management" is misleading).

⁷⁰ Free Software Foundation, *Rationale Document*, <http://gplv3.fsf.org/rationale> (last visited Mar. 7, 2006) [hereinafter *Rationale Document*] (on file with the North Carolina Journal of Law & Technology).

⁷¹ *Rationale Document*, *supra* note 70.

A. Privacy

The subject of privacy does not appear in Version 2, the current official version of the GPL.⁷² In contrast, GPLv3 explicitly addresses the issue, providing that “[r]egardless of any other provision of this License, no permission is given to distribute covered works that illegally invade users’ privacy”⁷³ Taking this language at its plain meaning, it marks a sharp doctrinal departure from GNU’s traditional ethos of permitting any and all uses of free software, as long as the GPL’s redistribution requirements are observed. While making clear that it views some uses of software as unethical, the FSF has in the past explicitly declined to enforce any ethical regime among users of the GPL.⁷⁴ In fact, in criticizing another license which attempts to invoke such restrictions, the GNU Project has stated that such restrictions in a software license would be both “unnecessary” and “ineffective.”⁷⁵ It goes on to claim that “[t]he GNU GPL is sufficient protection against privacy-violating features, because it ensures that someone can get the source code, find the [privacy-violating] feature, and publish an improved version of the software which does not have the feature.”⁷⁶ In early 2005, FSF went so far as to foreclose the idea that GPLv3 would contain any such software-purpose restrictions, declaring its intention to “reject restrictions on who can use free software, or what it can be used for.”⁷⁷ That such a

⁷² *GNU GPL: Version 2*, *supra* note 32.

⁷³ *GPLv3 Draft*, *supra* note 65.

⁷⁴ David Turner, *Censorship Envy and Licensing*, <http://www.fsf.org/blogs/licensing/20050211.html>, Feb. 11, 2005 (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). This piece is subtitled “Why we don’t have political terms in our licenses (even for really important issues),” and asks rhetorically: “Nuclear war is a really bad thing. It’s so bad that we want to work really hard to avoid it. As the copyright holders of a whole bunch of free software, FSF has a lot of power. So, why do we permit the use of free software in nuclear weapons?” *Id.*

⁷⁵ The GNU Project, *HESSLA*, <http://www.gnu.org/licenses/hessla.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁷⁶ *HESSLA*, *supra* note 75.

⁷⁷ Turner, *supra* note 74.

broad and explicit anti-privacy-invasion provision has suddenly found itself in GPLv3 is incongruous with its prior declarations and unexplained by FSF.

Even if GNU indeed intends to begin restricting GPL-covered software to what it considers to be “ethical uses,” this is not likely to be accomplished by the anti-privacy-invasion provision as written. The most fundamental flaw with the provision is that the terms *illegal* and *invasion* make the statement somewhat circular. A *legal* invasion of privacy is arguably not an invasion at all. Black’s Law Dictionary defines an “invasion of privacy” as “[a]n unjustified exploitation of one’s personality or intrusion into one’s personal activity”⁷⁸ Black’s definition suggests that if an *invasion* is an unjustified intrusion, then *some* intrusions *are* justified, removing justified intrusions from the category of invasion.

Furthermore, the provision as written is not particularly compatible with the stated purposes of GPL. The FSF states that opposition to certain legal changes in the past fifteen years is part of FSF’s rationale behind the revised license.⁷⁹ But in providing only for the prohibition of *illegal* invasions of privacy, GPLv3 appears to hypothetically permit the use of GPL-covered software for any invasions of privacy that an oppressive government wishes to impose on its citizens, so long as those invasions are legally within that government’s power. In other words, GPLv3 ironically begins as a response to unfavorable changes in the law, but then uses simple legality as its standard for acceptable behavior.

On closer inspection, this provision’s flaws extend beyond confusing wording. Privacy is a context-dependent concept. In the United States, warrantless governmental monitoring of a citizen’s Internet usage would risk running afoul of the Fourth Amendment, while the same activity is widely permitted when performed by an employer.⁸⁰ However, what if a warrant has been properly issued

⁷⁸ BLACK’S LAW DICTIONARY 829 (7th ed. 1999).

⁷⁹ *Rationale Document*, *supra* note 70.

