

# A Practical Guide to the GNU GPL

Revision 2.00

Sean C. Crandall  
Registered Patent Attorney  
Jackson Walker LLP  
112 E. Pecan Ste. 2400  
San Antonio, TX 78205  
e-mail: [scrandall@jw.com](mailto:scrandall@jw.com)

©2008 – 2010, Sean C. Crandall. All rights reserved.

I welcome your comments or feedback on this paper. E-mail is the fastest way to get in touch with me, though regular mail will do in a pinch.

Available online at <http://www.jw.com/site/jsp/attyinfo.jsp?id=494>.

THIS DOCUMENT IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. THIS DOCUMENT **does not constitute legal advice and should not be relied on as such**, EVEN IF YOU RECEIVED IT DIRECTLY FROM THE AUTHOR. THIS DOCUMENT IS SOLELY THE WORK OF THE AUTHOR, AND DOES NOT REFLECT THE OPINION OR POSITION OF JACKSON WALKER LLP OR ANY OF ITS PARTNERS OR AGENTS. THE AUTHOR MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AS TO THE ACCURACY OF THE INFORMATION OR ANALYSIS SUPPLIED HEREIN.

THIS DOCUMENT MAY CONTAIN REFERENCES OR LINKS TO EXTERNAL PRODUCTS, COMPANIES, AND/OR WEBSITES. THESE ARE SHOWN **by way of example only**. LINKED WEBSITES ARE THE WORK AND RESPONSIBILITY OF THEIR RESPECTIVE OWNERS. THE AUTHOR MAKES NO WARRANTY AS TO THE CONTINUED ACCURACY OF THE LINKS PROVIDED, AND **does not endorse, condemn, or otherwise express any opinion whatsoever** REGARDING THESE PRODUCTS, COMPANIES, AND/OR WEBSITES AND THEIR CONTENT.

Built with L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> (<http://miktex.org>)  
Edited with Vim (<http://www.vim.org>)

## Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>The Preamble</b>	<b>4</b>
2.1	Users . . . . .	5
2.2	Developers . . . . .	5
2.3	The Religion of Free Software . . . . .	5
<b>3</b>	<b>Section 0—Scope of the GPL</b>	<b>6</b>
3.1	What is Covered . . . . .	7
3.2	What is not Covered . . . . .	7
<b>4</b>	<b>Section 1—Verbatim Copies</b>	<b>8</b>
4.1	Copying and Distribution . . . . .	8
4.2	What is Covered . . . . .	9
<b>5</b>	<b>Section 2—Modifying the Program</b>	<b>10</b>
5.1	Notification Provisions . . . . .	10
5.2	Licensing . . . . .	11
5.3	Defining Derivative Works . . . . .	11
<b>6</b>	<b>Section 3—Binary Distribution</b>	<b>14</b>
6.1	Distribution . . . . .	14
6.2	Downloads . . . . .	17
<b>7</b>	<b>Section 4—Limitations</b>	<b>17</b>
7.1	Automatic Termination . . . . .	17
7.2	Use After Termination . . . . .	18
<b>8</b>	<b>Section 5—Acceptance</b>	<b>19</b>
<b>9</b>	<b>Section 6—Sublicensing</b>	<b>20</b>
<b>10</b>	<b>Section 7—Patents</b>	<b>20</b>
<b>11</b>	<b>Section 8—Territorial Restriction</b>	<b>21</b>
<b>12</b>	<b>Section 9—Revisions to the License</b>	<b>22</b>
<b>13</b>	<b>Section 10—Multiple Licensing</b>	<b>23</b>
13.1	Derivative Works . . . . .	23
13.2	Dual Licensing Your Work . . . . .	24
<b>14</b>	<b>Section 11—Warranties</b>	<b>27</b>
<b>15</b>	<b>Section 12—Limitation on Remedies</b>	<b>27</b>

<b>16 Conclusion</b>	<b>27</b>
<b>A The GNU GPL</b>	<b>29</b>

## 1 Introduction

This short paper is a practical guide for those who are thinking of entering the sometimes-daunting world of the GNU<sup>1</sup> General Public License (GPL). The GPL is a license published by the Free Software Foundation (FSF), a non-profit 501(c)(3) organization whose stated mission is to “promote computer user freedom and defend the rights of all free software users.”<sup>2</sup>

There are several reasons you might be thinking of using software licensed under the GPL. The first and simplest situation is that you want to use some GPL software for your own purposes. The second is that you may want to sell or otherwise distribute GPL software. The most complicated issues arise when you want to create derivative works from others’ GPL software.

This paper addresses each of those scenarios with a focus on version 2 of the GPL, which is presently the most widely used. When I refer to “the GPL,” I am referring to version 2. After reading this paper, you should have a feel for what you can and can’t do with software you have received under GPL version 2. If, after reading this paper, you feel that the GPL is right for you, you should consult with an attorney who can help you determine exactly how the GPL will affect your individual project.

## 2 The Preamble

The preamble is a summary of the purpose and content of the GPL, and like most preambles, contains no substantive terms. It starts out with a philosophical exposition on the purposes of the GPL. The opening lines, which claim that the GPL is somehow fundamentally different from other software licenses, are a little disingenuous. Like almost every other software license, the GPL conveys rights to the licensee that are less than the total interest in the licensed software, and places restrictions on your use and distribution.

This is not to say that the GPL is “bad” or “evil.” But it is important to recognize that the word “free” does not mean you are free of obligations. In fact, in many respects, the GPL is a very aggressive and far-reaching license. It has been called a “viral” license (often by its detractors) because it attaches itself to almost everything it touches.<sup>3</sup> So if you’re looking at using free software, you need to think seriously about what you want to do with it.

---

<sup>1</sup>“GNU” is a recursive acronym, which is a type of acronym popular among computer programmers. GNU is short for “GNU’s Not Unix” and refers to a Unix-like operating system that was the original focus of the Free Software Foundation.

<sup>2</sup>See <http://www.fsf.org/about/>

<sup>3</sup>This is sometimes used as a philosophical argument against using the GPL, but the argument hardly passes the giggle test. If you were to incorporate some proprietary source code into your project without a proper license, it would similarly infect all derivative works to the full extent allowed by copyright law. The GPL can do no more.

## 2.1 Users

If you're just planning to install the software for its intended use, there are far fewer pitfalls. You can legally download the software from wherever it's available, install it, and start working away. This is true even if you are a large enterprise operating for profit. This is true even if you are using the software to produce output that is a commercial product (for example, if you are using a GPL music mixer to create electronic music, you still retain copyright to the music you create). You can even modify and internally distribute<sup>4</sup> the software, all without having to worry about making your work free to the world.

## 2.2 Developers

If you are a developer, and you intend to distribute your product, the GPL can be either your best friend or your worst enemy. Are you committed to distributing free software? Are you prepared to hand over your source code to all who ask, free of charge? Are you comfortable with the fact that your competitors may be able to use your own code in a competing product (though it would also be bound by the GPL)? Do you want to let your end users contribute changes to the software? Do you believe the maxim that “many eyes make all bugs shallow”? Do you want to ensure that anything derived from your software will also be bound by the GPL? If you can safely answer “yes” to those questions—and many people can—the GPL is a great choice. If you can't, choose a different license for your own software, and give the GPL a wide berth if you are looking to re-use existing code.

