

## How Will Version 3 of the GPL Affect the Technology Industry?

Douglas E. Phillips

The 16<sup>th</sup> Annual Seattle Conference on New Developments in Technology Law  
Law Seminars International  
December 14, 2007

On June 29, 2007, Apple released the iPhone. On the same day, the Free Software Foundation (“FSF”) released version 3 of the GNU General Public License (“GPLv3”). No one had to stand in line for a copy of GPLv3. (It’s free for the taking at <http://www.gnu.org/licenses/gpl-3.0.txt>.) And GPLv3 probably appears on fewer Christmas lists. Still, like the iPhone, GPLv3 has been called a turning point for the technology industry. What remains to be seen is whether it will live up to expectations – or whether, in the marketplace of software licenses, it will flop.

### A. Why GPLv3?

According to FSF, the GPL “is the most widely used free software license worldwide: almost three quarters of all free software packages are distributed under this license” (<http://www.fsf.org/news/gpl3dd4-released>). The best known package distributed under the GPL is, of course, Linux (which FSF prefers to describe as the “Linux kernel” of the “GNU/Linux operating system”). Whether one views the GPL’s “viral” character as a disease, or as a vector for curing the ills of “proprietary” software, the GPL clearly has succeeded in propagating itself far and worldwide. Why, then, has FSF chosen to introduce a new and significantly revised version?

FSF states, in the Quick Guide, that GPLv3 “protects users from three recent threats.” The Guide describes these “threats” as follows:

- “[t]ivoization,” FSF’s name for the practice of preventing users from running modified GPL-licensed software on a particular hardware device
- “[l]aws prohibiting free software” – more specifically, the Digital Millennium Copyright Act (“DMCA”) and similar laws that “make it a crime to write or share software that can break DRM,” and
- “[d]iscriminatory patent deals” – and more broadly, practices that enable patent holders “to collect royalties for the use of free software.”

---

© 2007 Douglas E. Phillips. The author is Vice President and General Counsel of Promontory Interfinancial Network and author of *The Software License Unveiled: How Legislation by License Controls Software Access*, forthcoming in 2008 from Oxford University Press. He has represented proprietary software companies in the past, but is not currently doing so, and the views in this paper are solely his own.

GPLv3 contains new terms relating to these issues, and this paper addresses them. But these provisions are not the only reasons for the new version. Just as the iPhone brings a more intuitive user interface to the mobile phone, one of the stated objectives of GPLv3 is to improve the GPL's ease of use. In "A Quick Guide to GPLv3" (<http://www.fsf.org/licensing/licenses/quick-guide-gplv3.html>), FSF states:

Version 3 . . . has a number of improvements to make the license easier for everyone to use and understand. . . . Though a lot of text has changed, much of it simply clarifies what GPLv2 said. . . .

The importance of ease of use becomes clear when one considers "license proliferation." The Open Source Initiative ("OSI") currently lists 65 approved open source licenses (<http://www.opensource.org/licenses/alphabetical>). In the subset of open source software known as "free software," FSF lists 34 licenses that it considers "compatible" with the GPL and an additional 33 licenses that it considers incompatible with the GPL, but that nevertheless qualify as "free software licenses" (<http://www.fsf.org/licensing/licenses>).<sup>1</sup>

As the numbers suggest, license proliferation – now the subject of an OSI advisory committee study (<http://opensource.org/proliferation>) – means that the GPL faces more competition than in the past. FSF wants to maintain the GPL's position of leadership in free and open source software licensing, and trying to promote use of the GPL by improving the "user experience" is a natural approach.

Nevertheless, the two sets of objectives – protecting users from "recent threats" and improving ease of use – appear to be in tension. To address the threats, FSF has added a number of new provisions. As the word "threats" implies, this part of the effort reflects strongly-held views. For example, FSF states in an FAQ response (<http://www.fsf.org/licensing/licenses/gpl-faq.html>): "[A]nyone considering enforcing their patents aggressively [against free software] is an enemy of the community, and we will defend ourselves against such an attack."

For FSF, defending the community against its enemies appears to have taken priority over simplifying the license. The effect has been to produce a substantially longer and more complex document. In this respect, GPLv3 continues an expansionary trend that began with the last updating of the GPL, 16 years ago.

GPL version 1 ("GPLv1") (<http://www.gnu.org/licenses/old-licenses/gpl-1.0.txt>) appeared in February 1989. GPLv1, including the title, the copyright notice, and a how-to appendix, contains just over 2,000 words. The main elements of the GPL – royalty-free rights to use and modify, access to source code, an obligation to redistribute if at all under the GPL, and a disclaimer of warranties – are all there. The tone is light, with an example involving a Mr. James Hacker and a Mr. Ty Coon, President of Vice. The last sentence of the appendix reads: "That's all there is to it."

---

<sup>1</sup> FSF also lists yet another 24 licenses that it says are "often mistaken for free software licenses."

GPL version 2 (“GPLv2”) (<http://www.gnu.org/licenses/old-licenses/gpl-2.0.txt>), which appeared in June 1991, contains almost 3,000 words – making it about one and a half times the length of version 1. “That’s all there is to it” no longer appears. GPLv3 did not arrive until 16 years after GPLv2. GPLv3 contains more than 5,500 words (which do not include the names of James Hacker and Ty Coon, both of whom have been retired). This makes the new version almost twice the length of its predecessor and almost three times the length of the first GPL.

