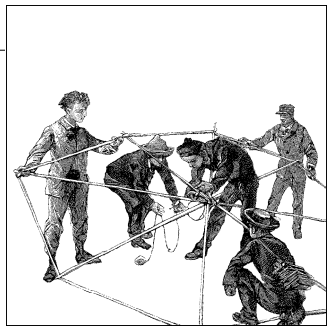
CHAPTER 3

The GPL, LGPL, and Mozilla Licenses



The licenses described in this chapter are very different from those described in Chapter 2. These licenses impose substantial limitations on those who create and distribute derivative works based on works that use these licenses. The GNU General Public License (the GPL License) explicitly requires that derivative works be distributed under the terms of the GPL License and also that derivative works may *only* be permitted to be distributed under the terms of the license. The Mozilla License imposes different and less restrictive terms on the licensing of derivative works. Both of these licenses (and a variation of the GPL License) are described in some detail in the following sections.

Before going into detailed descriptions of these licenses and their effects, it's a good idea to re-examine the limitations imposed by the licenses described in the previous chapter, if only for contrast.

The MIT License, probably the simplest of those licenses, imposes almost no restrictions on licensees and no meaningful restriction at all on licensees distributing derivative works. When the original work or "substantial portions" of it are distributed, the licensee is required to include a copyright notice and the notice giving permission to potential licensees of their rights to use the work. The licensee is not even required to include the disclaimer of warranties that was part of the original license. (Such licensees may, however, have good reason to include that disclaimer—in particular, to protect themselves from potential liability.)

The MIT License does not impose even these restrictions on licensees who choose not to distribute it or "substantial portions" of it, but rather only works derived from it. Such licensees need not include the copyright notice, the disclaimer of warranties, or the permission notice. As described in the previous chapter, this allows the creator of a derivative work to license that new work in any way that he may choose, whether under a proprietary license or under the MIT or another open source license.

By contrast, the BSD License, both pre- and post-1999, imposes explicit limitations on distribution of both the original and derivative works.* These limitations include the inclusion of the enumerated terms of the license so that these limitations will also govern the use of the derivative work: the non-endorsement provision, the copyright notice, the acknowledgment of the creator of the original work, and the inclusion of the disclaimer of warranties. These enumerated limitations, however, do not require that the creator of the derivative work license under terms no more restrictive than those applicable to the original work. Accordingly, as noted in the previous chapter, so long as these conditions are complied with, the creator of the derivative work may then license that work under a proprietary license, under another open source license, or under the BSD License, so long as the terms of that license do not conflict with the limitations of the BSD License. There is no requirement, for example, that the creator of the derivative work make the source code of that work available to others.

The licenses discussed in this chapter impose much more specific limitations on the way in which derivative works may be licensed. Essentially, by using a work licensed under the GPL, the LGPL, or the Mozilla Licenses, the licensee is agreeing not only to respect those limitations with regard to his or her own use of the licensed work but to impose those limitations (and with regard to the GPL and LGPL Licenses *only* those limitations) on licensees of any derivative work that he or she may choose to create from the original work.

GNU General Public License

The GNU's General Public License, or GPL, is one of the foundation open source licenses. Created by the Free Software Foundation (FSF), which has made many contributions to open source coding, it is the preferred license for projects authorized by the FSF, including the GNU Emacs Editor and the GNU C Compiler, among literally scores of others, including the GNU/Linux kernel.

The intentions behind the license and the premise underlying it are explained in the license's preamble, which is included here in its entirety. The preamble follows the copyright notice,† and a notice that prevents modifications, ironically enough, to the license itself: "Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed." While the license permits the creation of derivative works from the licensed *code*, it does not permit the creation of derivative licenses from the license itself.

* The BSD license phrases this as "Redistribution and use [of the work] in source and binary forms, with or without modification," a clause that seems intended to govern the distribution of both the original and derivative works. Whether a derivative work that incorporated only a small part of the BSD licensed work could reasonably be described as a "work . . . with modification" is, admittedly, arguable. A better reading of the license would bring derivative works within the enumerated restrictions, as this appears to be the intent of the license's drafters.

† The license described is Version 2.0 of the GPL and is Copyright © 1989, 1991 Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

> Copyright (C) 1989, 1991 Free Software Foundation, Inc.
> 59 Temple Place - Suite 330, Boston, MA 02111-1307, 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 Library 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.

This preamble clearly and concisely sets out the three main purposes of the GPL. The first, and by far the most important, is to keep software free, in the sense that it can be distributed and modified without additional permission of the licensor. This imposes a mirror-image restriction on the licensee: while the licensee has free access to the licensed work, the licensee must distribute any derivative works subject to the

* The most current GNU Library General Public License is now known as the GNU Lesser General Public License (LGPL) and is described in more detail later in this chapter.

same limitations and restrictions as the licensed work. The second purpose of the GPL is to ensure that licensees are aware that software under the license is distributed "as is" and without warranty. This purpose is not unique to the GPL, as we have seen. The third purpose (which is really a variant of the first) is that the licensed software be free of restrictive patents: to the extent that a patent applies to the licensed software, it must be licensed in parallel with the code. As we discussed in Chapter 1, a given piece of code may be subject to both a copyright and a patent. In order for the GPL to function properly, both copyright and patent licenses must be subject to the terms of the GPL.

The individual provisions of the license articulate each of these purposes in some detail. The GPL License is written with a great deal more specificity and in substantially more detail than the licenses described in the previous chapter. This meticulousness is obvious in the license's first provision, which defines the scope of the license and its critical terms.

> 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.

The term "Program" is roughly equivalent to what this book has described previously as "work," the term "work based on the Program" to derivative work, and the term "you" to licensee.* The exclusion of activities other than copying, modifying, or distributing the program or a work based on it is typical of the meticulousness of this license. This exclusion could reasonably be assumed to apply to the licenses discussed in Chapter 2, but only here is it specifically described.

The next provision describes all of the limitations that apply to distribution of the licensed work.

> 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

* The terms "work" and "derivative work" are terms of art defined by copyright law.

> 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.

This provision embodies the most important principles with regard to the distribution of the original work. The original licensed work can be distributed or sold by a licensee. This provision by itself creates the opportunity for a profitable business—any person can simply acquire and package GPL-licensed software, perhaps bundle it with an appropriate manual,* market it, and sell it. There is no need to "add value" other than by making the work available in a format convenient to consumers. The limitation, obviously, to this business model is that any other person is equally free to start a business on the same principles and distribute the same work or works. This is not necessarily fatal to such businesses. Businesses do not need to be monopolies in order to prosper. The FSF itself derives a substantial amount of income from distributing its own "free" works.

The other business model identified by this provision is the warrantying of a particular work. Any person can take a GPL-licensed work and sell a guarantee that the work will perform a particular function and make whatever changes or modifications to the work are necessary to achieve that goal. As previously noted, most open source licenses, including the GPL, expressly disclaim warranties of any kind. However, businesses strongly prefer to have reliable software, and, in particular, to have software that is backed up by knowledgeable professionals who are capable of adapting it to particular purposes and situations. This type of "value-adding" is expressly authorized by the GPL.

The second paragraph of the GPL is its most important, as it embodies the FSF idea of "copyleft," a variety of the generational limitation described in Chapter 1, which requires that derivative works be subject to the terms of the GPL and only the terms of the GPL.

> 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:

This is the first part of copyleft: subject to certain restrictions, modifications to the work or any part of it are permitted.

> a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

Like the restriction imposed by the BSD license, this provision serves to ensure that users are aware that the derivative work is not identical to the original work and to

* The application of open source licensing principles to works other than software, including manuals, is described in Chapter 5.

identify the person or persons who are responsible for the changes. This is intended to protect users and to protect the reputations of creators of work against injury arising from flawed derivative works.

> 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.

This key part of the second paragraph of the GPL is the most important provision of the license. Derivative works must be licensed under the GPL and be subject to all of its restrictions. Unlike works licensed under the MIT or the BSD License, works derivative of work licensed under the GPL (or the original work itself) may not be made proprietary or otherwise limited in their distribution. If a programmer is looking to create proprietary works, the entire universe of GPL-licensed software is closed off to her. Indeed, as described in Chapter 6, the inclusion of any GPL-licensed code in purportedly proprietary software could prevent the creator of that software from enforcing any of the rights otherwise available under copyright: any person could distribute, sell, or modify that software, in disregard of any rights that would otherwise be granted the creator under the copyright laws.*

> 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.)