## 2.3 The Religion of Free Software

Paragraph 3 of the Preamble deals with a common misunderstanding of the word “free.” This word (along with its cousin “open”) has been the subject of vitriol and flame wars on many a Usenet group. The catch is that “free” has nothing to do with the sticker price of a piece of software. It is an ideology, and one whose adherents are often characterized by religious zeal. They are not (at least in theory) bothered by you charging for your software. You can charge any price you want. But they are zealous about delivering software with the “freedom” intact.



### Hypothetical 2.1

Martha writes a utility called “CONXMP” that encodes, decodes and converts the hypothetical XMPL music format. She licenses it under the GPL. Jack wants to make a music player that works with many different media formats, including XMPL files. Jack downloads a copy of CONXMP and incorporate the routines that perform

---

<sup>4</sup>Note that any internal distribution would need to be with protections treating the modified code as a trade secret so that it is not treated as a “distribution” under copyright law. The law of distribution is beyond the scope of this paper. If in doubt, consult your attorney.

the encoding and decoding into his new music player “OMNIBUS.” OMNIBUS includes 99 other music codecs, and Jack wrote all of those himself.

Under copyright law, OMNIBUS is a derivative work of CONXMP, even though XMPL is only one of 100 codecs. CONXMP may comprise only a tiny fraction of Jack’s source code, but by using it, Jack has committed his entire project to the GPL. So can Jack charge money for OMNIBUS? Absolutely. Perhaps he wants to distribute the executable for \$29.95. That’s fine. But he also has to make all of his source code available, at no cost,<sup>5</sup> to anybody who receives a copy of OMNIBUS.

If Jack’s competitor Rose wants to develop her own media player called UNIVERSE, she can purchase a copy of OMNIBUS, and Jack will be obligated to give her his source code. Rose can then use that to create her UNIVERSE player, which will also be bound by the GPL. What’s more, Rose may not even need to pay Jack \$29.95 for the executable. For example, if Doc has already received a copy of the source code from Jack, Doc is perfectly free to give Rose a copy of the source code without Jack’s permission.

Does this seem a little unfair to Jack? He used just one GPL codec, and now the GPL gets to take over his entire project, including 99 codecs that he wrote himself? It’s really not unfair when you think it all the way through. Remember that Martha was never obligated to give Jack her codec, or to allow him to make derivative works. If Jack really wanted to keep his software proprietary, he could write his own XMPL codec, or just sell a music player without XMPL support. Jack chose to reap the fruit of Martha’s copyrighted labor, and the law gives Martha the right to impose conditions. If Jack doesn’t agree with those conditions, he shouldn’t use Martha’s work. But if he can live with those conditions, using CONXMP is a great way to save time and effort. Jack just has to be sure to decide *before* he lets CONXMP anywhere near his OMNIBUS code base, for to violate the GPL is to quickly incite the wrath of the Church of Free Software.<sup>6</sup>

### 3 Section 0—Scope of the GPL

If you’re not a programmer, you may scratch your head at the designation of section “0.” This is just an inside joke with programmers,<sup>7</sup> and may be an

<sup>5</sup>The GPL permits the distributor to charge its actual costs, but for practical purposes, the costs of reproducing a digital file are negligible and can be ignored.

<sup>6</sup>For example, see <http://blogs.zdnet.com/microsoft/?p=4547>. Free Software proponents were up in arms when GPL code was found in one of Microsoft’s utilities for Windows 7. Microsoft discovered that one of its vendors had used the GPL code without its knowledge. Microsoft quickly pulled the utility, and a few days later re-released it under the GPL.

<sup>7</sup>Most programming languages address the first value in an array with the index 0. This makes sense mathematically because the name of the array represents a location of sequential values in memory. The first value is at offset 0 from that location, while the second is at offset 1, and so forth. If this doesn’t make any sense to you, don’t worry. The joke isn’t that funny anyway

homage to Brian Kernighan and Dennis Ritchie’s *The C Programming Language*, which started with Chapter 0.

### 3.1 What is Covered

Section 0 is concerned with the scope of the GPL. It defines the word “Program,” which is used throughout the rest of the license to mean any program or work that has a GPL notice. It also addresses the term “work based on the Program,” which is expressly intended to include any derivative work as defined under copyright law. If you copy, modify, or distribute a GPL Program, or a work based on a GPL Program, you need to be aware of the terms of this license.

### 3.2 What is not Covered

Paragraph two of this section expressly limits the scope of the license to copying, modifying and distributing the Program. If you are not doing one of those things, then you won’t need to fret about the GPL. Honestly, this section could have been crafted more carefully. It states that acts other than copying, distributing, and modifying are outside of the scope of the license. Construed literally, that could mean that you have no right to actually run the program (since this is the only license you received). But the very next sentence states that “[t]he act of running the Program is not restricted,” which by granting a license, brings running the program into scope.<sup>8</sup>

So simply running the program does not “infect” your output in most cases. For example, I am using a GPL text processing program called L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> to process text I entered into a text editor called Vim that is distributed under a GPL-like license. Both are running on the GPL Linux 2.6 kernel<sup>9</sup>. None of this affects my copyright in this document or requires me to distribute the document under the GPL. If I am inclined, I can distribute it for profit with as restrictive a license as I please. Similarly, I may create or modify pictures in the GPL GIMP software, and my copyright is not affected. I can even write commercial software and compile it with the GPL gcc compiler, all without the GPL affecting my copyright.



#### Hypothetical 3.1

Doc downloads a copy of the OMNIBUS media player, licensed under the GPL, from Jack’s XMP Online Music (XOM). Jack provides a streaming media player powered by OMNIBUS and uses XOM to sell digital downloads in the XMPL format. Doc improves the OMNIBUS server functions so that it provides a better listening experience for end users, and calls his improved media server MULTIVERSE. Using MULTIVERSE as a backend, Doc opens a

<sup>8</sup>For a more detailed discussion of the interesting—but ultimately irrelevant—issues that arise from GPL’s generosity, see 8 below.

<sup>9</sup>If you’re wondering, my window manager is Fluxbox, which is distributed under the more liberal MIT license. So there are no GPL issues there.

competing online music service called BLUEBOX, which becomes wildly popular and soon eclipses Jack's XOM. Because MULTIVERSE gives Doc a competitive advantage, he wants to keep it proprietary. He particularly wants to ensure that Jack can't use the MULTIVERSE source code to improve XOM.