In contrast, version 2 of the “Do What the Fuck You Want to Public License” (“WTFPL”) (<http://sam.zoy.org/wtfpl>), dated December 2004, contains only 79 words. Its operative provision contains only nine: “You just DO WHAT THE FUCK YOU WANT TO.” FSF reportedly acknowledges that the WTFPL is a valid free software license (<http://en.wikipedia.org/wiki/WTFPL>).

As shown in table 1, the differences in word count are also accompanied by differences in readability, as measured by common tests. The table shows the results, obtained using an adaptation of the Lingua:EN:Fathom PERL module (at <http://writing.teiru.net/fog>), for the three versions of the GPL, together with the WTFPL and the Microsoft Windows Vista end user license agreement (“EULA”).

Table 1: License Readability

License	Word Count	Flesch Readability	Flesch-Kincaid Grade Level
GPLv1	2017	39.75	14.77
§ 1 of GPLv1	88	15.09	22.09
GPLV2	2925	36.62	15.72
§ 1 of GPLv2	98	14.66	23.39
GPLv3	5567	33.01	15.34
§ 4 of GPLv2 <sup>2</sup>	100	17.34	23.26
MS Windows Vista EULA	5568	45.24	9.55
§ 2 of EULA <sup>3</sup>	158	56.39	8.01
WTFPL operative term	9	113.10	-0.28

For Flesch readability, a higher score is better. For Flesch-Kincaid grade level, the score indicates the grade level for which the text is likely to be understandable. As the table shows, these measurements indicate that the readability of the GPL has declined with each version. To be likely to understand section 4 of GPLv3, one of the main

<sup>2</sup> Section 4 of GPLv3 is the successor to section 1 of GPLv1 and GPLv2.

<sup>3</sup> Section 4 of the Vista EULA is the main license-granting part.

permission-granting provisions, a reader may require as many as 23 years of formal education.<sup>4</sup>

It might be unrealistic to expect the GPL to achieve the (witty) brevity of the WTFPL. Still, any review of the GPLv3's substantive terms should also take into account the effects of its length and complexity. Like other "upgrades," GPLv3 is competing not only with other licenses but with itself – more specifically, with GPLv2. Some key players, including Linux-creator Linus Torvalds, have expressed a preference for version 2 (<http://www.ussg.iu.edu/hypermil/linux/kernel/0706.2/1928.html>).<sup>5</sup> Because GPLv3 restates the core provisions of GPLv2, a first step is to consider how clearly and effectively it does so.

### B. GPLv3 and the Four Freedoms

When Richard Stallman, President of FSF, announced the release of GPLv3, he pointed out that the GPL is designed to protect four "essential freedoms" ([http://gplv3.fsf.org/rms\\_gplv3\\_launch\\_transcript](http://gplv3.fsf.org/rms_gplv3_launch_transcript)). They can be summarized as follows:

0. Freedom to run the program.
1. Freedom to study and change the source code.
2. Freedom to distribute exact copies.
3. Freedom to distribute modified copies.

These four freedoms – and freedom from any royalty obligation in exercising them – have been part of the GPL from the start. Why, then, is GPLv3 so much longer? The answer lies, in part, in the scope of the license and in the effort to address practices that FSF considers abusive. But significant change has also occurred in how GPLv3 implements some of the basic concepts.

#### 1. Freedom to Run and Freedom to Distribute Unmodified Copies

GPLv2 implements freedom 0 – the freedom to run the program – by stating, in section 0, that "[t]he act of running the Program is not restricted."<sup>6</sup> In GPLv3, section 2 states that the GPL "explicitly affirms your unlimited permission to run the unmodified Program."<sup>7</sup> Although the version 3 formulation adds a few words, it does not substantially change how the GPL gives effect to freedom 0.

---

<sup>4</sup> In contrast, the WTFPL is extremely readable and likely to be understood by a child just entering the first grade (leaving aside the use of a word that some starting first-graders do not yet know).

<sup>5</sup> Under section 14, GPLv3 is not retroactive, and its use is optional, but it can be applied to new contributions to existing works. If that occurs, distribution of future versions of the works will be subject to GPLv3.

<sup>6</sup> GPLv2 defines the "Program" as "any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License."

<sup>7</sup> GPLv3 defines the "Program" as "any copyrightable work licensed under this License."

GPLv2 and GPLv3 use identical language for freedom 2 – the freedom to distribute (unmodified) copies. Section 1 of GPLv2 and section 4 of GPLv3 both state that, on specified conditions, “[y]ou may copy and distribute verbatim copies of the Program's source code as you receive it . . . .” Both versions (GPLv2 in section 3 and GPLv3 in section 4) also permit the licensee to copy and distribute verbatim copies in object code if the licensee also provides or makes available the source code.

## 2. Freedom to Modify and to Distribute Modified Copies

Versions 2 and 3 of the GPL diverge in their implementations of freedoms 1 and 3 – the freedom to study and change the source code and the freedom to distribute modified copies. Because the feature that perhaps best distinguishes free and open source software from proprietary software is the grant of permission to study and change the source code and to distribute modified copies, these provisions are key.

GPLv2 grants permission, on stated conditions, to “modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work . . . .” The most important condition of the permission to “copy and distribute” modifications – and the source of the GPL’s “viral” effect – is section 2(b), which states:

You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

GPLv3 states, in section 0, that to “modify” a work means “to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy.” But GPLv3 deletes the GPLv2 statement that “[y]ou may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work . . . .” How, then, does GPLv3 give effect to freedom 1 – the freedom to modify?