This provision is a necessary complement to provision 2(b). The fact that licensees of the derivative work may freely exercise rights under the GPL is of little importance unless those licensees *know* that they can exercise those rights. This provision attempts to inform those licensees of those rights.

> These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can reasonably be 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.

* Whether such a person would in turn be bound by the GPL is discussed in more detail in Chapter 6.

It is not uncommon for a particular program to be capable of both integration with other software to form a unified whole, such as into a calculator program that performs a variety of functions, and also functioning with minimal or no modifications as a separate entity, such as a program that only calculates square roots. This provision of the GPL allows the author of such software to license the software under another license (typically, a proprietary one) when distributed by itself and under the GPL when the program is distributed as part of a larger work, including GPL-licensed programs. This may provide some benefit to the software developer, but probably not if the developer chooses to distribute the GPL-licensed software publicly. In general, consumers would prefer to acquire the GPL-licensed work, which will likely have greater functionality and be more cheaply available, than to acquire the more limited proprietary work.

This provision may be of some comfort to software developers who are creating software primarily for their own use "in-house." Presumably, such developers could write programs or functions designed to work with GPL-licensed programs and simply limit the distribution of that GPL-licensed code to persons within the organization. If the developers' own code got to the point where it could be commercially distributed on its own, the developers could, with confidence, "disengage" that code from the GPL-licensed code and distribute it as part of standalone programs under a proprietary license or otherwise.

> 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.

This provision serves as a safeguard against overly broad interpretations of the GPL. This makes explicit what is implicit in the rest of the license: the provisions of the GPL are not contagious, like a cold. Mere proximity does not cause the license to govern a particular piece of code. To fall under the copyleft, the code must be integral to and/or derivative of a program that is GPL-licensed.

The following provisions of the GPL require that the licensees of the GPL-licensed code make available in one of two ways the source code to the program. The right to create derivative works from a program is obviously limited in practice if the source code is not available.

> 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:

Note that this requirement is equally applicable to derivative works created under Section 2 of the GPL.

> 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,

This is the most favored way to make source code available. It requires no additional effort from the distributee and is not time-limited. This is the best way to comply with Section 3 for all but the largest programs.

> 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,

This option furthers the purposes of open source and free software but does so in a way that imposes additional costs on both licensors and licensees. The licensor must maintain a facility for providing copies of the source code; the licensee interested in creating the derivative work must contact and pay for the copying of the source code. Moreover, this provision is limited to three years, which could result in potentially useful software "going closed" as a practical matter (at least for the creation of derivative works) once the licensor ceases making the source code available.

> 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.)

Section 3(c) allows noncommercial distributors of GPL-licensed software to "piggyback" on the original licensor's offer to make the source code available, if the source code of such software was originally made available under Section 3(b).

The following paragraph of the GPL defines "source code" as that term is used in the license.

> 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.

This limits the size of the source code that needs to be provided by narrowing the definition of program to exclude major components, like the operating system the program is intended to run on. Obviously, if the GPL-licensed programs being distributed (or one or more of them) are themselves major components of an operating system, the source code for those components must be made available, as described in 3(a–c).

> 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.

This is another provision that explains in greater detail something already implicitly stated elsewhere in the license. Offering access to copy the source in the same manner and with the same degree of ease as the executable code is sufficient to comply with the requirements of Section 3(a).

The first part of Section 4 of the GPL identifies the license as the exclusive license for use of the licensed software.

> 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.

In the event that a licensee violates any term of the GPL by, for example, distributing a proprietary derivative work based on GPL-licensed code, all rights under the GPL are voided. This brings back into play the ordinary protections of copyright law (and of patent law, if applicable) described in Chapter 1. In the event of such a breach, the ex-licensee would become legally liable to the licensor for violation of the copyright. The licensor could enjoin the ex-licensee from distributing the derivative work and could sue for damages, which could include, among other things, any and all profits the ex-licensee made from distributing the derivative work. This scenario is described in more detail in Chapter 7.

> 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.

This sentence acts as a savings clause, preventing liability from attaching to those persons who received the licensed work or a GPL-distributed derivative work from the ex-licensee.

Section 5 addresses a problem that applies to almost all software licenses: the uncertainty as to whether a binding contract is in fact created between the licensor and licensee.

> 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.

While no court has yet ruled on the effect of this provision, it is likely enforceable. As noted in Chapter 1, courts have found that "shrinkwrap" licenses—proprietary licenses that the licensee accepts by breaking the shrinkwrap on commercial software—are enforceable. The GPL can rest firmly on the fundamental (and intrinsic) protection of copyright. The licensor owns every part of the work and any use of it (excepting "fair use") is infringement. The potential licensee is thus faced with a choice: either refuse the GPL, which bars almost every use of the licensed work, or accept it, and use the work as permitted by the GPL. As described in more detail in

Chapter 6, knowledge of the applicable license should be implied even as to putative licensees who have no actual knowledge of the license. Some degree of diligence should be required of such users: if they truly believed that there was "no license" applicable to the program, they should have made no use of it at all other than the very limited uses permitted by copyright law.

Section 6 of the GPL creates a relationship between the licensor and each of the licensees, regardless of the number of generations of distribution that may lay between them.

> 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.

The GPL should be effective, regardless of the number of distributions through which it passes, because of the limitations and requirements of Sections 1 through 4. This provision, therefore, acts in some way as a back-up to those sections. More importantly, however, it also tries to create *contractual privity* between the licensor of the original work and all the licensees of that work.

Contractual privity is the legal state between two (or more) parties in which they are bound by contractual obligations to each other. In the GPL, it safeguards the *standing* of the licensor to bring a lawsuit against all the licensees of the work. *Standing* is a legal term of art, but, in simple terms, it means essentially that the person with standing to sue has been directly injured in some way by another such that that person has the right to bring an action for relief. Without this contractual privity creating the standing on the part of the licensor to sue, a licensee of the work could argue that she did not receive the license to use the work from the original licensor, but rather from some intermediate distributor (who may have no interest at all in defending the terms of the license), and that, accordingly, only that intermediate distributor has standing to sue for putative violations of the license. Section 6 attempts to head off this argument, by creating a relationship between the original licensor and all licensees of the work, regardless of the number of distributors.

The second sentence of Section 6 is the mirror image of Section 2(b). As that section required that derivative works be distributed subject to the restrictions of the license, so this sentence prohibits the addition of any restrictions to those present in the GPL.

> You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

As described in Chapter 6, this limitation has significant consequences on the compatibility of the GPL with other licenses.

The third sentence prevents liability from attaching to innocent distributors for license violations committed by distributees or any other person.

> You are not responsible for enforcing compliance by third parties to this License.

Section 7 prevents any outside act, including court judgments premised on patent rulings or otherwise, from limiting or altering the terms of the 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.

This section is constructed so that in the event that any court attempts to limit or modify the license by imposing obligations or restrictions inconsistent with the GPL, the license for all practical purposes ceases to exist.* Because of this, patent issues remain something GPL developers need to watch.

In practice, this could have dire consequences on the consumers of GPL-licensed software. Say, for example, that a small software company determines that part of a widely distributed and used GPL-licensed program, such as the GNU/Linux kernel, infringes on a software patent that it holds. The company brings suit and a court determines that the program infringes on the patent. Because the infringing part of the program is relatively trivial, the court determines that the appropriate remedy is for every licensee to pay a one-time fee of one dollar to the company. While both current and future licensees (or at least some of them) would gladly pay the fee and continue to use the software, this payment, because it is a restriction not part of the GPL license, is inconsistent with the license. Accordingly, Section 7, were it to be enforced, would bar any distribution of the program after the court judgment.†

Like the rest of the GPL, this section has not been interpreted by a court. However, it is unlikely that a court would allow this section to limit its own power to grant relief. It is certainly not impossible that a court in adjudicating such a dispute would give notice to the licensor and permit the licensor to make appropriate arguments concerning the license, but would then grant relief that would essentially rewrite the GPL in favor of the injured party and permit licensees to continue to copy, distribute, and modify the affected program. This is, of course, only my speculation.