In this case, Doc has not distributed any copies of MULTIVERSE to anybody; he simply deployed it on his own servers. The fact that third parties can *access* the running program and receive output from it does not a distribution make. And the “viral” provisions of the GPL only enter in when a derivative software product is *distributed*. So Doc is free to keep MULTIVERSE proprietary. In short, Doc's only repercussion from appropriating Jack's source code and using it to undermine Jack's business is bad karma. But Doc needs to be careful. As soon as he distributes a copy of MULTIVERSE to any third party, he has opened the flood gates. He must then license MULTIVERSE to the third party under the GPL, and that third party is free to send a copy to Jack or anybody else.

The situation above will become increasingly important as “Software-as-a-Service” (SaaS) gains popularity. In instances where commercial software vendors want to provide applications for online use (often for a recurring subscription fee), there is nothing in the GPL that prevents them from using GPL software in their products and keeping their improvements to themselves. For example, Google provides its online office suite “Google Docs” without, to my knowledge, distributing copies of the software. Google Docs may be brimming with GPL Programs from top to bottom, and Google may be keeping their improvements to those programs as proprietary trade secrets. This is not a violation of the GPL, and still would not be a violation if Google started charging a monthly subscription fee for Google Docs.

If you are writing open-source server-side software and are concerned about others misappropriating your work, you may want to look into the Affero GPL.<sup>10</sup>

## 4 Section 1—Verbatim Copies

### 4.1 Copying and Distribution

This section deals with your right to copy and distribute the GPL software verbatim. You are free to make and distribute verbatim copies of the Program with the condition that each copy must include a conspicuous copyright notice and disclaimer of warranty. You must not tamper with the GPL notices existent on the software, and you must provide each recipient a copy of the GPL along with the Program.

You are even permitted to sell the software for a fee (often with improved packaging or an accompanying service). This is how many companies have successfully commercialized the GPL. They offer services such as support and

<sup>10</sup> <http://www.fsf.org/licensing/licenses/agpl.html>.



installation, so that the revenue stream is centered on the services rather than the software itself. Some companies also provide warranties. Others go so far as to provide indemnity in case any of the software is found to violate patents or other intellectual property.



### Hypothetical 4.1

Rose downloads a free copy of the OMNIBUS media player, which Jack licenses under the GPL. Noticing that the downloadable file is very large, Rose thinks that it will be hard for users with slow internet connections to download OMNIBUS. Rose copies the files to a master CD, along with a file called “LICENSE” that contains a copy of the GPL and provides instructions for ordering a separate CD with the source code for a nominal \$1 fee. Rose presses 1,000 copies of her disk, prints attractive labels, and packages them in a professionally-designed box. She sells her boxed copies of OMNIBUS for \$79.95.

Rose has done nothing that violates the GPL. Users who are willing to buy her boxed CDs can do so. But remember that there will be strong downward pressure on Rose’s price, because Amelia could, for example, create copies of Rose’s CD<sup>11</sup> and sell them in plain sleeves for \$10. Or, feeling generous, Doc could distribute free CDs for anybody with a slow internet connection.

## 4.2 What is Covered

If you are going to distribute GPL software, you will need to understand what, exactly, the author has licensed under the GPL. You may download a single binary file, thinking it’s just a “GPL file,” but the issue could be much more complicated. Very often, that binary file is either an installer or a zipped archive. This will expand into many different files, and you may receive those under different licenses. It is important to be aware of exactly what you have the right to distribute before you start uploading files to your server.



### Hypothetical 4.2

Doc downloads Jack’s OMNIBUS media player, which is licensed under the GPL. OMNIBUS comes as a zipped archive (with an extension like .zip, .tgz, or .tar.gz). The archive contains the following components:

- A binary executable, including a statically-linked XMPL codec
- Codec libraries for the other 99 codecs
- A source tree for the binary executable
- A source tree for the codec libraries

<sup>11</sup>But see the example in the next section for a caution about wholesale copying.

- Skins
- Sample music files
- A file called “LICENSES”

Upon reading the LICENSES file, Doc finds that the following licenses apply:

- GPL—The binary executable and source tree for the binary executable
- Lesser GPL (LGPL)—Codec libraries and source tree for the codec libraries
- Non-exclusive, personal, non-transferable license—Skins and sample music files

If Doc distributed the zipped archive as-is, he would be in violation of the license on the skins and sample music files. Jack has reserved the exclusive right to distribute those himself. Doc can still distribute the binary executable and codec libraries, but he will need to do it separately.

Furthermore, Jack may retain certain trademarks. For example, he may retain the sole right to distribute OMNIBUS-branded media players. In the example above, it may be that without the skins, the media player provides only a very plain interface with the simple title “Media Player.” Because Jack has not granted Doc a license to distribute the skins or use the trademark OMNIBUS, Doc will need to find a new look for his media player. He may decide to make his own set of skins, and with them re-brand the binary as the MULTIVERSE media player.

## 5 Section 2—Modifying the Program

You can modify the Program to your heart’s content. And if you keep those modifications to yourself, or use them only internally, you do not have to give your work to the world. But as soon as you distribute a copy, serious obligations kick in.

### 5.1 Notification Provisions

If you have changed a file, you must provide a prominent notice in that file indicating that it has changed, and the date of any changes. This encourages downstream recipients of the software to go to you with bug reports before complaining to the original author. This also preserves one of the incentives of free software culture. Many programmers who work on free software do so not for financial rewards (or at least not solely for financial rewards), but for the recognition that it brings. This provision ensures that downstream hackers<sup>12</sup>

<sup>12</sup>The term “hacker” as used here is different than its occasional use in popular culture. To the general public, the term may imply somebody who illegally accesses others’ computers.

know who contributed what and who is responsible for what. This not only gives later contributors credit, but also insulates the original author’s reputation from problems stemming from downstream changes.

Another important notice provision is found in subparagraph (c). This paragraph is largely appropriate to a program that operates on a command line interface (CLI), in which the user interacts through text commands. Such programs are still common in Unix-like systems, and were ubiquitous in 1991 when GPL v.2 was released. If you are using GPL source code to create a CLI program, you must provide an introductory message when the user first opens the program. This message should provide a copyright notice, as well as disclaim warranties (or alternately, provide your own warranty message if you are selling warranty services). It must also make users aware that the software is licensed under the GPL, and tell them how to view a copy of it. An exception is provided if the original Program was also interactive but did not provide a similar introductory message. In modern graphical programs, a similar notice is often provided in a popup window. In many cases, this can be accessed by using a menu feature like Help→About, Help→License or some similar mechanism.

## 5.2 Licensing

This provision is the heart of the GPL. Yes, you can copy and distribute the software. Yes, you can modify it as much or as little as you please. Yes, you can even distribute those modifications with or without a fee. But if you do, you must license those modifications under the GPL. What’s more, you cannot charge a fee for the license—you must give it freely, just as you received it freely. Note that this is distinct from the provisions that permit you to charge for distributing the software. The license specifically permits you to charge “for the physical act of transferring a copy.” This careful distinction preserves your right to make money if you can find willing buyers. But you cannot add an additional charge for the license itself.<sup>13</sup> And once you have licensed the code under the GPL, anybody else can give away your program for free, even if you don’t. This is why many companies do not rely (at least not solely) on a business model of selling GPL software. The competition is too open. Others can take your work and make derivative products and give them away.