⁸⁰ *Burdeau v. McDowell*, 256 U.S. 465, 475 (1921) (holding that the Fourth Amendment applies only to searches and seizures by government, not by private parties). *See also* Yohei Suda, *Monitoring E-mail of Employees in the Private*

for a legal wiretap on a citizen? What if a citizen has consented to being monitored, or if the party with whom the citizen is communicating has so consented? Whether the monitoring in any of these cases is legal is highly fact-dependent. Unfortunately, any software being used to monitor an individual's Internet usage will be unable to even ascertain the specific facts of any given instance, much less adequately take them into consideration.

In addition to being a context-dependent concept, privacy lacks a consistent meaning across jurisdictional boundaries. For example, while the Charter of Fundamental Rights of the European Union recognizes an individual's right "to respect for his or her private and family life, home, and communications"⁸¹ and "to the protection of [his] personal data,"⁸² there is no such explicit privacy right in the United States Constitution. Privacy guarantees in the United States are instead based on "certain areas or zones of privacy."⁸³ Because "privacy" has no universal legal meaning, the concept is too nebulous to be effectively protected by a software license.

An additional problem is that this provision does not adequately explain who constitutes a "user." Is the provision meant to protect only the privacy of the person running the program, for example, by prohibiting GPL-covered "spyware?"⁸⁴

Sector: A Comparison Between Western Europe and the United States, 4 WASH. U. GLOBAL STUD. L. REV. 209, 234-40 (2005) (commenting that the Electronic Communications Privacy Act of 1986, while prohibiting certain unauthorized interceptions of electronic communications, contains exceptions that permit an employer to monitor an employee's email under a broad range of conditions).

⁸¹ The Charter of Fundamental Rights of the European Union, art. 7, available at http://www.europarl.eu.int/charter/pdf/text_en.pdf (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁸² *Id.* at art. 8.

⁸³ See *Roe v. Wade*, 410 U.S. 113, 152 (1973) (noting that while the U.S. Constitution does not explicitly mention any right of privacy, a right of personal privacy is nonetheless recognized by the Constitution).

⁸⁴ "Installed on your computer without your consent, spyware software monitors or controls your computer use. It may be used to send you pop-up ads, redirect your computer to websites, monitor your Internet surfing, or record your keystrokes, which, in turn, could lead to identity theft." Federal Trade Commission, *Spyware*, <http://www.ftc.gov/bcp/conline/pubs/alerts/spywarealrt>.

Or does the provision also cover the privacy of users other than the person running the program? Consider that several GPL-covered programs are designed for activities that can be either legitimate or privacy-invasive, depending on how they are used. For example, some “network scanners” are covered by the GPL.⁸⁵ These programs can be used to determine how many and what kinds of computers are present on a network, typically without the knowledge of other users.⁸⁶ This activity is neither ethical nor unethical in the abstract. It can be carried out by a network administrator who needs to know what has attached itself to the network, or it can be carried out by a criminal as a first step in a break-in. Another dual-use tool is a “password cracker,”⁸⁷ which can be used for either recovering a lost password for a user (ethical) or for guessing another’s password without permission (unethical). Another example is a “packet sniffer,”⁸⁸ which can be used for either analyzing raw network traffic between two machines to find communication problems (ethical) or for silently intercepting network traffic to look for passwords or other private information (unethical). Some of these programs are also GPL-covered, yet GPLv3’s privacy provision does not make a distinction between dual-use tools and the uses to which such tools are put. A broad reading of GPLv3’s anti-privacy-invasion provision arguably will prohibit the distribution of *any* program that *may* be used as a tool for privacy invasion.

htm (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). “In the field of computing, the term spyware refers to a broad category of malicious software designed to intercept or take partial control of a computer’s operation without the informed consent of that machine’s owner or legitimate user.” Wikipedia, *Spyware*, <http://en.wikipedia.org/wiki/Spyware> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁸⁵ See, e.g., *Nmap—Free Security Scanner for Network Exploration & Security Audits*, <http://www.insecure.org/nmap/index.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁸⁶ *Nmap*, *supra* note 85.