* For an interesting discussion of the effects of the bringing of a patent infringement action on a licensor's ability to continue to distribute under the GPL a work that the licensor itself claims violates its own patent rights and the subsequent effect of this on the GPL's compatibility with the Apache License, v2.0, see *http://www.apache.org/licenses/GPL-compatibility.html*. As of this writing, the FSF has taken the position that the Apache License, v2.0, is incompatible with the GPL because the Apache License, v2.0, has a slightly different treatment of this scenario.

† Because Section 7 refers only to distribution, and because Section 0 limits the application of the license to "copying, distribution and modification," the licensees could continue to run the affected program. However, the licensees could not copy, distribute, or modify the program, drastically limiting its usefulness in the open source/free software model.

> 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.

This sentence is another variety of the savings clause, intended to preserve the remaining parts of this section even if some part of it is invalidated by a court. It is hard to see, however, what effect any part of this section could have if the critical part of it is superseded by a court, as described earlier.

The following part of the section does not really apply a legal limitation on licensees as much as it articulates a defense of Section 7.

> 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.

The thesis is that the licensor's choice to use the GPL license is, in some sense, a political one, and that choice should be protected and defended against encroachment. Licensees, obviously, may see the situation differently.

Although the potential results from the application of Section 7 may seem draconian, Section 7 is probably necessary to protect the integrity of the GPL and of the GPL distribution model. The license prevents licensees from altering the GPL contractually, through provisions that are very likely to be enforced. However, private parties are not the only entities capable of altering legal obligations. Courts have an even greater power, to alter, to cancel, and to rewrite contracts to effect appropriate relief on any number of grounds. The GPL's use of a strategy of "if we're not playing my game, I'm taking my ball and going home" is probably necessary to prevent the model from being undermined by courts. How the courts will react to the restriction of Section 7, however, is still unknown.

Section 8 addresses a similar problem, where the laws of certain jurisdictions would limit or otherwise modify the GPL.

> 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.

To the extent that there are jurisdictions in which the licensor is limited from licensing the program due by pre-existing patents or copyrights, the licensor is free to carve

them out from the area in which the GPL is effective. This gives the licensor maximum flexibility, by permitting the GPL-licensed software to spread as widely as possible, if it is restricted in certain jurisdictions. This is an example of an area in which the GPL can itself be modified, at least under one set of circumstances.

Section 9 of the GPL gives notice that the FSF may issue updated or revised versions of the 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.

Unlike most of the other parts of the GPL, this provision really serves to give notice to potential licensors—i.e., those who choose to use the GPL to license a new program—not to licensees.

> 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.

This paragraph makes clear that in order to preserve specific guarantees or rights, licensors should identify the GPL version used by version number. If they do not, the licensee can exercise rights under *any* of the GPL licenses. Moreover, if the licensor adds the language "and any later license" following the identification of the version number, the licensee can exercise the rights under that version and any subsequent version of the GPL.

A different option, albeit not one permitted by the GPL, would be the inclusion of language to the effect that "this software is licensed under GPL Version X.Y. This license is subject to periodic revision and amendment by the Free Software Foundation. Upon publication of such a revised or amended license by the Free Software Foundation, such revised or amended license is deemed to have superseded the license previously applicable to this software, and such revised or amended license shall from that time govern the contractual relationship between licensors and licensees. Accordingly, any further copying, distribution, or modification of this software after that time will be subject to the terms of the revised and amended license. You have the obligation to track such revisions and amendments to the GPL."*

This option was not included in the GPL, most likely because it further complicates the already somewhat thorny issues related to providing notice of the license to licensees and forming a binding contract between licensor and licensee described in Chapter 6. It is one thing to expect a licensee to be bound by the terms of a license, which are made clear to the licensee upon the first use of the program; it may be something entirely different to require that licensee to track the actions of the FSF

* A similar provision is contained in the Mozilla Public License described later in this chapter.

and conform behavior accordingly. Nonetheless, if those issues could be addressed, this option could offer some benefits, particularly in allowing the FSF to address threats to the GPL, such as those described in Section 8.

Section 10 is less a binding provision than an explanation to licensees as to how to address the GPL's incompatibility with other licenses.

> 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.

As we have seen, the second sentence of Section 6 of the GPL bars licensees from imposing any additional restrictions on recipients' exercise of rights under the license, and Section 4 terminates all rights under the license in the event that any provision is not complied with. The effect of these two sections is to make the GPL incompatible with most other open source licenses.* Section 10 provides a possible solution, although one that may be impractical in many situations. The original licensor of the program, holding the copyright to the software and having licensed the software under the GPL, cannot withdraw or alter the terms of the license already granted; the licensor, however, in addition to licensing the software under the GPL License, can also license it under another license, such as the Artistic License. If the original licensor is willing to undertake such parallel licensing, the code can be made available under a non-GPL compatible license and thereby avoid the problem.

The rest of the license consists substantially of disclaimers of warranty similar to those in the licenses described in the previous chapters. These disclaimers are also in all-caps.

> 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

* Not all licenses are incompatible, however. For a list of licenses the Free Software Foundation considers compatible with the GPL, which include the MIT (or X license) and the post-1999 BSD license, see *http://www.gnu.org/licenses/license-list.html*.

> 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

One interesting, potentially significant distinction between this disclaimer of liability and those discussed in the previous chapter is that the disclaimer does not expressly disclaim liability for "direct damages." As discussed in the previous chapter, direct damages are measured by the price of the software alleged to be defective. This decision to exclude direct damages may be deliberate—it would not be inconsistent with the ideas underlying the GPL to hold distributors liable for the price of the software, for example, if it was poorly copied. However, the better reading of the provision is that it disclaims all damages, and that the list of "general, specific [etc.]" damages that are disclaimed is illustrative, not definitive.

More importantly, Sections 11 and 12 permit one kind of modification to the GPL, in that they permit a separate written agreement between two parties to establish warranties or permit suits for damages. One business model that is available for open source is the provision of warranties and maintenance of open source and free software. The GPL does not prohibit the provision of such services by inserting these exceptions into Sections 11 and 12. Such services are also explicitly authorized in Section 1.

As noted in the notice immediately following Section 12, this is the close of the provisions of the license. The remainder of the GPL text is instructions for implementing the license, which follow here:

> 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)
> >
> > 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., 59 Temple Place, Suite 330, Boston, MA 02111-1307 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 Library General Public License instead of this License.

GNU Lesser General Public License

The GNU Lesser General Public License (LGPL) is another license created by the FSF for the purpose of permitting a certain class of programs, generally subroutine libraries, to be licensed under an FSF license but be permitted to link with non-GPL software programs. Subroutine libraries provide various functions to other programs, and because as part of their function they link with such programs, the resulting program plus library could be considered as a legal matter to be a derivative work. Accordingly, if the other program were licensed under a proprietary license and the library under the GPL and the program and library were distributed together under the proprietary license, the GPL would be violated, as the program plus library would be considered a derivative work that would be subject to limitations on copying, distribution, and modification that are inconsistent with the GPL.*

* The *use* of a GPL-licensed program with a proprietary-licensed library (or any other program, whether under a proprietary license or some other non-GPL license) is not a violation of the GPL license. Rather, the GPL license comes into play only when the GPL-licensed software is copied, distributed, or modified—none of which is implicated by the simple use of the software. As explained in more detail later, libraries present some unique technical problems for licensing in that their use may result in the "modification," as that term is defined in the GPL, of the program that uses them.

The LGPL provides an alternative license that preserves many of the benefits of the GPL model for such libraries—in fact, the Lesser General Public License was in its first incarnation known as the Library General Public License. LGPL-licensed libraries can be linked with non-GPL licensed programs, including proprietary software. However, libraries need not be licensed under the LGPL, and as the following preamble to the license points out, the preferable way to license libraries, at least under some circumstances, is under the GPL.