## 5.3 Defining Derivative Works

The second paragraph of subpart (c) excludes from the GPL “identifiable sections of [the] work [that] are not derived from the Program.” These may be treated as independent works by themselves, but only if they are distributed separately. If they are distributed as part of a whole, including the GPL Program from which the new work is derived, they must be licensed under the GPL.

---

Among programmers, those people are sometimes called “crackers,” because they crack security mechanisms. “Hacker” is a more general term referring to those who write and modify code, and may even be used as a mild accolade.

<sup>13</sup>This is no mere distinction without a difference. See 10 below.

There is a recognition, however, that a “mere aggregation” of a separate work with the Program does not necessarily bring the work under this provision.

The definition of what constitutes a derivative work is one of the most controversial and hotly-debated issues surrounding the GPL. Much of the controversy revolves around whether dynamic linking creates a derivative work.<sup>14</sup> Since this can be confusing, I will present several examples below to help clarify the issues.



### Hypothetical 5.1

#### Fully Derivative Works

Martha distributes the standalone CONXMP utility under the GPL. Martha also operates BLINK, an online music store that sells XMPL-encoded songs. As a crack programmer, Doc analyzes the source code and realizes that the software could be 60% more efficient with some clever changes. Doc makes the changes and creates a new utility called DocXMPL. Doc uses DocXMPL to create a library of XMPL-encoded music that he encumbers with a Digital Rights Management (DRM) scheme and sells through his own online music store. DocXMPL gives Doc a competitive advantage, allowing him to quickly create and maintain a larger music catalog than Martha. Doc maintains DocXMPL as a trade secret.

DocXMPL is a statically-linked derivative work of CONXMP. Doc can use DocXMPL for his own purposes, or distribute it within his company for internal use without having to give his changes to Martha. This is true even though he uses DocXMPL in competition with Martha and even though it gives him a competitive advantage over Martha. But if Doc distributes even a single copy of DocXMPL to Rose, he must also make the source code available to her, including his improvements. Rose is then free to give those improvements to Martha if she wants.



### Hypothetical 5.2

#### Statically-Linked Works

Jack creates the OMNIBUS media player, including Martha’s XMPL codec, and 99 other codecs he wrote himself. Jack created the XMPL codec by taking a copy of the Martha’s CONXMP utility, which she licenses under the GPL, and delivering it to Mickey, a contract programmer. Mickey extracted the important subroutines that perform the encoding and decoding and added some code to enable them to interface with OMNIBUS. He then provided the modified source files to Jack, who never looked at them. Jack statically compiles all the codecs into a single large executable, along with the rest of the program.

---

<sup>14</sup>The issue of dynamic linking versus static linking will look very arcane to non-programmers. Attempting to explain the difference here would be tedious and well beyond the scope of this paper.

In this case, the entire executable is considered a derivative work. This is true even if the XMPL codec comprises less than 1% of the full source tree, and even though Jack never so much as looked at CONXMP himself. Jack may distribute the OMNIBUS media player as long as he complies with all the terms of the GPL.

But OMNIBUS is not automatically and irrevocably committed to the GPL. If Jack decides, before distributing OMNIBUS, that he wants to drop XMPL support, he can remove the XMPL codec and distribute OMNIBUS as a proprietary commercial product. Since the rest of Jack's work is identifiably separate from the XMPL codec, there are no GPL issues. Even if, unlike the hypothetical above, Jack performed the work on the XMPL codec himself instead of hiring Mickey, the rest of his work would not be tainted as long as it was not actually derivative of CONXMP.



### Hypothetical 5.3

#### Aggregate Works

Jack receives an XMPL codec from Martha under the GPL. Jack creates the OMNIBUS media player and distributes it under the GPL with the XMPL codec statically linked. Jack's OMNIBUS player is able to dynamically change its look and feel based on user-definable "skins." Jack creates a number of skins, which turn out to be enormously popular, and which give him a competitive advantage over Martha. The skins are loaded by the software at runtime and have no executable code. Jack charges \$29.95 for a CD containing the binary executable and the skins. He makes the source code for the OMNIBUS player available for download over FTP, but license the skins on a restrictive per-machine basis.

Although Jack's OMNIBUS player is necessarily licensed under the GPL, his skins are unaffected. The skins are a separately-identifiable work, and Jack's restrictive licensing of them is proper. The fact that they are distributed on the same CD as the GPL binary does not make them part of the GPL work. In fact, many companies that deal in GPL software provide these kinds of for-pay extras that create a real or perceived added value.

Hypothetical situation:



### Hypothetical 5.4

#### Dynamically-Linked Works

Jack creates the OMNIBUS media player as a modular system, with a main executable that dynamically links to various codecs through a defined interface. Jack provides 99 codecs that he wrote himself.

Mickey downloads a copy of Martha's XMPL codec, which she distributes under the GPL. Mickey modifies the codec to work as an OMNIBUS library and names his new library XMPIBUS. Mickey

offers XMPIBUS as a separate download. Jack likes XMPIBUS, downloads a copy, and starts distributing it with OMNIBUS.

This is the controversial situation that has drawn much commentary from GPL proponents. The question involves not only whether the GPL *purports* to taint OMNIBUS in this situation (or conversely, whether OMNIBUS taints XMPIBUS), but if it *can*, even if that is the intent of the GPL. It is clear that XMPIBUS is a derivative work of XMPL, and therefore must be licensed under the GPL. But does the dynamic linking of XMPIBUS to OMNIBUS make OMNIBUS a derivative work too? The answer is we just don't know. I am not aware of any U.S. case that has ruled on the issue.

This question has famously manifested itself in the matter of certain Linux hardware drivers that manufacturers provide as binary-only, non-GPL kernel modules. GPL purists claim that linking these modules taints the Linux kernel, and insist that the manufacturers instead release key specifications so that volunteers can develop GPL drivers. Hardware manufacturers counter that releasing their key specifications would disclose valuable trade secrets. Until a court has resolved this issue definitively, there will be serious risks involved with dynamically linking proprietary software with GPL software.<sup>15</sup>

A related license, the Lesser GPL (LGPL) avoids this issue by expressly permitting dynamic linking to libraries. This is, in my opinion, the saner approach because it doesn't require new case law to know what you are allowed to do.<sup>16</sup> Until we have that case law, the only completely safe course is to not link proprietary software to GPL software.

## 6 Section 3—Binary Distribution

Section 3 permits distribution of the Program or a derivative work in binary form. Binary distribution is often a logical choice for several reasons, including the fact that most users do not want or need the source code. They want to *use* the program for its intended purpose. This also eases distribution and installation, as compiling software can take a long time and be frustrating to users. When software is provided as a binary, it can be seamlessly integrated into a packaging system, or deployed by an executable installer program.

### 6.1 Distribution

When GPL software is distributed in binary form, additional requirements come into play that ensure that end users have access to the source code if they want

---

<sup>15</sup>On the other hand, if you're well-funded, like risk, and are anxious to create new case law, give me a call.