## 3. Freedom to Modify in GPLv3

Surprisingly, GPLv3 appears to contain a rather large drafting error – or at least a significant convolution – in the provisions that grant permission to modify covered works. As the Preamble makes clear, GPLv3 is obviously intended to permit modification. For example, the Preamble states: “The GNU General Public License is intended to guarantee your freedom to share and change all versions of a program.” But as open source advocate Lawrence Rosen has noted: “The preamble, of course, is not an operative part of the GPL license. It is not among its *terms and conditions*. There is nothing in its words that must be obeyed.”<sup>8</sup>

---

<sup>8</sup> Lawrence Rosen, *Open Source Licensing: Software Freedom and Intellectual Property Law* 109 (2004) (emphasis in original).

In GPLv3, section 8 of the actual Terms and Conditions declares: “You may not propagate or modify a covered work except as expressly provided under this License.”<sup>9</sup> In other words, the right to modify is not unlimited, and there is no implied license to modify a covered work. Permission must be found in a provision that expressly grants it.

Section 2 of GPLv3 provides: “You may make, run and propagate covered works that you do not convey, without conditions . . .” Section 0 states that to “propagate” a work means to do anything with the work that, without permission, would make the licensee liable for copyright infringement, except running it or “modifying a private copy.” Because modifying a work, under the definition quoted earlier, means copying from it or adapting it in a way that requires copyright permission, propagation appears to include modification (as long as it is modification of something other than a “private copy”).<sup>10</sup>

Therefore, section 2 appears to grant permission to modify covered works – but it grants such permission only for “covered works *that you do not convey*.” (Emphasis added.)<sup>11</sup> Nowhere does GPLv3 explicitly permit the licensee – with or without conditions – to modify covered works that the licensee *does* convey.<sup>12</sup>

Section 5 permits the licensee, on stated conditions, to convey “a work based on the Program, or the modifications to produce it from the Program, in the form of source code,” and Section 6 permits the licensee to convey a covered work in object code if source code is also provided.

Under the Copyright Act, however, the right to *distribute* copies of works to the public, provided by 17 U.S.C. § 106(3), is separate and distinct from the right to *prepare derivative works* based on the copyrighted work, provided by 17 U.S.C. § 106(2). And under the GPL, the right to *convey* the Program or a work based on the Program does not include the right to *modify* the Program or a work based on the Program. To “convey”

---

<sup>9</sup> Section 8 continues: “Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses . . .).”

<sup>10</sup> Section 8 – by twice linking “propagate” and “modify” with the disjunctive “or” – could be taken to imply that that propagation does not include modification. The definition of “propagate,” on the other hand, suggests that, in some cases, it does. Although the license language is unclear, this analysis assumes that propagation does include modification of a non-“private copy.” The conclusion of the analysis would not change if propagation did not include modification.

<sup>11</sup> Section 0 states: “To ‘convey’ a work means any kind of propagation that enables other parties to make or receive copies.”

<sup>12</sup> One might argue that “works that you do not convey” means “works that you do not simultaneously (at the time of modification) convey.” In that case, as long as you modify first and convey later, you are (arguably) within the permission. Apart from the fact that the word “simultaneously” does not appear, however, presumably most if not all conveying of modified works would occur only after the modification has occurred, rather than while it is occurring, in which case the clause “that you do not convey” would seem pointless. Moreover, the permission is granted “without conditions,” and certain conditions do seem intended to apply to modified works that are subsequently conveyed (such as conditions relating to the preservation of notices). For all these reasons, “works that you do not convey” seems to apply to works that you do not convey at any time and thus not to works that you do subsequently convey.

Table 2: The “Four Freedoms” in GPLv2 and GPLv3

	GPLv2	GPLv3
0. Freedom to run the program.	§ 0. Running the Program is not restricted.	§ 2. Affirms unlimited permission to run unmodified Program.
1. Freedom to study and change the source code.	§ 2. “You may modify your copy or copies of the Program or any portion of it” with conditions.	§ 2. “You may . . . propagate covered works that you do not convey, without conditions . . . .”  No explicit permission to modify covered works that you do convey.
2. Freedom to distribute copies.	§ 1. “You may copy and distribute verbatim copies of the Program’s source code as you receive it” with conditions.  § 3. “You may copy and distribute the Program . . . in object code or executable form” with conditions, including source code.	§ 4. “You may convey verbatim copies of the Program’s source code as you receive it” with conditions.  § 4. You may convey the work in object code form, with conditions, if “you also convey the machine-readable Corresponding Source.”
3. Freedom to distribute modified copies.	§ 2. “You may . . . copy and distribute [your] modifications or work” based on the Program, with conditions.	§ 4. “You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code,” with conditions, and you may also convey the work in object code if you convey source code.
The “viral” condition.	§ 2-b) “You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.”	§ 4-c) “You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy.”

refers only to propagation “that enables other parties to make or receive copies,” but modifying a work does not, in and of itself, so enable others.

Section 10 states: “Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License.” But this clause (which is part of a section that deals with patents) does not grant any license to the conveying licensee – the “you” addressed in the clause. Nor does it grant a license from the conveying licensee to the recipient permitting the recipient to modify the conveying licensee’s contribution, if any.<sup>13</sup>

As noted above, the apparent absence of express permission in GPLv3 for the licensee to modify a covered work that the licensee conveys can only be a mistake. Nevertheless, the difficulty tracking this core concept through the complex new language indicates that the most significant practical effect of GPLv3 on the industry, at least in the near term, may be to keep technology lawyers even busier than they were with GPLv2.

### C. GPL and the Three Threats

The expansion in the length and complexity of the GPL is not limited to the parts that restate existing provisions. The broader scope of GPLv3 also reflects the extension of intellectual property protection for software into the patent realm, as well as perceived abuses in areas such as DRM – which, depending on one’s point of view, refers to “digital rights management” or “digital restrictions management.”