> Preamble
>
> The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users.
>
> This license, the Lesser General Public License, applies to some specially designated software packages—typically libraries—of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.
>
> When we speak of free software, we are referring to freedom of use, 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 and use pieces of it in new free programs; and that you are informed that you can do these things.
>
> To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.
>
> For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.
>
> We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.
>
> To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.
>
> Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

> Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.
>
> When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.
>
> We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.
>
> For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.
>
> In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.
>
> Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.
>
> The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.*

Much of this preamble parallels the language in the GPL described earlier. There are two new points, however, worth identifying. The first is the decision on the part of the developer as to which license to use for a particular library. The Preamble posits this choice as if it were between the GPL on one hand and the LGPL on the other. To begin with, obviously, developers can choose to license their programs, including their libraries, under any license, FSF-approved or not. For those who are interested in using the GPL-distribution model, however, the Preamble identifies those situations in which LGPL may be favored, such as when the library is intended to replace an already available commercially licensed product.

* This version of the LGPL is 2.1, distributed February, 1999. It is copyright © 1991, 1999 by the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA. As was the case with the GPL, "Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed."

The second point worthy of mention is the distinction in the LGPL between "work based on the library," which is subject to essentially the same restrictions as imposed by the GPL, and "work that is used with the library," which is not. This distinction is explained in more detail later.

As was the case with the GPL, the first section after the "Terms and Conditions for Copying, Distribution, and Modification" is Section 0, which defines the basic terms used in the license and sets out its fundamental premises.

> 0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

The next full paragraph defines small-l "library" as it is used in the LGPL.

> A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The next paragraph defines capital-L "Library," a term of art used to refer to the licensed program, and "work based on the Library," another term of art that is equivalent to this book's use of "derivative work."

> The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

In contrast to the GPL, the LGPL also includes a definition of "source code" in this section; the parallel definition is in Section 3(c) of the GPL.

> "Source code" for a work means the preferred form of the work for making modifications to it. For a library, 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 library.

This is most likely included here to include a number of files related to the library—modules, interfaces, and scripts—to maximize the functionality of the source code.

The final paragraph of Section 0 is substantially identical to the paragraph found at the end of the GPL's Section 0.

> Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

Many of the provisions of the LGPL are identical or near-identical to provisions in the GPL. Accordingly, the annotations in the section focus on those provisions in which significant changes have been made. Examine the earlier discussion of the GPL if you have questions about any of the provisions that are not thoroughly discussed here.

Section 1 of the LGPL is substantially identical to Section 1 of the GPL, except that it refers to the "Library" instead of to the Program.

> 1. You may copy and distribute verbatim copies of the Library's complete 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 distribute a copy of this License along with the Library.
>
> 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.

Section 2 of the LGPL appears to be substantially identical to the equivalent section of the GPL. There are, however, a few noteworthy changes relating to the specific qualities of libraries, including one that sharply limits the LGPL's applicability to programs other than libraries.

> 2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

This paragraph reads substantially like the first paragraph of the GPL's Section 2, again with the distinction that it uses "Library" in place of "Program."

The first of the clauses of this section, however, imposes a limitation absent from the GPL, i.e., it limits the type of derived work that can come from an LGPL-licensed program.

> a) The modified work must itself be a software library.

In Section 0, the LGPL had noted that the license applied to "any software library or other program." This provision, however, limits the ability to create derivative works to those circumstances in which the resulting work is a library, as that term is defined in the LGPL. This may complicate, or, more likely, entirely prevent the creation of derivative works from programs that are licensed under the LGPL but are not software libraries. If the LGPL were to permit such derivative works to be made from programs other than software libraries, Section 2(a) should have read something like "The modified work must itself be a software library if the Library [i.e., the original work] is itself a library." Note that the definition of big-L Library under the LGPL includes both small-l libraries and "work" that has been distributed under the license. This bar on the creation of derivative works other than libraries from LGPL-licensed works makes the LGPL essentially useless as a license for such works. Creators of such works should look to the GPL or another open source license.

Sections 2(b) and 2(c) mirror equivalent provisions in the GPL.

> b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
>
> c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

Section 2(d) adds specific limitations on licensed libraries' use of tables or other functionality provided by the program with which the library is intended to function.

> d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.
>
> (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

This maximizes the utility (and the value to other open source and free software developers) of the library by encouraging them to be as portable as possible. The closer a given library comes to standing alone, the easier it is to conform it to function with an application other than the one for which it was originally written.

The last three paragraphs of Section 2 are substantially identical to the parallel provisions in the GPL.

> These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, 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 Library, 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 Library.
>
> In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

Section 3 of the LGPL addresses a change in licensing from the LGPL to the GPL.

> 3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

This part of the section apparently addresses the bar inherent in the LGPL or creating derivative works that are not libraries from an LGPL-licensed work. This provision is interesting in that it permits any licensee to "upgrade" the license to the GPL license. It operates as a savings clause, in that it would provide an escape in the event that any interpretation of the LGPL or the GPL prevented a program licensed under one from being used with a program licensed under the other.

This change in the license applicable to a given copy of a library is a one-way street. Once a program is re-licensed as a GPL program, it cannot go back to licensing under the LGPL.

> Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

Of course, as other copies of the library would still be available licensed under the LGPL, this sentence really addresses derivative works.

> This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

This sentence is slightly misleading. Re-licensing a program under the GPL is not just "useful" but necessary if the derivative work is not a library, as explained above.

Section 4 substantially parallels similar provisions of the GPL with regard to providing the source code with the binary code.

> 4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you 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.

As is apparent from the following provisions of the LGPL, the distribution of source code of a library standing alone is more restricted under the LGPL than under the GPL: when the executable or binary code is distributed standing alone, it must be accompanied by the source code.

> If distribution of 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 satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

The LGPL, like the GPL, does permit the distribution of the source code by offering it on equivalent terms as the executable, such as on an FTP site, if the binary code is so offered.

Section 5 provides the critical definition of the "work that uses the Library." The LGPL was designed to permit open source code to function with code licensed under other models. This section serves that purpose by excluding from the terms of the license "work that uses the Library."

> 5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

"[I]n isolation" is the critical phrase of this paragraph, as the rest of the section makes clear.

> However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library".* The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

While the "work that uses the Library" remains free to be licensed as the creator wishes, when that work is linked with the Library, the resulting work is considered to be a derivative work (as defined by copyright law) and the LGPL imposes specific terms applicable to the distribution of that program plus library provided in Section 6. This compromise allows creators of "works that use the Library" to retain control over their own works, while imposing some limitation when those works are distributed together with the LGPL-licensed Library.

> When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.
>
> If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)
>
> Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

These paragraphs of the LGPL attempt, among other things, to distinguish between different uses of a given Library—what is a "work based on the Library" and what is a "work that uses the Library." These paragraphs attempt to draw a distinction that may be impossible to make, except on a case-by-case basis. The LGPL, however, seems to make three distinctions. First, if the putative "work that uses the Library" includes a header file that is part of the Library, it may well be a "work based on the Library" (and therefore be covered by LGPL), particularly if that work can be linked without the Library or if that work is itself a library. Second, if the putative "work that uses the Library" draws only to a limited extent on the Library, measured by reliance only on the specified categories of functionality—i.e., only "numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length)"—then it is deemed a "work that uses the

* This reference to small-l library should probably be to capital-L Library.

Library" (which falls outside the scope of the LGPL), even if it is otherwise a derivative work, as that term is used in copyright law. While the executable file incorporating the Library must be distributed under Section 6, the "work that uses the Library" itself may be licensed free of any limitation. Third, in a sentence probably included as a savings clause if a work is a "derivative work" of the Library, in the sense that it incorporates any code from the Library (as opposed to the object code "in isolation" described in the first paragraph of this section), it is subject to the distribution requirements of Section 6.

These distinctions are unclear and the impact of this section on creators of potential "work that uses the Library" may be hard to predict. Some interpretations of the LGPL distinguish between the dynamic (compiled together with the underlying program) and the static (not so compiled) linking of programs with libraries. Such distinctions are beyond the scope of this book. However, at least at the time of this writing, FSF-licensed libraries may not be dynamically linked, while libraries affiliated with Linus Torvalds and the Linux project may be. Because of the complexity of such problems, users facing these questions should contact the licensor of the Library in question.