⁸⁷ See, e.g., *Lepton’s Crack*, <http://usuarios.lycos.es/reinob/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁸⁸ See, e.g., *Neteclipse*, <http://sourceforge.net/projects/neteclipse/> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

In attempting to prohibit privacy-invading programs, the FSF has failed to recognize that it is people, not programs, who invade privacy. A piece of software cannot be held liable for an act. GPLv3 ignores the distinction between *tools* that can be used to carry out digital privacy invasion and the *act* of invasion itself. Thus, a number of software tools with legitimate uses, presently distributed under the GPL, would risk running afoul of GPLv3.

Apart from present-day programs, there are also potential conflicts with programs that work in conjunction with emerging technologies. For example, the use of biometric technology⁸⁹ in security and surveillance activities has raised a number of privacy issues.⁹⁰ If some aspect of an emerging technology is judged by the FSF to be “privacy-invasive,” a strict reading of GPLv3 implies that no GPLv3-covered software may be used for that purpose. Declaring entire fields of endeavor to be off limits for GPLv3-covered software in turn raises another problem—determining when a connection to such a field is attenuated enough to fall within GPLv3’s permissions. For example, how should the use of a GPLv3-covered word processor by a government surveillance agency be evaluated? Does the agency violate GPLv3’s privacy provisions when it uses the word processor to maintain lists of surveillance subjects? Or is *any* use of the software in a “privacy-invasive” field in turn a “privacy-invasive” use? GPLv3 provides no guidance in this area.

⁸⁹ “Biometrics are automated methods of recognizing a person based on a physiological or behavioral characteristic. Biometric technologies are becoming the foundation of an extensive array of highly secure identification and personal verification solutions.” National Institute of Standards and Technology (NIST), Information Technology Laboratory, *About Biometrics*, <http://www.itl.nist.gov/div893/biometrics/about.html> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁹⁰ “We are particularly worried about [The International Civil Aviation Organization’s] plans requiring passports and other travel documents to contain biometrics . . . the implementation of biometrics will have disproportionate effects on privacy and civil liberties.” Privacy International, *An Open Letter to the ICAO*, Mar. 30, 2004, <http://www.privacyinternational.org/issues/terrorism/rpt/icaoletter.pdf> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

B. “Effective Technological Protection Measure[s]”

A comparison of the key words in the second part of the “Digital Restrictions Management” provision of GPLv3 suggests that it is intended as a frontal attack on the Digital Millennium Copyright Act (DMCA).⁹¹ The DMCA is designed to provide additional protections to copyright holders who protect their works using technological measures,⁹² and provides that “[n]o person shall circumvent a *technological measure* that *effectively* controls access to a work *protected* under [copyright].”⁹³ GPLv3 provides that:

“[n]o [GPL-]covered work constitutes part of an *effective technological protection measure*: that is to say, distribution of a covered work as part of a system to generate or access certain data constitutes general permission at least for development, distribution and use, under this License, of other software capable of accessing the same data.”⁹⁴

The intent, evidently, is that GPL-covered software may never be used as a tool for furthering the purposes of the DMCA. The “Rationale Document” for GPLv3 lends some support to this reading, noting that “[t]he second paragraph of section 3 declares that no GPL-covered program is part of an effective technological protection measure, regardless of what the program does.”⁹⁵ However, this provision suffers from at least two deficiencies: (1) it is ambiguous about exactly what a GPLv3 licensee is required to make available; and (2) under a practical interpretation of that requirement, it raises problems for those wishing to distribute GPL-covered works alongside non-GPL-covered works—a right which the license claims to grant.

1. Ambiguity of the “General Permission” Requirement

As an example of the first deficiency, consider a hypothetical medical information management system for small offices, which employees use via their web browser on the office network. The

⁹¹ 17 U.S.C. § 1201 (2000).

⁹² MERGES, *supra* note 5, at 500.

⁹³ 17 U.S.C. § 1201(a)(1)(A) (2000) (emphasis added).

⁹⁴ GPLv3 Draft, *supra* note 65 (emphasis added).