#### 1. “Tivoization”

In 1989, when GPLv1 appeared, the consumer who wanted to record a television program did so by “taping” a standard-definition analog broadcast on a video cassette recorder (“VCR”). Today, the broadcast is in high definition and digital, and the VCR has given way to the digital video recorder (“DVR”). The TiVo DVR distributed by Tivo, Inc., like many current consumer devices, is a computer with an operating system.

Although the TiVo DVR client application is proprietary software, the DVR’s operating system includes modified parts of the Linux kernel and the GNU/Linux operating system. Complying with GPLv2, TiVo provides the source code for the modifications on its website and on disc (<http://dynamic.tivo.com/linux/linux.asp>). But the TiVo DVR also checks for a digital signature in the software. As a result, if a user further modifies

---

<sup>13</sup> At most, the quoted clause grants permission from the “original licensors” of the work to a subsequent downstream recipient of the work – a person to whom the licensee, referred to as “you,” conveys the work. It is unclear, however, whether even a subsequent downstream recipient receives an effective grant under this clause. Once the initial licensee conveys the work to a recipient, the recipient becomes “you,” the licensee, and the clause, in referring to a recipient to whom “you” convey the work, refers to a further downstream recipient. The licensee in any given instance of the license is always a step upstream of the potential beneficiary – and thus, like Tantalus, never quite reaches the benefit of the clause.

TiVo's GPL-licensed software, the user-modified version – lacking the appropriate digital signature – will not run on the TiVo device.

According to FSF's Quick Guide, "[t]ivoization is a dangerous attempt to curtail users' freedom" because it prevents users from running the modified code that the GPL permits them to create on the device for which it is designed. Others have maintained that enabling users to run modified versions of the operating system on the TiVo device could allow them to defeat usage tracking and control features. It has been suggested that such an ability could, in turn, affect TiVo's ability to charge for its services and its ability to make and fulfill DRM commitments to content providers.

The "anti-tivoization" provision of GPLv3 appears in the second part of section 6. The first part of this section authorizes the GPL licensee to convey covered works in object code form if the licensee also conveys machine-readable Corresponding Source. With some qualifications, "Corresponding Source" is defined as "all the source code needed to generate, install, and . . . run the object code and to modify the work, including scripts to control those activities."

Section 6 imposes an additional requirement for object code conveyed for use in a "User Product," which is defined as a "consumer product" or "anything designed or sold for incorporation into a dwelling." Unless the object code for the User Product is on a read-only memory chip or for some other reason cannot be modified by anyone, the Corresponding Source must be accompanied by "Installation Information." Such information includes "any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source." In other words, the manufacturer can require a key to run software on the device, but must give the user the key to run a modified version.

One result, if the manufacturer complies, is that the user can cause the device to function in a way the manufacturer did not intend. FSF acknowledges that the manufacturer should not be responsible for resulting performance problems. Section 6 provides that the conveyer of the GPL-licensed software is not required "to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed."

Section 6 also states that "[a]ccess to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network." It is not clear, however, whether this provision permits the conveyer to deny access to a network to enforce payment for services or DRM restrictions. This potential uncertainty poses significant questions for any use of software licensed under GPLv3 in devices offered on a fee-for-services basis.

For example, if a user modifies the operating system of a device so that the device no longer reports usage to the service provider, and as a result the user receives the service free, is the modification one that "itself materially and adversely affects the operation of

the network”? Does it violate “the rules and protocols for communication across the network”? The inability to charge for the service may ultimately have a significant economic impact that affects the network, but the software modification “itself” may not impair the “operation” of the network in a technical sense. Likewise, the modified version may comply with “the rules and protocols for communication” at an operational level, so that it can in fact communicate, even though it does not comply with the substantive business or legal rules.

TiVo, Inc., apparently considers the risks of GPLv3 potentially serious enough to preclude the use of software licensed under it. In its Form 10-K Annual Report for 2006, filed April 16, 2007 (<http://investor.tivo.com/sec.cfm?DocType=Annual&Year=2007>), TiVo stated, under “risk factors,” as follows: “We could be prevented from selling or developing our TiVo software if the GNU General Public License governing the GNU/Linux operating system and Linux kernel and similar licenses under which our product is developed and licensed is not enforceable or changed substantially.” More specifically: “If the currently proposed version of GPLv3 is widely adopted, we may be unable to incorporate future enhancements to the GNU/Linux operating system into our software, which could adversely affect our business.”

This statement – that TiVo could be unable to incorporate future enhancements to the GNU/Linux operating system in its software – indicates that TiVo may not be prepared to subject itself to GPLv3. In that case, TiVo’s choices are to continue using existing versions or to cease using any GPL-licensed software to which GPLv3 applies. Over the long term, if GNU/Linux becomes subject to GPLv3, TiVo presumably will switch to different operating system software.

Other manufacturers of user products that use GPL-licensed software, if they wish to retain the ability to update the software on their devices but do not want users to be able freely to run user-modified versions, face similar choices. The effect could be that, rather than enabling users to modify GPL-licensed code on such devices, GPLv3 will discourage the use of GPL-licensed code on the devices in the first place.

## 2. DRM

At the GPL’s birth in 1989, the practice now known as DRM consisted largely of all-or-none “copy protection.” This was achieved by software means or by hardware devices such as “dongles.” By the late 1980s, most PC software companies, at least in the United States market, had largely done away with copy protection for competitive reasons. Digital content was still largely in the future. Not surprisingly, GPLv1 and GPLv2 are silent on DRM technology.