So far, we have seen that the LGPL makes distinctions between essentially three different types of work:

1. The LGPL-licensed Library.
2. The "work that uses the Library."
3. The combined "work that uses the Library" and Library together, which I will refer to here as the "combined work," a term not used in the LGPL.

Putting to one side the problem of linking and the extent to which a "work that uses the Library" and the Library are truly distinct programs, the requirements of the LGPL are fairly clear. The "work that uses the Library," when distributed as a "standalone" may be licensed and distributed however the creator wishes, whether under the GPL, the BSD, a proprietary, or any other license. The "Library" must be distributed under the LGPL: the source code must be available under the same terms as the binary code and licensees of the Library must be given the same rights (and be bound by the same restrictions) as the licensor of the Library. Section 2 of the LGPL also states that when a "combined work" is distributed, it is also subject to distribution under the terms of the LGPL. These terms are spelled out in Section 6.

> 6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

This provision is on its face somewhat unclear. Does this mean that by distributing a combined work, the distributor must distribute the source code for or authorize modifications to the "work that uses the Library"? As is made clear by the following paragraphs, Section 6 requires no such thing.

> You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

After this paragraph follows provisions similar in purpose to those in Section 3 of the GPL. They are designed to give notice of the application of copyright to the Library and the fact that the Library is licensed under the LGPL. They also give licensees access to the source code of the Library and allow them to make modifications to it.

> a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

Accordingly, the distributor must distribute the source code to the Library (including any modifications made by the distributor), and the binary code of the "work that uses a Library" provided in such a way so that licensees can modify the Library and relink it to the "work that uses the Library."

> b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

This provision describes another option for distributing the combined work that may be more user-friendly.

The following provisions are substantially identical to those in Section 3 of the GPL:

> c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.
>
> d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.
>
> e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

Section 6(e) of the LPGL offers an option unique to the LGPL, which may be useful when the distibutor is distributing a modified version of the "work that uses the Library" to users who have already received the Library used as part of the combined work.

The form of the executable of the "work that uses the Library" is defined in the following paragraph of Section 6.

> For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be 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.

Accordingly, to distribute the combined work in compliance with Section 6, the distributor must include data and utility programs (and any other components) that are necessary to allow the combined work to function as originally intended with the Library, unless those components are already included with the operating system upon which the combined program is intended to run. If the combined work relies on libraries (or other programs) that are intrinsic either to the "work that uses the Library" or the Library itself (or to the operating system), the combined work cannot be distributed without violating the LGPL. This is made explicit in the following paragraph.

> It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

It may be that a distributor would like to distribute a work that consists of the "work that uses the Library" (which the distributor has the power to distribute), the Library, and another program, such as another library, which the distributor does not have the authority to distribute, but that users already own or may be able to purchase. Such a distribution is not permitted under the LGPL. If the distributor cannot distribute all the components of the combined work, the distributor cannot distribute any part of it. End users, of course, are free to combine the combined work with libraries (or other programs) that they may otherwise have access to, as such combinations are outside the scope of the LGPL. However, they may not copy, distribute, or modify such works.

Section 7 addresses the situation in which a distributor has created a work based on the Library and has placed it side by side with another library under a proprietary license (or license other than the LGPL that permits the distributor to distribute it) to make it into what is in effect a single library. The distributor may do so without nullifying any license provisions applicable to the other library, subject to certain conditions.

> 7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:
>
> > a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

> b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

The standalone executable form of the Library must be distributed along with the combined library, subject to the terms otherwise applicable under the LGPL (i.e., with the source code accompanying the Library) and prominent notice must be given as to where the uncombined form of the Library may be found (and presumably accompanied by its source code). This is somewhat confusing because the uncombined form of the work based on the Library must be part of the package. Presumably, identifying the filename would be sufficient.

Section 8 of the LGPL operates much like Section 4 of the GPL.

> 8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library 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.

Section 9 of the LGPL likewise corresponds to Section 5 of the GPL.

> 9. 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 Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

The remaining sections of the LGPL, 10 through 16, are substantially identical to Sections 6 through 12 of the GPL. They are included here for completeness.

> 10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library 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 with this License.
>
> 11. 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 Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library 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 Library.
>
> 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.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library 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.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser 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 Library 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 Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, 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

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "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 LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. 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 LIBRARY 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 LIBRARY (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 LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
>
> END OF TERMS AND CONDITIONS

Following the end of the LGPL's terms and conditions are instructions on how to implement the LGPL. Again, these mirror the instructions in the GPL.

> How to Apply These Terms to Your New Libraries
>
> If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).
>
> To apply these terms, attach the following notices to the library. 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 library's name and an idea of what it does.
> > Copyright (C) year name of author
> >
> > This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.
> >
> > This library 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 Lesser General Public License for more details.
> >
> > You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
>
> Also add information on how to contact you by electronic and paper mail.
>
> You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:
>
> > Yoyodyne, Inc., hereby disclaims all copyright interest in the library 'Frob' (a library for tweaking knobs) written by James Random Hacker.
> >
> > signature of Ty Coon, 1 April 1990
> > Ty Coon, President of Vice
>
> That's all there is to it!

The Mozilla Public License 1.1 (MPL 1.1)

In January, 1998, Netscape Communications decided to release the binary code of its Communicator web-brower for free. Less than 24 hours later, it decided to release the Communicator source code as well. As a result, at the same time that Netscape

was addressing the many technical problems with transitioning Communicator into open source (including removing substantial amounts of code written by third parties who were unwilling to have their code "open sourced"), Netscape had to address the complex licensing issues involved.*

The Netscape Public License (NPL) and the Mozilla Public License (MPL) were the result of these efforts.† The NPL was substantially similar to the MPL, but it reserved certain rights to Netscape, most importantly, the right on the part of Netscape to relicense code developed by third parties that is derived from Communicator code under a proprietary or other license. Third-party modifiers of NPL-licensed code could thus lose any benefits that might flow from their contributions, without the guarantee, as for instance under the GPL, that their code will remain available to the community of programmers. The MPL does not contain the particular provisions embodying this grant of rights to Netscape.

The MPL constitutes an interesting hybrid of the ideas of the GPL and the BSD licenses already described. While code that falls within the scope of what the license describes as "Covered Code" is subject to many of the restrictions present in the GPL, such as the requirement that it be made available in open source form, the MPL, through its Section 3.7, also permits the use of such "Covered Code" in "Larger Works," meaning that MPL-licensed code can be combined with code licensed under another license. This latter result is expressly prohibited by the GPL and permitted by the BSD License. The MPL establishes something of a middle ground between the two licenses.

Thanks to its heritage as the product of a large American corporation, the MPL reads much more like a standard corporate contract, beginning with a long list of definitions, before going into another long list of numbered paragraphs and sub-paragraphs. Section 1 of the MPL consists entirely of definitions.

> 1. Definitions.
>
> 1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

Commercial Use is defined in a somewhat counterintuitive way. As defined, it includes any form of distribution, whether in exchange for payment or not.

> 1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

* For a full description of the considerations that went into the decision to release Navigator as an open source project and the development of the Netscape Public License and the Mozilla Public License, see *Freeing the Source: The Story of Mozilla* in *Open Sources: Voices From the Open Source Revolution*, p. 197 and following. (O'Reilly, 1999).

† The name "Mozilla" is derived from the name for the Navigator code used at Netscape—a combination of "Mosaic," an early web browser, and Godzilla.

The explicit definition of the term "Contributor," and the use of that term throughout the MPL 1.1, distinguishes this license from the others we have previously examined. In the BSD and GPL, for example, no distinction is made between the "Contributor" and "You" the licensee: it is presumed that those persons are one and the same. Such a "Contributor" is also distinguished from the "Initial Developer," which is identified in Appendix A to the license. Obviously, with regard to the Mozilla project itself, the Initial Developer was Netscape.

This idea of "Contributors" to the code reflects the centralized notion behind the MPL and the Mozilla project that it was intended to license. Although it was certainly not mandated by the license itself, the MPL reflects a development model under which "Contributors" would be supplying their work to a continuing project, not one under which licensees would be free to appropriate the code to their own uses, subject to certain restrictions on their distribution of the code.

> 1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

This provision is a natural outgrowth of the use of the idea of Contributor, as that term is used in the license. This has important effects as the Contributor retains a number of important rights over his contribution, as described in more detail later in the license.