<sup>16</sup>The LGPL includes some language specific to libraries, which may be confusing for standalone applications. Although this language doesn't make it impossible to license standalone applications under the LGPL, you may want to consider other options. For example, the Mozilla Public License (MPL) or Common Development and Distribution License (CDDL) are also "copyleft" licenses that permit dynamic linking. Note that both of these are considered GPL incompatible by the FSF.

it. The license provides three possible means for distributing source code:

1. The source code may accompany the binary.



#### **Hypothetical 6.1**

Jack distributes OMNIBUS on a CD. To comply with the GPL, Jack includes all of his source code on the CD.

2. The software may be accompanied by a written offer to provide the source code. The offer must be valid for at least three years after delivering the binary, and you must provide a machine-readable copy on a medium customarily used for software interchange. This prevents shenanigans like providing the source code as a useless hard copy printout, or by delivering it on useless 8-inch floppies.



#### **Hypothetical 6.2**

Jack sells OMNIBUS on CDs for \$29.95, but includes only the executable code. Jack can fulfill his GPL obligations by providing a publicly-accessible FTP site for three years, or by giving his customers instructions for ordering a source CD for at most a nominal charge for reproduction.

3. The third option is only available for non-commercial distribution and only if you also received the program in binary form. In that case, you may simply provide the recipient with the instructions you received for obtaining source code.



#### **Hypothetical 6.3**

Donna downloads a copy of OMNIBUS from Jack's website. Since she is not a programmer, and is only interested in listening to music, she only downloads the executable installer. The installer includes a copy of a README file, with instructions for downloading source code from Jack's website. Donna likes OMNIBUS, and wants to give a copy to her friend River. Donna fulfills her obligations under the GPL as long as she delivers the installer intact, since it includes Jack's original instructions for downloading the source code.



#### **Hypothetical 6.4**

Rose downloads a copy of the OMNIBUS installer from Jack's website. She creates professional CDs with packaging and labeling to sell for \$79.95 each. Since Rose's use is commercial, she cannot just distribute the installer with Jack's instructions for downloading the source code. She will need to choose another option to comply with this section.

In most cases, the term “source code” will be self-explanatory. For example, if a program is written in C, appropriate C source files and header files should be provided. But paragraph 5 of this section requires that certain ancillary files, including interface definitions and scripts for compiling and installing the program, be provided too. Standard operating system components, such as a kernel and compiler, need not be provided. Although libraries on which the software is dependent are not expressly addressed, it is common practice to exclude those as long as they are readily available under a GPL-compatible license.

Libraries may be a problem if the program must link to a proprietary library to function properly. For example:



### Hypothetical 6.5

Jack creates the OMNIBUS media player as a derivative work of Martha’s CONXMP utility. Jack also maintains a proprietary library called JACKMEDIA that provides Jack’s own proprietary media format. JACKMEDIA is a completely separate project from OMNIBUS, but Jack decides to add calls to JACKMEDIA in his OMNIBUS code, and then link them at compile time. OMNIBUS cannot successfully link without JACKMEDIA.

In this case, Jack will need to provide JACKMEDIA to end users under the GPL. This would probably be true even if the facts were varied slightly, and OMNIBUS could successfully compile and link without JACKMEDIA (perhaps resulting in linker warnings), but would still be missing significant functionality.

Consider, however, an alternative situation, where OMNIBUS *dynamically* links to JACKMEDIA, but is substantially crippled without the functionality provided by JACKMEDIA. This alternative recalls the controversy over dynamic linking discussed in section 5.3 of this paper. In this case, because OMNIBUS cannot function properly without JACKMEDIA, JACKMEDIA looks more like an integral part of OMNIBUS, which is itself a derivative of a GPL Program. Even if JACKMEDIA is completely separate and usable as-is with other programs, it would probably have to be licensed under the GPL if it were included with OMNIBUS. This scenario presents a reasonable case for calling dynamic linking a derivative work and insisting that JACKMEDIA be provided under the GPL.

The question becomes more difficult when JACKMEDIA starts to look less like a critical component of OMNIBUS. For example, if Jack provides JACKMEDIA as only one of many codecs for OMNIBUS, and if OMNIBUS is completely functional without JACKMEDIA except for the ability to play one particular file type, is Jack required to provide JACKMEDIA, with all of its source code, when he distributes OMNIBUS? Free Software purists will probably say yes. In their view, Jack could simply provide OMNIBUS without JACKMEDIA’s functionality if he doesn’t like this result. Others may argue that a completely separate dynamically-linked module providing non-core functionality cannot reasonably be called a derivative work, and thus is not bound by the GPL. The “correct”



answer, of course, is you simply don't know until there is binding judicial precedent in your jurisdiction.

## 6.2 Downloads

This section also addresses the issue of making binaries available in a more passive sense (for example, providing an FTP download). In that case, the obligation to provide source code is met as long as the source code is also available through an equivalent means, even though users will not always choose to retrieve it. Note that in this case, the requirement of subpart (b) that the source code be available for three years is not applicable. As long as it was available at the time of download, your obligation is satisfied.



### Hypothetical 6.6

Jack provides OMNIBUS for download under the GPL. He has both the binary executable and the source files available on his server in separate directories. Donna downloads a copy of the binary, but not the source code. After six months, Jack decides to stop distributing OMNIBUS and removes all of the files from his server. Jack is not required to continue providing the source code, because it was available to Donna for download when she downloaded the binary.

## 7 Section 4—Limitations

### 7.1 Automatic Termination

This very short section merely indicates that any attempt to copy, modify, sublicense, or distribute the Program on any terms other than this license will be void and will automatically terminate your license. But anybody who received a copy of the Program from you will still retain her GPL rights, even though they were originally derivative of your rights as a licensee.

Such clauses are common in software licenses, and ensure that the prohibited act terminates the license without further action from the licensor. One of the desirable effects of such a clause is to ensure that acts taken after the automatic termination are copyright violations rather than “mere” breaches of contract. Copyright violations carry some more generous statutory penalties, and give the copyright holder access to powerful remedies like injunctions. This is especially important to the GPL, which the FSF prefers to treat as a conditional grant of license rather than an “agreement.”



### Hypothetical 7.1

Jack downloads a copy of the GPL CONXMP utility from Martha's website. Jack integrates CONXMP into his OMNIBUS media player, which he distributes as a proprietary product. Rose, Donna, and Mickey purchase copies of OMNIBUS from Jack.

Although Jack’s subjective intent was to distribute OMNIBUS as a proprietary product, he didn’t violate the GPL just by selling binary software to Rose, Donna, and Mickey. Remember, there is no prohibition from charging money for GPL software. But when Jack sold the software, he presumably didn’t provide instructions for requesting a copy of the software, or a copy of the GPL itself. So under section 4 of the GPL, Jack’s license was terminated as soon as he distributed OMNIBUS.