⁹⁵ Rationale Document, *supra* note 70.

system is developed and sold as a unit by a system supplier. It consists of: (1) a personal computer running the Linux operating system, which is covered by the GPL; (2) additional open-source software packages, including a web server and database server, which are either covered by the GPL or a free software license compatible with the GPL;⁹⁶ and (3) additional, proprietary software developed by the system supplier, custom written for managing medical records.⁹⁷ Office employees use the system to maintain information about new and existing patients, including names, addresses, and medical histories. As GPLv3 leaves most of the terms in the provision undefined, one can reasonably construe the product as constituting “a system” under the provision, with Linux (and any other GPL-covered software present) constituting “covered works” that are part of that system, and the medical information it manages to constitute “certain data” which is “generated” and “accessed” by the system.⁹⁸ However, what the system supplier must actually permit or make available, in order to grant this “general permission” for development of software capable of “accessing the same data,” is not specified.

A literal interpretation suggests that other developers should have access to the medical data itself. This is undoubtedly not the intent, particularly given GPLv3’s preoccupation with privacy, as discussed above. A more realistic reading is that the system supplier must give permission to utilize the same data *format*; that is, that the internal structure of the data cannot be kept proprietary, but must be made open to anyone interested in learning the

⁹⁶ The FSF maintains a list of licenses it considers to be “compatible” with the GPL. See *FSF—Licenses*, http://www.fsf.org/licensing/licenses/index_html (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁹⁷ This is a common web application architecture, and is often referred to as “LAMP,” or “Linux, Apache, MySQL, and Perl / PHP / Python.” See Data & Analysis Center for Software, *DoD and Information Technology (DoD/IT) Acronyms (Acronyms Beginning with “L”)*, http://www.dacs.dtic.mil/databases/acronym/acronym_display.htm?beginAcronym=L (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

⁹⁸ Of these key terms, GPLv3 explicitly defines only “covered work,” as “either the GPL-covered Program or any work based on the [GPL-covered] Program.” *GPLv3 Draft*, *supra* note 65.

structure. This reading is consistent with the FSF philosophy that free software must include “the freedom to study how the program works, and adapt it to your needs . . . [for which] [a]ccess to the source code is a precondition”⁹⁹ However, this result is inconsistent with another key GPL provision: that non-GPL-covered works, even proprietary works, may be distributed along with GPL-covered works as long as the GPL-covered and the non-GPL-covered portions are kept sufficiently separate from one another.

2. Conflict with the “Mere Aggregation” Provisions

Under the more realistic reading of the “general permission” requirement, requiring that details about the internal workings of non-GPL-covered works be made available when they are distributed together with GPL-covered works, it is dubious whether a proprietary product such as the specialized medical information management program as described above can be distributed and remain proprietary.¹⁰⁰ Section 2 of GPL Version 2 specifies that source code must be supplied when modified versions of GPL software are distributed.¹⁰¹ However, this requirement is limited to works either: (1) derived from the GPL-covered program itself; or (2) separate from the GPL-covered program, but distributed as part of another program which is based on the GPL-covered program.¹⁰²

Other works that are not based on, or an extension of a GPL-covered program, but instead are separate, independently-created works may be distributed with GPL-covered programs without

⁹⁹ *Free Software Definition*, *supra* note 17.

¹⁰⁰ At least one individual believes that by definition, *all* medical software should be covered by the GPL. “Releasing medical software under a proprietary license is very wrong, like lying to a priest. There are always consequences when a proprietary (non-GPL) software is used in the place of GPL software. In medicine, these consequences mean lost lives.” GPL Medicine, *The Morality of Licensing in Medicine*, <http://www.gplmedicine.org/index.php?module=htmlpages&func=display&pid=3> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

¹⁰¹ *GNU GPL: Version 2*, *supra* note 32.

¹⁰² *GNU GPL: Version 2*, *supra* note 32.

falling within this requirement.¹⁰³ GNU has further clarified this “mere aggregation” provision, going so far as to state that simply because two programs—for instance, a GPL-covered database server and a proprietary program to retrieve data from that database—may interoperate by communicating with each other, they are not combined into one program for the purposes of the GPL.¹⁰⁴ This provision appears with different wording in Section 5 of GPLv3,¹⁰⁵ but the “Rationale Document” declares that “the underlying meaning is unchanged.”¹⁰⁶ If that is the case, it is unclear how one is still free to distribute proprietary works together with GPL-covered works, while simultaneously granting permission to “access” anything that the proprietary program would access. GPLv3 gives no further guidance as to what constitutes this general permission or how it is to be granted, leaving much room for misinterpretation.