> 1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

This is one of the fundamental terms in the license: Covered Code, Modifications, and Original Code are the three works that are governed by the terms of the license.

> 1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

This idea is one that has been described in previous licenses, for example, the GPL's requirement that source code be made available through FTP sites or comparable methods.

> 1.5. "Executable" means Covered Code in any form other than Source Code.

Again, this definition of Executable is somewhat different than the form of binary or executable that has been described previously. Rather than define Executable in terms of its function—i.e., as a program that does work—the MPL defines it by what it is not, as not-Source Code, which is defined later. This provides a broader definition than would be provided by a function driven definition.

> 1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

This term has already been addressed, as contrasted to Contributor.

> 1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

Unlike the GPL, as will be seen shortly, the MPL permits the combination of work governed by the license—i.e. the Covered Code—with code not governed by the license as part of a Larger Work. This is subject to certain restrictions relating to the making available of the Covered Code; but if those restrictions are satisfied, a Larger Work may otherwise be distributed as the licensee sees fit, including under a proprietary or other license.

> 1.8. "License" means this document.

This term is self-explanatory.

> 1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

This term is probably not strictly necessary, as the term is used in the license in a manner that does not vary from that in common usage. It does preserve, with the clause "whether at the time of the initial grant or subsequently acquired," the licensee from liability if the Initial Developer or Contributor lacks the right to license certain pieces of intellectual property, say, a patent, at the time the licensee exercises rights under the license, but subsequently acquires such a right. One would hope that such situations would be fairly rare, as the licensee (as well as the infringing Initial Developer or Contributor) could probably be held liable for infringement during that interim period.

> 1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:
>
> > A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.
> >
> > B. Any new file that contains any part of the Original Code or previous Modifications.

This term is also largely self-explanatory, including any works made by modifying the Original Code, which is defined immediately below as that code contributed by the Initial Developer, whether by the licensee (the "You" that is also defined below) or by a Contributor. The "Modifications" definition is important because it marks the extent of Covered Code as distinct from a possible larger work, as described in Section 3.7. The decision about what code constitutes Original Code or a Modification is made on a file-by-file basis.

> 1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

The Original Code, as would be expected, is that source code providing the foundation for the license.

> 1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

As described in later clauses, both the Initial Developer and any Contributor grant royalty free licenses to any licensee of the MPL for patents held by them, which are related to the MPL software. The use of the term Patent Claims is a means to grant the broadest rights possible in such patents.

> 1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or dearchiving software is widely available for no charge.

As with many of the terms given specific definitions under the MPL, the term Source Code is somewhat different than the term "source code" discussed in connection with previous licenses. There are two principal distinctions. First, Source Code can mean one of two things: either, "the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable," which is substantially identical to the use of source code that has been discussed previously; *or* "source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice," i.e., only that part of the source code that is different from the source code in the Original Code or another well-known version of the Covered Code, the source code of which is presumably available. Using this second option may make distribution of Source Code under the MPL logistically simpler and use less bandwidth.

Second, and to the same end of easing distribution, the MPL permits distribution of the Soure Code, defined either way, in the form of compressed or archived files, so long as the file can be decompressed using widely available free (free as in no-charge) software.

> 1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1.

The first part of this definition is similar to the use of "You" as licensee in the GPL and other licenses. One variation is that compliance with the terms of the license is expressly made a condition of the exercise of the rights of the license in the definition of itself.

> For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

This second part of the definition is present to include within the scope of the restrictions of the license, with regard to "legal entities," i.e., corporations, partnerships, limited liability companies or other artificial persons recognized by the law, parents, subsidiaries, and sister corporations of the licensee, and is included in similar form in other licenses, as already noted. Any exercise of rights under the license by modification or distribution of the Covered Code or the like would almost certainly bind the related entity to the terms of the license directly, without need for recourse to this provision. Moreover, to the extent such a related entity had not itself exercised any rights under the license, it could argue, perhaps successfully, that it was not bound by the agreement of the licensee.

Nonetheless, this part of the license does narrow the obligation to distribute source code, as required by Section 3.2, at least within the organization. If related entities were deemed to be distinct, they would each have obligation to distribute source code along with executable versions of the code to each other—a result that might be unnecessarily cumbersome, and which is avoided by this definition.

Section 2 of the MPL embodies the first of the two licenses contained in the MPL. As already noted, the MPL distinguishes between the Initial Developer and subsequent Contributors to the program. This distinction is embodied in the two different grants of rights in the MPL, the first of which is the grant of rights by the Initial Developer.

> 2. Source Code License.
>
> 2.1. The Initial Developer Grant.
>
> The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:
>
> (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and
>
> (b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

This grant distinguishes between the category of rights applicable to distribution, modifications, and sublicenses of the Original Code when included as part of a Larger Work, as that term is defined earlier or with modifications, and the Original Code when distributed not as part of such a Larger Work. Simply put, the MPL grants a license only of rights excluding patent and trademark rights, i.e., only those rights arising under copyright, when the Original Code is distributed as part of a Larger Work *or* when the Original Code is distributed with Modifications. Rights under any patent applicable to the Original Code are only granted to the distribution (including by sale) of the Original Code (or portions thereof), which stand alone.

This is an important distinction, and any person intending to distribute the Original Code with Modifications or as part of a Larger Work should be wary of it. In the

event that such patent or trademark rights do apply to the Original Code, the user should contact the Initial Developer to see if such rights could be obtained separate from the license before proceeding.* In addition, no trademark rights are granted, even for distribution of the unmodified Original Code.†

> (c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

This provision is largely self-explanatory and is not likely to have much practical effect. If the Original Code has not yet been released under the License, then the Initial Developer retains all intellectual property rights over it and any subsequent user is limited to that very narrow set of rights described in Chapter 1, such as the "fair use" of copyright material. The License would simply not come into effect at all.

> (d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

This section acts largely as a savings clause to the limitation of patent rights provided in Section 2.1(b). Section 2.1(d)(1) and (2) provide that any patent rights granted by the license are limited to their application in the Original Code. Accordingly, a licensee may not use that patent in another piece of code. Section 2.1(d)(3) appears to reiterate what is already stated by the distinction between Section 2.1(a) and (b): the MPL does not license patent rights for modified versions of the Original Code or Larger Works incorporating the Original Code.

Section 2.2 parallels Section 2.1 but governs contributions to the Covered Code made by Contributors.

> 2.2. Contributor Grant.
>
> Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

This echoes Section 2.1, applying similar terms to grants from Contributors as to the grant from the Initial Developer.

> (a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

* Mitchell Baker, the original drafter of this license, says that "It's possible [this] interpretation is correct, but this was not the intent. The intent was that the patent grant would be for the Original Code whether or not combined with other code, but not for changes to the Original Code." Baker's comment applies to both Section 2.1 and 2.2 of the MPL.

† Assuming that such a user is not seeking to associate such a derivative work with the name or trade dress of the Initial Developer, it is hard to see how a right arising under trademark would effect the use of a piece of functional code. Nonetheless, this is the type of situation in which consultation with an experienced attorney is not only advisable but necessary before proceeding.

Like Section 2.1(a), Section 2.2(a) makes an important distinction between the licensing of patent and trademark rights and the licensing of "other" rights, i.e., rights under copyright. Only the latter rights are granted when the the Contributor's code is used with Modifications or as part of a Larger Work.

> (b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

This provision works much like the equivalent provision governing the Original Code. The one addition is that the patent rights are granted both for the Original Code plus the Contributor's modifications, the "Contributor Version," and for the Modifications made by that Contributor standing alone.

> (c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

This provision seems to make fairly restrictive the conditions under which the Contributor Version falls within the scope of the license, i.e., not until "Commercial Use." However, as we have already seen in the definitions used in the MPL, Commercial Use simply means any distribution to a third party. Emailing a copy of the Contributor Version to a friend make the license effective on the code, with all that entails. Again, this is something contributors (or potential contributors) need to be wary of.

This distinction becomes important in light of the following section:

> (d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version;
>
> 2) separate from the Contributor Version;

As with the Original Code, patent rights are granted for use only in the Contributor Version, a point already made clear by Section 2.2(b). The last two numbered subparts of this section state what is essentially a legal truism: the Contributor does not (and, legally, cannot) grant patent rights that he does not hold.