All that has changed. The “Quick Guide” describes the second recent threat that GPLv3 addresses as “[l]aws prohibiting free software” and cites DMCA as an example. Whatever one thinks of DMCA, however, it seems rather clearly to be an overstatement to refer to DMCA as “prohibiting free software.” DMCA provides: “No person shall circumvent a technological measure that effectively controls access to a work protected

under this title” (*i.e.*, by copyright). 17 U.S.C. § 1201(a)(1)(A). Nothing in DMCA *requires* that an effective technological measure (“ETM”) be used.

Nevertheless, DMCA does prohibit the use of free software – or any other software – to circumvent DRM that qualifies as an ETM. An FSF FAQ indicates that section 3 of GPLv3 is designed to reverse that prohibition when the DRM has been implemented using software licensed under GPLv3(<http://www.fsf.org/licensing/licenses/gpl-faq.html>). The FAQ item states:

Does GPLv3 prohibit DRM?

It does not; you can use code released under GPLv3 to develop any kind of DRM technology you like. However, if you do this, section 3 says that the system will not count as an effective technological “protection” measure, which means that if someone breaks the DRM, he will be free to distribute his software too, unhindered by the DMCA and similar laws.

In other words, GPLv3’s “anti-DRM” language appears to be included not to protect free software from DRM restrictions, but rather to make free software a less attractive choice for developing DRM technology.

The provision of GPLv3 section 3 that says DRM based on GPL-licensed software will not “count” under DMCA reads as follows:

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996 [such as DMCA], or similar laws prohibiting or restricting circumvention of such measures.

The GPL already requires that the covered work itself be accessible and modifiable and thus already prohibits locking the covered work with DRM. The effect of this provision thus appears to be on DRM that controls access to a copyrighted work other than the covered work. Such a work may be either non-GPL-licensed code, or content.

But when GPL-licensed code implements DRM for non-GPL-licensed code, or content, the owner of copyright in the DRM-protected code or content is not necessarily a GPL licensor or licensee. Therefore, it is not obvious how such a person can be bound by what the GPL “deems” to be, or not to be, the case.

For example, assume that a developer uses GPLv3-licensed code to write media player software with DRM for content played on it. A recipient of the media player software uses it to play songs. The songs are protected by copyright, and the copyright is owned by a third party. Now assume that a user of the software (who is a GPL licensee) modifies the media player code to circumvent the DRM.

GPLv3 probably leaves the original developer of the media player code without grounds to complain. The second sentence of section 3 states:

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.<sup>14</sup>

It is less clear, however, that section 3 affects the power under DMCA of the owner of copyright in the song, or of the government, to act against the circumvention. The song copyright owner did not necessarily convey a work covered by the GPL, so that person may have neither waived any rights nor disclaimed any intentions under the second sentence. Nor, as a third party, has the song copyright owner accepted the statement in the first sentence that the covered work will not be “deemed” part of an ETM. And if third parties or the government are not affected by section 3, anyone who attempts to circumvent the DRM in the work still faces a risk of liability.

It is possible that the song copyright owner would need to use GPL-licensed code to apply the DRM protection to the song. But using the code is not the same as conveying it, and the waiver clause, at least, requires conveying.<sup>15</sup> It is also possible that those whose works are protected by GPL-licensed DRM code could be presumed to know that the GPL permits the DRM code to be broken and that, as a result, the DRM is not an “effective” means of protection. But to know that the DRM is not effective because of what the GPL permits, they would have to know that the GPL actually is legally effective to permit the DRM code to be broken – and that is the question on the table.

It therefore appears that the anti-DRM provisions do not necessarily offer reliable protection from liability under DMCA for all those who circumvent DRM that has been implemented in GPL-licensed code. To the extent the provisions create doubt about the integrity of DRM implemented in GPL-licensed code, however, they may discourage the use of free software to implement DRM. As with the anti-tivoization provision, the anti-DRM section may have the practical effect of causing software other than GPL-licensed software to be used, without necessarily changing the underlying practice (“tivoization” or DRM) as FSF desires.

---

<sup>14</sup> Although the significance of disclaiming an “intention” is not clear, the waiver in the first part of the sentence does seem to apply to the developer.

<sup>15</sup> Section 2 provides: “The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work.” The encrypted song does not become a covered work merely because it is out put from a covered work.

### 3. Patents

In no area has GPLv3 broadened the GPL's scope more than patents. FSF's opposition to software patents has long been known, but GPLv3 adds several new patent-related terms. To see how the GPL's treatment of patents has evolved, it is useful to begin by reviewing how patents have been dealt with in previous versions.

#### a. GPLV1: Focus on Copyright

Viewed broadly, a license is a permission to engage in conduct that, without a license, the law prohibits. For example, James Bond requires a "license to kill" because, without one, killing generally runs afoul of the criminal law. In the context of private property rights, as Professor Eben Moglen of Columbia University has pointed out, a "license" is a permission to use someone else's property (<http://www.gnu.org/philosophy/enforcing-gpl.html>). Such a license is required because, without it, use of the property would infringe the other person's private rights.

In February 1989, when FSF released GPLv1, the main property right that prohibited the use of software without permission was copyright. The permission-granting provisions of GPLv1 therefore operated entirely in relation to acts that are the subject of the copyright owner's exclusive rights under 17 U.S.C. § 106 – namely, reproducing a copyrighted work in copies, distributing the copyrighted work to the public, and preparing derivative works based on the copyrighted work.