But the termination of Jack’s license doesn’t affect Rose, Donna, and Mickey. Section 4 explicitly does *not* terminate the GPL rights they received under section 6 when they received OMNIBUS. And since the GPL doesn’t restrict the act of running the program anyway, they can keep using OMNIBUS for as long as they please.



### Hypothetical 7.2

Jack sells OMNIBUS as a binary product with a copy of the GPL attached and instructions for requesting a copy of the source code. But when Mickey asks for a copy of the source code, Jack refuses.

In this case, Jack was still in compliance when he sold the software. His license didn’t terminate until Mickey asked for a copy of the source code and he refused.



### Hypothetical 7.3

Donna bought a copy of OMNIBUS from Jack, and after hearing about the fracas over Jack’s GPL violations, she decides to send a copy of OMNIBUS to her friend Amy.

The problem here is that section 4 preserves Donna’s GPL rights only as long as she herself is in compliance with the GPL. As long as she’s just running OMNIBUS, she’s fine. But when she wants to send a copy to Amy, Donna has now way of complying with section 3. Section 7 points out that inability to comply with the GPL is no excuse. So Donna is limited in her right to share OMNIBUS.

Although this means that Donna has lost one of the “Four Freedoms” touted by the FSF, the result is actually consistent with the FSF’s philosophy. They want to stop the proliferation of proprietary software, and that’s exactly the result here.

## 7.2 Use After Termination

Notably, this section does not address whether a party that violates the GPL would still be entitled to *use* the software after committing the prohibited act. Since the FSF advocates that a person should not need a license to run legally-acquired software, I believe that the intent is that you could continue using

the software, even after your license is terminated.<sup>17</sup> I base my analysis on the limitations in scope from Section 0, and my belief that this conclusion is consistent with the philosophy of the FSF. The FSF does not want to restrict your right to use the software. They just want to make sure that you pass the freedom on to everybody else. But remember that you may not have received your license from the FSF itself. You received it from the copyright holder, whoever that may be. So the subjective belief or intent of the FSF will not be dispositive. If in doubt, talk to your attorney about what you want to do and how it plays with the GPL.

## 8 Section 5—Acceptance

This section correctly points out that you are not required to accept this license (which is a bit of a relief, since we all know that nobody actually reads software licenses). But the only grant of permission you have to copy, modify or distribute the Program is this license. If you don't accept it, you don't have the right to do those things. So by doing any of those things, you agree to be bound by the terms of the license.

There is some textual ambiguity whether, under this clause, a person may *use* GPL software without accepting the GPL. For example, when the authors say, “you are not required to accept this license,” were they speaking in the abstract legal sense that you are *never* required to accept a contract against your will? Or did they mean, “you are permitted to use the software without restriction, even if you do not accept the terms of this license”? If that is truly what they meant, they might have said so more clearly. The problem here is that there is some tension between ideology and clarity. According to the FSF's Free Software Definition, the first of the Four Freedoms is “freedom to run the program, for any purpose,” and “[b]eing free to do these things means . . . that you don't have to ask or pay for permission.”<sup>18</sup> To Free Software purists, the concept of needing a license to operate and tinker with software is anathema to their philosophy. They don't think you *should* need a license to do those very reasonable things. So they don't want to explicitly grant you that right, lest by doing so they implicitly admit that you didn't have it in the first place.



### Hypothetical 8.1

Donna downloads a copy of OMNIBUS from Jack's web server. After reading the GPL carefully, she is very offended, believing it to be a bunch of “socialist drivel.” She sends Jack a letter in which she says, “I do NOT accept the terms of the GPL for OMNIBUS. I

<sup>17</sup>In any case, the chances of this moderately-interesting question ever being answered are vanishingly small. Somebody would have to violate the GPL, and the copyright holder would need to sue the violating party for its ongoing personal use of the software. Generally, distribution is what the copyright holders care most about. So we will probably never know.

<sup>18</sup><http://www.gnu.org/philosophy/free-sw.html>

refuse to comply with the GPL, but I still intend to use OMNIBUS to listen to music every day. You just try and stop me.”

This issue is mildly interesting, partly circular, and entirely academic. On the one hand, the prevailing legal authority<sup>19</sup> holds that loading a copy of a program into RAM constitutes making a copy of the program, which is ordinarily illegal without authorization from the copyright holder. On the other hand, 17 U.S.C. § 117 permits “the owner of a copy of a computer program” to make a copy that is “an essential step in the utilization of the computer program” (such as loading the software into RAM). If Donna is the “rightful owner” of a copy of OMNIBUS, she has the right to load it into RAM (i.e., use it) without explicit permission from Jack. But since the only license she received is the GPL, and she explicitly rejected the GPL, is she really a “rightful owner”? It doesn’t really matter. The Free Software Foundation has been adamant that the GPL is not a contract, but rather a unilateral grant of a limited license, meaning that in their view, there is no need for the end user to “accept it” to be a rightful owner. And even if the GPL is a contract, the GPL gives Donna *broader* rights than 17 U.S.C. § 117, so by accepting the GPL, she would essentially be agreeing to “do whatever she wants” (a rather circular prospect, both logically and legally). The interesting parts of the GPL only come into play when she copies (other than into RAM), modifies, and/or distributes the Program. In those cases, 17 U.S.C. § 117 doesn’t apply, so Donna must agree to and abide by the terms of the GPL.

## 9 Section 6—Sublicensing

This section automatically grants to anybody who receives the Program (or any derivative work) from you a license equivalent in scope to your own. You may not place further restrictions on her use. But neither are you required to police her compliance with the GPL.

## 10 Section 7—Patents

This section addresses the thorny issue of software patents, which tend to be extremely unpopular in the Free Software community. If you are in a position where you cannot fulfill the terms of this license (for example, as a result of a judgment that you infringe a patent), you are not excused from performance. This means that your license is essentially terminated. The section also includes a savings clause, limited to this section, that indicates that if any part of this section is found to be invalid or unenforceable, the remainder will remain valid and enforceable to the extent possible.

This section will also affect you if you either hold a software patent, or need to license a software patent to run your software. The purpose of this section,

<sup>19</sup>MAI SYSTEMS CORP. v. PEAK COMPUTERS, INC., 991 F.2d 511, 519 (9th Cir. 1993).

which is expressed in the Preamble, is to ensure that if you license a software patent, you license it for all downstream recipients of the software. If you receive only an individual license to the patent, you cannot comply with the GPL. You will need to either negotiate a blanket patent license for all recipients of your GPL software, or you will need to distribute your code under another license.



### Hypothetical 10.1

Jack wants to distribute OMNIBUS with a codec for the XMPL media format. But Martha holds a patent on the XMPL format. Jack negotiates a license with Martha that requires each recipient of OMNIBUS to directly pay Martha a royalty for a license to the XMPL codec.