IV. PROPOSED REVISIONS

Assuming that FSF does indeed intend to give GPLv3 additional privacy-protection power, the license will have a better chance at success in protecting privacy with a provision more limited in scope. FSF first must recognize that privacy invasion is ultimately carried out by institutions and individuals, not software programs, and software licenses will thus be largely ineffective at preventing the problem. Problems can only be *contained* by software to the extent that those problems are actually *caused* by software. For instance, a GPLv3 prohibition on self-propagating programs such as viruses or worms, which spread to other computers and perform actions without the user’s knowledge, would survive under the criteria of GNU-style freedom. GPLv3 may reasonably provide that each “distribution” permitted by the license must be at the hands of user action, not some self-

¹⁰³ *GNU GPL: Version 2, supra* note 32.

¹⁰⁴ The GNU Project, *Frequently Asked Questions about the GNU GPL*, <http://www.gnu.org/licenses/gpl-faq.html#MereAggregation> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology).

¹⁰⁵ *GPLv3 Draft, supra* note 65.

¹⁰⁶ *Rationale Document, supra* note 70.

propagating “distribution” by the work itself, outside the control of the user. While it is both impractical and contrary to the purposes of GNU and the GPL to restrict the tasks that software may be written to perform, the GPL has been quite successful in defining how humans may distribute GPL-covered works. A “no automatic propagation” provision can capitalize on this success and address privacy issues at the same time.

In the area of “effective technological protection measures,” a better solution to upholding the GPL’s philosophy is to not place the GPL at permanent odds with the DMCA, but to encourage the use of GPL-covered software in such applications. As discussed above, some device manufacturers have discovered economic advantages to sharing their work, and GPLv3 must encourage rather than discourage this trend. While those seeking effective technological protection measures may not turn out to use and share GPL-covered software in large numbers, there is no reason to forbid the practice. Open-minded manufacturers may well find a way to balance their proprietary interests with the interests of consumers and developers of GPL-covered software. There is no technical reason that a GPL-covered program cannot protect proprietary data while the protection program itself remains free and open.¹⁰⁷ Such a balance would both maintain the integrity of the GPL’s software-purpose neutrality and encourage more openness on the part of device makers.

¹⁰⁷ This approach is exemplified by the GNU Privacy Guard Program (GNUPG), which is itself free software, but designed to keep other data private. See *GNUPG Frequently Asked Questions*, [http://www.gnupg.org/\(en\)/documentation/faqs.html](http://www.gnupg.org/(en)/documentation/faqs.html) (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology). Nothing in the GNUPG documentation suggests that any resulting, encrypted data created or accessed by GNUPG are subject to the terms of the GPL, or need be otherwise shared. *Id.* See also The GNU Project, *Frequently Asked Questions about the GNU GPL*, <http://www.gnu.org/licenses/gpl-faq.html#WhatCaseIsOutputGPL> (last visited Mar. 7, 2006) (on file with the North Carolina Journal of Law & Technology) (asking the question “In what cases is the output of a GPL program covered by the GPL too?” and providing the answer, “Only when the program copies part of itself into the output.”).

V. CONCLUSION

The “Digital Restrictions Management” section of the proposed GPLv3 is a philosophical detour masquerading as a license provision. While the motivations behind the new provision—privacy protection and continued freedom for free software—are praiseworthy, the provision is an inappropriate mixture of rigid and vague guidelines. It is not the type of provision previously seen in the successful GPL, and certainly not something that belongs in any new version of the license. The provision introduces ambiguities to the license and, thus, threatens the license’s enforceability.¹⁰⁸ Much of the GPL’s historical success has been due to its recognition that legal and technological realities create a harsh environment for free software. Instead of adapting itself to legal realities as in previous versions of the GPL, the “Digital Restrictions Management” provision in GPLv3 disregards this reality. The inclusion of such a provision in the new license will be ineffective at best, and counterproductive at worst, resulting in a weaker legal instrument. GPLv3 can best continue GPL Version 2’s success by remaining unambiguous in its goals, remaining mindful to present legal realities, and limiting its mission to those goals it can realistically achieve.

¹⁰⁸ This result is succinctly summarized in the doctrine of *contra proferentem*: “[I]n interpreting documents, ambiguities are to be construed unfavorably to the drafter.” BLACK’S LAW DICTIONARY 328 (7th ed. 1999).