For example, section 1 of GPLv1 states that the licensee "may copy and distribute" verbatim copies of the source code (on specified conditions), and sections 2 and 3 state that the licensee "may modify" copies and "copy and distribute" such modifications, again on specified conditions. Section 4 states that the licensee "may not copy, modify, sublicense, distribute or transfer" the program except as expressly provided. In GPLv1, patents simply do not enter the picture.

#### b. GPLV2: An Ambivalent Foray Into Patents

Software patents were looming on the horizon in 1989, and by 1991 they could no longer be ignored. On February 28, 1991, the League for Programming Freedom, founded by Richard Stallman, issued a manifesto entitled "Against Software Patents" (<http://lpf.ai.mit.edu/Patents/against-software-patents.html>). The paper, which appealed both to free software developers and to proprietary software companies, began:

Software patents threaten to devastate America's computer industry. Patents granted in the past decade are now being used to attack companies such as the Lotus Development Corporation for selling programs that they have independently developed. Soon new companies will often be barred from the software arena – most major programs will require licenses for dozens of patents, and this will make them infeasible. This problem has only one solution: software patents must be eliminated.

GPLv2, released the same month as this manifesto, introduced the first GPL provisions relating to patents. GPLv2 did not contain an explicit patent license. This may have been because it was assumed that developers of free software were unlikely to go to apply for patents. It may have also reflected the ideological goal, which was not to license software patents but to make them go away.

Indeed, the patent provisions of GPLv2, which appear in its section 7, seem to reflect an ambivalence about addressing patents at all. The first sentence of section 7 states in general terms that the GPL's "conditions" are not "excuse[d]" by contradictory "conditions" that may be "imposed" on the licensee by a court, a patent infringement allegation, or "for any other reason." The next sentence states: "If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all." These sentences to some extent appear to state the obvious – namely, that the License does not excuse that which it does not excuse.

The next sentence of section 7 refers specifically to patent licenses:

For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

In this example, even though the patent license might permit the immediate GPL licensee to distribute the Program, that licensee could not do so, under the GPL, if the patent license did not grant the same permission to downstream GPL licensees. Indeed, even if it turned out that the patent license was unnecessary (because the patent was invalid or the Program did not infringe), acceptance of the patent license (which nevertheless might have been prudent as a precaution) would have terminated the immediate GPL licensee's GPL license. In this sense, the example arguably encourages GPL licensees to live dangerously as far as patents are concerned.

Perhaps for this reason, the third paragraph of section 7 of GPLv2 rather defensively states:

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices.

Whether or not section 7 of GPLv2 is an inducement to infringe, it is an inducement not to accept patent licenses that are narrower in scope than the GPL. As a result, section 7 of GPLv2 affords few options for licensees who want to comply both with the GPL and with obligations created by the patent rights of third parties. It also provides limited

options for patent owners who want to cooperate with free software users but are not prepared to grant universal royalty-free patent licenses.

c. Patents in GPLv3

GPLv3 expands the treatment of patents by adding a number of provisions, each of which seems aimed at a particular objective. These include, in section 10, a “patent retaliation” clause and, in section 11, an express patent license, a provision that restricts reliance on patent licenses, a purported “automatic extension” of certain patent licenses, and a provision regarding “discriminatory” patent licenses.

i. Patent Retaliation and Contributor License

Section 10 of GPLv3 states that, when “you convey a covered work,” the recipient of the work “automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License.” Section 10 also states that “[y]ou may not impose any further restrictions on the exercise of the rights granted or affirmed under this License.” It adds that, for example, “you may not initiate litigation . . . alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.”

FSF FAQ document calls section 10 a “patent retaliation clause” (<http://www.fsf.org/licensing/licenses/gpl-faq.html>). The FAQ goes on to say: “Section 10 prohibits people who convey the software from filing patent suits against other licensees.” The clause refers to a license only from the “original licensors” so that a conveying licensee who merely conveys the work without contributing to it is not said to be thereby granting a patent license to downstream recipients.

Section 11 grants an express patent license. It states, in part: “Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.”<sup>16</sup>

Although the section 11 license extends to patent claims “whether already acquired or hereafter acquired,” it is limited in three key respects. First, it is granted only by a “contributor.” A contributor is a copyright holder authorizing use under the GPL of the Program or a work on which it is based. If a licensee redistributes a work without modifying it, the licensee is not a “copyright holder” in the work, and the licensee’s authorization to use the work is neither required nor granted. As a result, a licensee who merely redistributes a verbatim copy grants no patent license under this clause.

---

<sup>16</sup> Section 11 defines “contributor” as “a copyright holder who authorizes use under this License of the Program or a work on which the Program is based,” and it states: “The work thus licensed is called the contributor’s ‘contributor version.’” The contributor’s “essential patent claims” are claims “owned or controlled by the contributor . . . that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version.”

Second, the section 11 patent applies only to “the contents of [the] contributor version.” If a GPL licensee creates a modified version of the work, the contributor is still permitted, as far as section 11 is concerned, to assert its patent claims against any contents of the modified version that are not contents of the contributor version. For example, if a downstream licensee improves a patented software process by adding new “contents,” section 11 permits the contributor to assert its patent claims in the process against the improvements.

Third, section 11 applies only to “essential patent claims.” If the contributor version is covered by claims 1 through 5 of a patent, and a GPL licensee adds contents that are covered by claims 6 through 10, the contributor can still assert claims 6 through 10 against the GPL licensee.

The interaction of sections 10 and 11 is not entirely clear when the “contributor” is not an “original licensor,” but a conveying licensee. Under section 11, the contributor has retained the right to assert its “essential patent claims” against new “contents” in downstream works and its non-essential patent claims against any contents of downstream works. Section 10 indicates, however, that the contributor who is also a conveying licensee may not assert any patent claims that would restrict the GPL “rights” of downstream recipients, which include permissions to run and convey the works that appear to be broader than the permissions granted by the section 11 patent license.