In this case, Jack cannot distribute OMNIBUS under the GPL because additional restrictions are imposed on recipients, including a fee for the license. Jack may try to negotiate a blanket license with Martha so that all downstream recipients have an automatic license to the XMPL codec. But remember that Jack cannot charge a royalty for those downstream licenses, so he will either need to pay a large fee to Martha himself, or hope that Martha is feeling generous. If Jack is the original author of the entire OMNIBUS project, and he can't negotiate a blanket license, he will need to either drop support for XMPL or choose a license other than the GPL. If OMNIBUS is a derivative work of a GPL Program, and if Jack can't negotiate a blanket license, he will need to drop support for XMPL to be able to distribute OMNIBUS.

The limitations of this section probably do not add anything new to the license. Inability to perform the license does not magically grant an unrestricted license in the copyrighted software; rather, it means that you can no longer use the software pursuant to the license. In that case, you have no other rights than the very minimal rights you automatically have under copyright law.

## 11 Section 8—Territorial Restriction

Because laws vary from nation to nation, and because intellectual property rights are generally territorial, this license may encounter issues in some countries but not others. This section gives the original copyright holder a *retroactive* right to place geographic limits on distribution of the Program, so that it is not distributed in countries where performance of the license would violate some other intellectual property right. This helps to insulate the copyright holder from charges of contributory infringement. Normally, a unilateral retroactive option to modify an agreement without consideration is suspect, and as with other provisions of the GPL, this section has not been tested in court. But since the right is triggered only on specific, narrow conditions, and the exercise of the option is specifically limited, most U.S. jurisdictions will probably enforce this clause.

## 12 Section 9—Revisions to the License

The GNU GPL is updated from time to time. For example, this paper deals specifically with version 2 of the GPL, which is currently the most widely used version. But the FSF has already issued version 3, and is strongly encouraging its adoption.

There are three types of language you may commonly find on a GPL notice.

1. The Program is licensed under a specific version of the GPL. In this case, the licensee may copy, modify, and distribute the Program only according to the terms of the specified version.
2. The Program is licensed under a specific version “or any later version.” For example, many programs indicate that they are licensed under version 2 of the GPL or any later version. The licensee may elect to copy, modify, and distribute the Program according to the terms of either version 2 or version 3 of the GPL.
3. The Program is licensed under the GPL, but no version number is specified. In that case, the licensee may copy, modify, and distribute the Program according to any version of the GPL. Currently, that would give the licensee the right to operate under any one of version 1, version 2, or version 3.

The FSF [encourages licensors](#) to use the “any later version” language because it ensures that downstream users immediately get the benefit if a later version is more permissive, but also ensures that no later version can take away rights (because the user still has the option of operating under the older version). As with section 8, this section grants a unilateral option to retroactively modify the terms of the license (this time in favor of the licensee), and has not been tested in court. But as with section 8, this option is limited enough that it will probably be enforceable in all U.S. jurisdictions.

There is a possible problem to consider with the “any later version” clause.



### Hypothetical 12.1

The evil Kelad Corporation manages, through a number of shady acquisitions, mergers, hostile takeovers, and exterminations, to gain a controlling interest in the Free Software Foundation. Kelad then promptly publishes version 4 of the GPL, which includes the clause, “Any other provision of this License notwithstanding, Kelad Corporation has the unlimited right to do anything it wants with the Program.” Kelad now wants to exercise its option to apply GPL v.4 to all of the GPL software that is licensed under “any later version.”

Although this scenario is as over the top as it is unlikely, the concern is legitimate. What if the FSF “turns evil” (a very subjective question to begin with), or at least makes a new version of the GPL that is murky, confusing, and

ambiguous? What if I just don't like the new version and don't want others to be able to apply it to my software?

Against the “turning evil” scenario, the licensor at least has the vague promise that “new versions will be similar in spirit to the present version...” How much protection this covenant actually affords will be a matter for courts to decide if and when a new license is challenged. Against the possibility of just not liking a future version, your only defense is to specify a particular version of the GPL and stick with it. This is my personal preference as I am naturally wary of agreeing to licenses that don't even exist yet.

## 13 Section 10—Multiple Licensing

You may want to incorporate code from multiple free or open source projects into your own software, and each may be licensed under the author's license of choice. This can create problems if the terms of the licenses clash. The safest advice when mixing licenses is, if in doubt, ask for permission. If what you are doing fits generally within the spirit of the GPL, there's a good chance the original author will let you do what you want to do.

### 13.1 Derivative Works

If you are working on a GPL Program and want to incorporate code that is licensed under another “Free Software License” approved by the FSF, you may want to check their compatibility list.<sup>20</sup> The list is a little thick on ideology (for example, it will consistently recommend that you not use a license that is incompatible with the GPL, which they frankly believe should be used for all software), and the list *should not be taken as legal advice*, but it is a useful reference to see what those who know the GPL best have to say about compatibility. If the code you want to use is listed as GPL-compatible, it's unlikely you will ever run into any problems from mixing the code.



#### Hypothetical 13.1

Jack wants to distribute OMNIBUS under the GPL. He also wants to include support for the XMPL format, but do not want to write his own codec from scratch. He has a copy of Martha's XMPL codec, which she distributes under the FreeBSD license. Jack checks the compatibility list and sees that FreeBSD is listed as a GPL-compatible Free Software License.

In this case, Jack can safely incorporate the XMPL codec into OMNIBUS. The FreeBSD license will not prevent him from fulfilling the terms of the GPL. When he distributes OMNIBUS, he may license the entire project, including

<sup>20</sup>As of the date of this writing, the list is maintained at <http://www.fsf.org/licensing/licenses/index.html#GPLCompatibleLicenses>

the XMPL codec, under the GPL.<sup>21</sup> This is permissible because the FreeBSD license is not one of the so-called “copyleft” license, which require you to provide the same license to your sublicensees.

But note that GPL-compatibility may be a one-way street.



### Hypothetical 13.2

Jack wants to distribute OMNIBUS under the highly-permissive FreeBSD license. He wants to include XMPL support, but doesn't want to write his own codec from scratch. He has a copy of Martha's XMPL coded, which she distributes under the GPL. Jack checks the compatibility list and sees that FreeBSD is listed as GPL-compatible. He believes that this means he will be able to distribute his entire OMNIBUS project, including the XMPL codec, under the FreeBSD license.

Jack is wrong, of course. By incorporating Martha's XMPL codec, he has made OMNIBUS a derivative work of a GPL Program. That means the work *must* be distributed under the GPL. GPL compatibility means that a program with that license is safe to incorporate into your GPL Program, not vice versa.

## 13.2 Dual Licensing Your Work

The GPL does not *give* you the right to dual license your work, but you may *inherently* have the right to do so. This distinction is important because you can only dual license when you have an absolute license in the work to begin with. If you are the original author (or own the entire copyright, or received it under a highly-permissive license like the FreeBSD license) in a work, you may choose to distribute that work under a dual license, meaning that some users receive it under the GPL (usually for free), and others receive it under a more permissive commercial license (usually for a fee). This model has been used by companies like Trolltech for its Qt toolkit<sup>22</sup> and Oracle for its MySQL database.

If you are thinking about dual licensing, there are some issues you should consider carefully.