Subpart 3 states that the Contributor does not grant patent licenses for patents infringed by modifications to the Contributor Version by a third party.

> 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices;

Subpart 4 states that the Contributor does not purport to license parts of the Covered Code (either from the Original Code or modifications made by another contributor) that infringe patent rights that are not the Contributor's own work.

> or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

These last two subparts of Section 2.2(d) may seem meaningless because they merely state a legal truism: a person cannot license that which he does not own. However, they are meaningful in that they protect the Contributor from legal liability in that they make clear that the Contributor is not *purporting* to license that which he lacks a right to license.

Much of this license's provisions regarding the fine points of software patents will likely be of little or no importance to most contributors to open source projects: if the contributor's intent is to meaningfully contribute to an open source project without (substantial at least) payment in return, the costs and difficulty of applying for and defending a patent may not be justified.

Section 3 of the MPL imposes the generational limitations of the license, which parallel fairly closely those of the GPL.

> 3. Distribution Obligations.
>
> 3.1. Application of License.
>
> The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2.

The modifications made by any licensee to the Covered Code must be licensed by the terms applicable to Contributors, as provided for by Section 2.2. Section 3.1 continues:

> The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

The Source Code, the distribution of which on the same terms as the Executable Code is made mandatory by Section 3.2, can only be distributed under this or a future version of the MPL and the License must be distributed with it. "Additional rights," however, may be granted as provided below.

> 3.2. Availability of Source Code.
>
> Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

Section 3.2 makes mandatory the distribution of the Source Code on terms no less favorable than that of the Executable, much like the GPL.

Like the BSD License, the MPL also requires certain attributions of credit for developing the software, albeit only for the Initial Developer and not for any subsequent Contributor.

> 3.3. Description of Modifications.
>
> You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

This section also requires a "comment" document describing the type and date of any Modifications to the Covered Code. The practical importance of such a requirement is clear. If Contributors scrupulously adhere to this, the decisions by future users (and Contributors) as to which version of a licensed distribution they want to use should be made signficantly easier.

Section 3.4 reflects the MPL's concerns with patent laws and patent infringements.

> 3.4. Intellectual Property Matters
>
> (a) Third Party Claims.
>
> If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

This section says, in so many, words that to the extent a Contributor is aware of third-party patent claims to the Contributor Version, reasonable efforts—i.e., the inclusion of the "LEGAL" file—should be taken to alert future users or contributors that they must secure the appropriate rights from that third party prior to using, distributing, or modifying the Contributor Version. The legal effect of this provision, in terms of protecting the Contributor from claims of infringement, is questionable at best. A holder of a third-party patent certainly could consent to such an arrangement and reach independently negotiated licenses of his patent rights with each of the users or potential contributors that would be interested in licensing his patent. But such a patent holder could probably just as easily object and sue the Contributor for patent infringement for distributing the (admittedly) infringing code, "LEGAL" file or not. Accordingly, to the extent that any Contributor would want to take advantage of this mechanism, it is imperative that such a Contributor reach an understanding with the third-party patent holder before proceeding.

Section 3.4(b) states that the same model (with all the same defects and potential for liability) governs the Contributor Version's use of application programming interfaces. This was included to address issues raised by standards created by participants who later disclose that they have patents on basic mechanisms required to work with those standards.

> (b) Contributor APIs.
>
> If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

Finally, Section 3.4(c) contains representations implicitly made in Section 2.2, to the effect that the Contributor represents that he believes he has the rights to grant the license he licenses as part of the Contributor Version.

> (c) Representations.
>
> Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

As with many provisions of this license, its legal effects are unclear at best. To the extent the Contributor Version infringes on a third-party patent, the holder of that patent can still sue users, modifiers, or distributors of that version for infringement, representation or not. This subsections is designed to avoid that situation by ensuring that Contributors don't knowingly add patent-infringing code to the project.

Section 3.5 requires that certain notices be attached and provided with the Covered Code with the intention of giving future users of that code notice of the provisions governing it, through inclusion of a notice (attached as Exhibit A) and a copy of the MPL.

> 3.5. Required Notices.
>
> You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code.

The second part of Section 3.5, like similar provisions in the GPL, explicitly permits licensees to offer and to charge fees for warranty and support agreements in connection with the Covered Code. Such permission is contingent upon making clear that the licensee is the only person undertaking any such obligation, not the Initial Developer or any Contributor.

> You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

Section 3.6 governs the terms of the distribution of the executable version of Covered Code. It has two principal effects. First, it conditions any distribution of the executable version on compliance with Sections 3.1 through 3.5 already described. Second, it permits distribution of the executable under a different license than the source code, including under a proprietary license.

> 3.6. Distribution of Executable Versions.
>
> You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code.

The second part governs the terms of distribution of the Executable Code.

> You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

The ability of a licensee or Contributor to "cash in" on the distribution of the Executable Code under a proprietary license is limited by the ability of any licensee to access and compile the source code for himself. Distribution of the executable under different terms also requires indemnifying the Initial Developer and Contributors for any liability that might be incurred—although it is hard to see what, if any, additional liability could possibly accrue to them that would not accrue from the distribution of the source code.

> 3.7. Larger Works.
>
> You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

This provision works hand in hand with the second part of Section 3.6. The Executable Version of the Covered Code can be distributed as part of a Larger Work with code licensed under proprietary or other licenses, subject to the limitations of the MPL, most importantly that the source code of the Covered Code be made available as required by Section 3.2.

This provision is the most important distinction between the MPL and the GPL, which, as already discussed, does not permit integration with non-GPL licensed work, and the LGPL, which permits such integration only on fairly restrictive terms. This is an elegant solution to the problem in its simplicity but is subject to a couple of caveats. First, the other code that is combined with the Covered Code to make the Larger Work must itself be susceptible to such combined distribution: for example, such Larger Work cannot include any GPL-licensed code. It does, however, permit combination with proprietary code, at least to the extent the distributor has the right to distribute such code. Second, and perhaps more importantly, Section 3.7 may provide a substantial incentive for putative or potential Contributors to implement their work where possible as "other code" as opposed to Modifications, which become part of the Covered Code as Contributions upon the first Commercial Distribution of that code, as already described. By adding utility to the Covered Code without falling within its restrictions, such "other code" gives licensees the possibility to profit from the contributions (or Contributions) of others, by selling or otherwise distributing the Larger Work, without sharing the benefits of their own code with that community.

Section 3.7 is the key provision of the license and the permission it gives to Contributors and distributors of Covered Code to incorporate that code into larger works may be seen as a real advantage over the GPL. It avoids the GPL's strict limitations on combining GPL-licensed code with code under other licenses, and it also avoids some of the uncertainties and complexities associated with the LGPL.

The MPL also takes a different approach than the GPL to the situations when statutes, regulations, or judicial decisions invalidate or make impossible the enforcement of terms of the license. As already noted, the GPL and the LGPL forbid exercise of the rights under the license in that situation. The MPL does not.

> 4. Inability to Comply Due to Statute or Regulation.
>
> If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

The licensee simply must comply with the License to the extent possible and notify other licensees of limitations on the license that result from statute, regulation, or

judicial decisions. It is still possible for software to be distributed under the MPL after a hypothetical judicial decision prohibits distribution of some but not all the source code of a given Contributor Version or requires payment of royalties to one or more Contributors but not to others.

> 5. Application of this License.
>
> This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

This provision is self-explanatory. Section 6 substantially provides that Netscape has the right to revise and update the MPL (although they have not done so since issuing MPL 1.1), and have those modified terms govern all code licensed under the MPL. This, obviously, could substantially change the rights or standing of licensees and Contributors to the MPL.

> 6. Versions of the License.
>
> 6.1. New Versions.
>
> Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.
>
> 6.2. Effect of New Versions.
>
> Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

Persons exercising rights under the license, such as users of Covered Code, will not lose their rights—they will still be able to use them as defined by previous versions of the license. However, Contributors to the Covered Code might find that the rights they are required to grant have substantially changed or expanded by the new license. Netscape transferred the right to modify the MPL to the Mozilla Foundation when it was founded in July, 2003, and Mitchell Baker reports that there will be a Version 1.2 of the MPL reflecting this at some point.