#### ii. Reliance on a Patent License

Section 11 of GPLv3 goes on to say:

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

Among other things, this clause appears to permit the licensee to convey a covered work that the licensee knows is infringing of a third-party patent as long as the licensee relinquishes any patent license that the GPL licensee has received. This provision is similar to the “example” in section 7 of GPLv2. Interestingly, though, GPLv3 no longer disclaims an intent to induce patent infringement.

iii. “Procuring Conveyance”

In 2006, Microsoft and Novell entered into an arrangement in which Microsoft did not grant a patent license directly to Novell, but agreed not to sue Novell’s Linux customers under certain Microsoft patents. They also agreed that Microsoft would distribute “support certificates” to customers that entitled them to support and updates for SUSE Linux, which Novell distributes.

The so-called “gotcha” clause of GPLv3 section 11 states:

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

On July 5, 2007, shortly after the release of GPLv3, Microsoft announced (<http://www.microsoft.com/presspass/misc/07-05statement.mspx>):

At this point in time, in order to avoid any doubt or legal debate on this issue, Microsoft has decided that the Novell support certificates that we distribute to customers will not entitle the recipient to receive from Novell, or any other party, any subscription for support and updates relating to any code licensed under GPLv3. We will closely study the situation and decide whether to expand the scope of the certificates in the future.

Even if Microsoft had not taken this action, however, it seems doubtful that the GPL could cause a patent license granted by a party in Microsoft’s position to third parties to be “automatically extended.”

For purposes of this clause, section 11 states:

[A] “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

Notwithstanding this definition of “patent license” as an “express agreement or commitment”<sup>17</sup> Professor Moglen has insisted that the GPL operates by granting

---

<sup>17</sup> GPLv3 does not define “commitment.” The most relevant sense given in Merriam-Webster’s online dictionary is “the obligation or pledge to carry out some action or policy” or “an engagement by contract or purchase order to assume a financial obligation (as to accept goods at an agreed price, to pay for subscribed

“unilateral permissions” (<http://www.gnu.org/press/mysql-affidavit.html>) and is *not* a contract (<http://www.gnu.org/philosophy/enforcing-gpl.html>). GPLv3 elsewhere seems to adopt this concept, stating in section 9:

You are not required to accept this License in order to receive or run a copy of the Program. . . . However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

FSF contends that a GPL *licensor* binds itself to the license or permissions granted in the GPL. Section 2 of GPLv3 states that the “rights” granted in the license are “irrevocable” if the stated conditions are met. Whether unilateral permissions that are not part of a contract can be irrevocable is open to question. Promises must be supported by consideration, or a consideration substitute, to be binding. The Nimmer treatise on copyright states: “The most well-known aspect of Anglo-American contract law is undoubtedly the doctrine of consideration. . . . [C]onsideration is necessary to render a nonexclusive license irrevocable.” 3-10 Nimmer on Copyright § 10.03[A][8]. The permissions in the GPL may well be revocable at will if they are, as FSF maintains, truly unilateral and not contractual.

But even if the rights granted by the licensor are somehow irrevocable, conditions of unilateral permissions by the licensor that are not part of a contract cannot impose contractual obligations on the *licensee*. And even if merely “procuring conveyance” requires copyright permission, a GPLv3 licensee who procures conveyance of a work, but has not modified or otherwise contributed to the work, is only a GPL licensee – not a GPL licensor. Permission under the license is subject to the licensee’s compliance with the conditions of the permissions. But the only possible consequence of not complying with the conditions is loss of the license permissions and potential liability to the licensor for copyright infringement.<sup>18</sup> Such liability of the GPL licensee to the GPL licensor cannot affect the patent liability of third parties to the GPL licensee.

GPLv3 could have been written to state that, if a GPL licensee does not extend a patent license it has granted to third parties to all recipients of the covered work, the GPL licensee loses the permissions granted to it in the GPL. But this is not what GPLv3 says. Rather, it states that the act of “propagat[ing] by procuring conveyance” has the effect of

---

stock, or to make a mortgage loan upon the completion of a building).” An equivalent might be “binding promise.” But a promise, like an agreement, requires consideration, or a consideration substitute, to bind.

<sup>18</sup> FSF seems to acknowledge this point in describing the effect of the patent retaliation clause in section 10. FSF states (<http://www.fsf.org/licensing/licenses/gpl-faq.html>) that, if someone who conveyed the software filed patent suits against other licensees despite section 10’s “prohibition,” “section 8 explains how they would lose their [GPL] license and any patent licenses that accompanied it.” Because the GPL is a grant of permissions and not a contract, one who does not comply with the GPL’s conditions loses the conditional permissions it grants, but has no contractual obligations and is not in breach of contract.

“automatically extend[ing]” a patent license that the GPL licensee has granted to third parties. The GPL, as a grant of unilateral permissions from the GPL licensor to the GPL licensee, cannot of its own force change the legal relations between the GPL licensee and third parties. Nor can the GPL, as something that is not a contract, impose a contractual obligation on the GPL licensee to change its legal relations with third parties.<sup>19</sup>

If merely “procuring conveyance” actually requires copyright permission, and if the clause in question were held to operate as a condition of the GPL permissions, the threat of a copyright infringement suit with statutory damages against a GPL licensee could be significant. Such a suit would have no legal effect, however, on the ability of the GPL licensee to enforce the GPL licensee’s patents against third parties. And it is not clear that a clause that purports to change the legal relations between the GPL licensee and third parties could be converted, by an adjudicating court, into a condition of the GPL license grant when the license text does not present it as one.