- Your proprietary distribution will definitely be incompatible with the GPL. Make sure you have the necessary rights in every piece of software you are distributing under a proprietary license. Even if you are the original author, make sure your project does not include improvements that were licensed back to you under the GPL.<sup>23</sup>

<sup>21</sup>Note that if OMNIBUS is a derivative work of a GPL Program, Jack is *required* to license the XMPL codec under the GPL.

<sup>22</sup>But note that Nokia acquired Trolltech in 2009, and the latest version of Qt is licensed completely under the LGPL

<sup>23</sup>For example, Linus Torvalds could not dual license the Linux kernel, even though he is the original author; the kernel contains many improvements from community contributors who have licensed their improvements under the GPL.





### Hypothetical 13.3

Doc is the original author of SIDRAT, a GPL astronomy program that simulates the movement of the stars. Martha, Jack and Rose have all contributed improvements to the GPL SIDRAT, and licensed them back to Doc under the GPL. Doc would not be able to create a proprietary version of SIDRAT without either negotiating a new license with Martha, Jack, and Rose, or completely removing all of their contributions from the proprietary version.

- Decide whether you are prepared to “fork” your project. One of the benefits of the GPL is that licensees can and do make improvements to the software. You can incorporate these into your GPL version, but you must keep them out of your proprietary version.



### Hypothetical 13.4

Doc is the original author of SIDRAT, a GPL astronomy program that simulates the movement of the stars. He sells a proprietary version with some extras, but also makes a basic version available under the GPL. Over time, Martha, Jack, and Rose all contribute to the GPL version. Doc cannot incorporate their work into his proprietary version. Doc also makes his own improvements to put in the proprietary version that he does not add to the GPL version. Over time, the two versions become significantly different.

- If you want to avoid a “fork,” you will need to either keep community contributions out of your source tree or acquire the proper licenses from the contributors. Some contributors may be unwilling to license their improvements to your for inclusion in your proprietary version. In those cases, you will have to reject those contributions.



### Hypothetical 13.5

Doc is the original author of SIDRAT, a GPL astronomy program that simulates the movement of the stars. He sells a proprietary version with some extras, but also makes a basic version available under the GPL. He wants to ensure that both versions have the same code base, so he rejects contributions from Martha, Jack, and Rose. Martha, Jack, and Rose are upset that Doc won’t accept their contributions, so they create their own project based on the GPL SIDRAT and call it VORTEX.

- Some community developers, especially Free Software purists, may be turned off by the fact that you are even selling a commercial version.

This may mean that they are unwilling to contribute to your project,<sup>24</sup> whether or not you fork it.



### Hypothetical 13.6

Doc is the original author of SIDRAT, a GPL astronomy program that simulates the movement of the stars. He sells a proprietary version with some extras, but also makes a basic version available under the GPL. Doc thinks that Martha, Jack, and Rose have some good code contributions, and he wants to incorporate those into both versions. But Martha, Jack, and Rose are unwilling to let their volunteer efforts be used for profit. Doc hires them as full-time employees so that he can use their work in both versions.

- Remember that the GPL permits commercial use of the Program. So make sure there is a meaningful incentive to use your proprietary version.



### Hypothetical 13.7

Doc wants to have a GPL version of SIDRAT so that he can increase its popularity, but he also wants to be able to sell a proprietary version because universities are interested in reselling customized versions to their students. On the other hand, NASA likes the the GPL version, and sees no need to buy the proprietary version, because they don't distribute software.

- If the incentive for buying a commercial license is the inclusion of value-added proprietary extras (like skins, utilities, support, or warranty service) that are not derivative works of the GPL Program, remember that you may not need a dual license. You can license the Program itself under the GPL and sell the extras.



### Hypothetical 13.8

Doc provides SIDRAT under the GPL so that he can increase its popularity. He also sells a proprietary boxed version with professional labels and packaging. The boxed version contains a much larger database, including more stars, and more information about each star, and a simplified installation program. Doc provides the SIDRAT source code under the GPL, but sells the boxed version with a restrictive per-user license. Although users can get the identical software by downloading

---

<sup>24</sup>For most projects, this may not be a major drawback. My personal observation is that the software “bazaar,” where a large community of diverse volunteers contribute code to a project, is a myth in most cases. Most contributions for most projects come from a semi-structured core group. If you can get them to buy into your dual license (for example, by hiring them as full-time employees), you will probably be in good shape.

the source code, there is an incentive to buy the boxed version to get the simplified installer and better database.

If you have a handle on all of those issues, go with a dual license if it makes sense. But remember that your competitors are still free to use the GPL version in competition with you if they can find a way to make money with it.

## 14 Section 11—Warranties

This is a standard warranty disclaimer. It means that the author or licensor takes no responsibility for the program. It excludes even the implied warranty that the program is suitable for what it purports to do. In short, when you use GPL software, you do so at your own risk. If a program is intended to display a smiley face on the screen, there is no promise that it will not accidentally overwrite your entire directory structure and cause your hard drive to malfunction so that the platter is physically damaged. If you're not okay with that, don't use the program (or buy it from a distributor who also sells warranty coverage with it).

If this seems weaselly to you, note that most commercial software comes with a similar warranty disclaimer. Windows could do the same thing, and you would have no recourse against Microsoft (and presumably, you paid good money for the warranty-free Windows).

## 15 Section 12—Limitation on Remedies

To the extent permissible by law, which will vary by jurisdiction, the license limits your recovery of any general, special, incidental, or consequential damage arising from your use of the program. As with section 11, this is a very common disclaimer in software licenses. Most commercial licenses will limit your recovery to whatever you actually paid for the software. Since it is most likely that you received the GPL Program for free, you will have a hard time recovering any money from anybody if there is a problem. The thrust of this section (as well as section 11) is that you use this software at your own risk.

That said, most free software is not dangerous. Very rarely will it have a problem more serious than simply crashing.<sup>25</sup>

## 16 Conclusion

You should now have an idea of whether the GPL makes sense for you. You should have thought about some critical questions, and considered both the costs and the benefits of either releasing your own work under the GPL, or

---

<sup>25</sup>And this does not imply, even, that free software is inherently unstable or crash-prone. Many free software programs are extraordinarily stable and reliable; sometimes more so than their commercial counterparts.

using somebody else's GPL Program. If the GPL doesn't seem right for you, remember that there are many other licensing options, even if you want your software to be free and/or open source. As always, you should consult your attorney for an analysis of exactly how the GPL and other licenses apply to your specific situation.

## A The GNU GPL

GNU GENERAL PUBLIC LICENSE  
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA  
Everyone is permitted to copy and distribute verbatim copies  
of this license document, but changing it is not allowed.

### Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free

software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE  
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to 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. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You 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 under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- 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.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are



prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. 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. 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.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

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. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

#### NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED

TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

#### END OF TERMS AND CONDITIONS

#### How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify  
it under the terms of the GNU General Public License as published by  
the Free Software Foundation; either version 2 of the License, or  
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License along  
with this program; if not, write to the Free Software Foundation, Inc.,  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type 'show c' for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate

parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program  
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989  
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.