#### iv. “Discriminatory” Patent Licenses

As previously noted, section 7 of GPLv2 provides that a GPL licensee who receives a patent license that does not “permit royalty-free redistribution of the Program” by all downstream recipients may not, under the GPL, distribute the Program. The Microsoft-Novell arrangement avoided section 7 of GPLv2 by not granting a patent license to Novell, but instead agreeing with Novell that it would not enforce certain patent claims against Novell’s customers.

Seeking to close the “loophole,” GPLv3 provides as follows:

You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.<sup>20</sup>

This provision gives effect to the Microsoft-Novell arrangement as it existed before 28 March 2007, but would prohibit a company in Novell’s position from entering into a similar arrangement in the future. (It would not, however, prohibit a third party that is

---

<sup>19</sup> Raymond Nimmer concludes that, even if the GPL is a contract, the clause may not be effective (<http://www.ipinfoblog.com/archives/licensing-law-issues-gpl-30-and-third-party-patent-owners.html>).

<sup>20</sup> “A patent license is ‘discriminatory’ if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License.”

not “in the business of distributing software” from doing so – even though a software distributor seems more likely to have legitimate business reasons for entering into such an arrangement than some other third party.) The “grandfather clause” is presumably designed to give effect to the “gotcha clause” against Microsoft, but as outlined above, the enforceability of the gotcha clause appears doubtful.

#### D. Still No IP Warranty

After the “Terms and Conditions” section, in “How to Apply These Terms to Your New Programs,” GPLv3 states:

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<http://www.gnu.org/licenses/>>.

The use of “should” and the placement of this statement in the how-to appendix indicates that it is not mandatory. In addition, the title indicates that it applies only to “new” Programs and not, for some reason, to modifications of existing works. And in any event, GPLv3, like its predecessors, contains no warranty that the licensor has the right to grant the license or that the use of the work will not infringe third-party rights.

Rather, like prior versions, GPLv3 disclaims all warranties. Section 15 states that, to the extent permitted by applicable law, “there is no warranty for the program.” Section 16 states that, unless required by applicable law or agreed to in writing, no “copyright holder” or “any other party who modifies and/or conveys the Program as permitted above” will be liable to the licensee “for damages, including any general, special, incidental or consequential damages arising out of the use [of] or inability to use the Program . . . .”

The drafters may have wanted to avoid requiring a GPL licensor to indemnify downstream licensees against third-party patent infringement claims, but the effect of these provisions is that there is not even a warranty that the GPL licensor is the true author or copyright owner of the work – and if such a warranty can be said to be present by operation of law, liability for its breach is still disclaimed.

### CONCLUSION

FSF’s comments indicate that GPLv3 pursues two broad sets of objectives. One is to combat “recent threats” to free software, which includes closing perceived loopholes in GPLv2. The other is “to make the license easier for everyone to use and understand.” As discussed above, these two sets of objectives are in tension. The ambitious substantive objectives have produced a new version that is almost twice the length of GPLv2 and almost three times the length of GPLv1, and the GPL as it stands is likely to be understood only by readers with post-graduate educations.

The complexity of GPLv3, and at least one apparent technical drafting error of some significance, are not merely aesthetic problems. Because much of GPLv3 is new or reworded, interpretations over the past 16 years under GPLv2 will provide limited guidance at best. And because GPLv3 is so abstruse, any serious attempt to understand its terms and conditions is likely to require the assistance of legal counsel. This may be good news for technology lawyers, but it means that use of GPLv3 will impose significant new transaction costs on both licensors and licensees.

Whether or not one shares FSF's objectives, it is not at all clear that the costs that will be imposed by GPLv3 are offset by benefits of the new provisions. The attempts to address "recent threats" are some of the most problematic parts. The "anti-tivoization" provisions are legally coherent, but their primary practical effect may well be to discourage manufacturers of consumer devices from using GPL-licensed software, rather than enabling users to modify the software that operates such devices. The anti-DRM provisions, to the extent they purport to bind third parties, are of doubtful legal effect – and to the extent they do have legal effect, their primary practical effect may be to discourage developers from using GPL-licensed software to implement DRM, rather than limiting the use of DRM itself.

The most complex provisions of GPLv3, in the patent realm, are also in many ways the most difficult to interpret and apply. The patent retaliation clause of section 10 and the contributor license grant of section 11 reflect complicated compromises and do not neatly dovetail. They appear to allow for cases in which no explicit patent license is granted, but a contributor who is also a conveying licensee could lose its own GPL license under section 10 for bringing a patent suit that, as far as section 11 is concerned, the contributor is free to bring.

The clause that purports "automatically" to extend any patent license that a "procurer" of conveyance may grant seems unlikely to be given legal effect, because merely procuring conveyance, even it requires copyright permission, at most makes the procurer a GPL licensee, not a GPL licensor. Although a GPL licensor grants licenses, a GPL licensee who does not modify a covered work or distribute a modified work has no copyrightable contribution in which to grant a license and therefore is not a GPL licensor. The permissions granted to the licensee may be subject to conditions, but if the licensee does not fulfill the conditions, the only consequence under FSF's "unilateral permission" approach is that the licensee loses its license and is vulnerable to a copyright infringement suit, if in fact its actions required copyright permission. The ability of the GPL licensee to enforce its own patent rights against third parties is unaffected.

For all these reasons, the success of GPLv3 in the increasingly competitive marketplace of free and open source software licenses seems far from assured.