Developers who are interested in using the MPL, but not in connection with the Mozilla project, are permitted by Section 6.3 to use their own version of the MPL, free of any third party's ability to change its terms.

> 6.3. Derivative Works.
>
> If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

Such a license must not be called the MPL and must otherwise be distinct from the MPL and the related NPL. This provision again contrasts with the GPL and the LGPL, which explicitly prohibit the creation of derivative licenses.

Section 7 provides the now-familiar disclaimer of warranties.

> 7. DISCLAIMER OF WARRANTY.
>
> COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

Section 7 includes among its disclaimers, in addition to other things, that the Covered Code is "non-infringing." This disclaims all warranties regarding the effect of Sections 2.1, 2.2, and 3.4(c) to the extent that they may be read as representations that the Initial Developer or a Contributor has the right to license a given piece of code under copyright, patent, or trademark law. While the licensee can feel some assurance that he will not be sued by the Initial Developer or Contributor for infringement, provided of course that he complies with the terms of the MPL, he is on his own with regards to third-party intellectual property claims. By using the Covered Code, he undertakes the risk that he may be sued for infringement, without any recourse to the Initial Developer or the Contributors.* (As noted in Chapter 1, this is true of other open source licenses as well, and it is not unique to the MPL.)

Section 8 of the MPL governs the termination of the license. Like much of the rest of MPL, it is similar to, but more forgiving than, parallel provisions of the GPL.

> 8. TERMINATION.
>
> 8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

This is more permissive than Section 4 of the GPL, which voids all rights under the license upon its infringement. The MPL, by contrast, provides a 30-day "cure" period following the discovery of such a breach for the licensee to cure. In addition, like the GPL, the breach of the MPL by a distributor does not void sublicenses granted by that distributor to distributees.

* To the extent that the Covered Code infringes on third-party intellectual property rights, either the Initial Developer and/or the Contributors are in the same position as the licensee and would have similar liability.

The MPL provides that the license terminates as a consequence of patent litigation brought by a putative licensee with, however, some important limitations. It should be noted that these termination provisions apply only to *patent* claims and not claims alleging infringement of other forms of intellectual property, such as copyright and trademark.

> 8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:
>
> > (a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2. 1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

If it occurs, the termination of rights under the license is prospective only, i.e., only bars future use of the licensed code, and does not create liability for past uses of the licensed code. The termination is also subject to a 60-day "cooling off period" in which the person alleging infringement can either withdraw the claim or negotiate another resolution with the person against whom the claim is brought, whether the Initial Developer or a Contributor.

Curiously enough, the MPL provides for more punitive termination provisions if the patent infringement alleged against the Initial Developer or a Contributor does not relate to Covered Code but to some other action of such persons.

> > (b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

Such termination has no "cooling off period" and is, moreover, retroactive. The revocation is "backdated" from the first use of the code under the license by the person suing. The threat of enforcement of this provision creates a strong disincentive for the filing of such patent litigations.

> 8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

This provision has no binding effect and can really be read as exhortatory only. Such a pre-litigation termination would presumably be reached through settlement, which could be done on any terms agreed to by the parties involved, taking into consideration such "reasonable value of the licenses" granted by the MPL.

> 8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

This duplicates the effect of Section 8.1.

The following section disclaims liability to the extent permitted by law, like many of the other open source licenses already examined.

> 9. LIMITATION OF LIABILITY.
>
> UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

Except as provided by separate agreement to warranty or otherwise indemnify against loss, as permitted under Section 3.5, any software provided under the MPL is provided "as is," with the user taking responsibility for its use, except to the extent such a disclaimer is prohibited by law.

> 10. U.S. GOVERNMENT END USERS.
>
> The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

These provisions ensure that United States government agencies may be bound by commercial software licensing agreements.

Section 11 contains a bundle of provisions typical in a commercial contract.

> 11. MISCELLANEOUS.
>
> This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be

> reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

This section provides that California law governs interpretation of the contract and provides that the venue for all disputes—in which one of the participants (i.e., either the plaintiff or the defendant) is a citizen or an entity registered to do business in the United States—shall be the federal district court in Santa Clara, California (not coincidentally a venue that was convenient for Netscape when the contract was written, before their acquisition by AOL). It also explicitly provides for shifting of attorneys fees and costs, meaning that whoever loses the lawsuit (if there is a loser) is responsible for paying all the costs associated with the lawsuit, including the other side's "reasonable" attorneys fees.

The next section of the license addresses the situation in which a legal claim is made against the Initial Developer and one or more Contributors and attempts to impose a responsibility on both to jointly address such a claim.

> 12. RESPONSIBILITY FOR CLAIMS.
>
> As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

It is not clear what legal effect this section has, if any. It is certainly possible that both the Initial Developer and a Contributor could be held jointly and severally liable (meaning that each is fully responsible for the violation of the other), say, if the Covered Code was found to infringe on a patent or copyright held by a third party, and both the Initial Developer and the Contributor had distributed the Covered Code. Courts, however, are typically reluctant to enforce such relatively vague obligations. Agreeing to "work with" Initial Developer is exactly such a vague obligation: it may have an exhortative effect, but that is likely to be it.

Section 13 describes a legal arrangement for multiple licensing of the Covered Code.

> 13. MULTIPLE-LICENSED CODE.
>
> Initial Developer may designate portions of the Covered Code as Multiple-Licensed. Multiple-Licensed means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

With the MPL, as with any other license, the creator of the work (in this case, the Initial Developer) may issue the work under one or more licenses, and potential licensees are free to choose, *at the beginning*, the license regime under which they will use the work. Once that choice is made regarding the licensee's own contributions to the work or modifications, the licensee is bound to the terms of the license that was chosen and cannot "go back." For example, the Covered Code may be licensed by the Initial Developer both under the MPL and the GPL. A Contributor chooses to work with the Covered Code under the GPL, makes modifications, and distributes those modifications. Having done so, the Contributor then discovers that his Contribution includes a patentable process, which he would much rather license under the more protective MPL. Now, he cannot do this. Having distributed the code under the GPL, with all the terms applicable thereto, the genie cannot be put back in the bottle. The Contributor is, of course, not required to continue to maintain or develop the Covered Code. But he would be prevented from enforcing patent claims to subsequent users or modifiers of that program who adhere to the terms of the GPL. This example is not unique to the MPL, but rather arises under any situation in which a piece of code is multiple-licensed.

Section 13 is the last section of the license. A model "fill-in-the-blanks" Exhibit A follows, for those who choose to become Initial Developers of their programs under the MPL.

> EXHIBIT A -Mozilla Public License.
>
> "The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at
>
> http://www.mozilla.org/MPL/
>
> Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specificlanguage governing rights and limitations under the License.
>
> The Original Code is _____.
>
> The Initial Developer of the Original Code is _____.
>
> Portions created by _____ are Copyright (C) _____ _____. All Rights Reserved.
>
> Contributor(s): _____.
>
> Alternatively, the contents of this file may be used under the terms of the _____ license (the [___] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [___] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [___] License."

> [NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

The MPL provides a novel solution to the problems faced by Netscape in bringing into open source an already well-established set of code and setting up terms and conditions that would both protect its rights and encourge contributors to modify and improve that work. Its terms and focus reflect its origins: its distinction between the Initial Developer and subsequent Contributors sets it apart from, say, the more freeform development contemplated by the GPL. Its emphasis on patent rights and the limited grant of them provided by the license also reflects its corporate origin and the intent on the part of Netscape to limit, to the extent possible, the grant of rights while still remaining consistent with an open source model. By doing so, the MPL attempts to ensure that both open source volunteers and commercial developers are comfortable cooperating in this legal environment.

Application and Philosophy

The GPL and MPL both have had symbolic as well as practical impacts in the world of software development. Those effects of these licenses, beyond their strict terms, are described next.

The GPL License and the Free Software Philosophy

The impact of the GPL, and its offshoot, the LGPL, on the development of software cannot be overstated. The GPL project that grew up with the license, the GNU/Linux constellation of applications, better known simply as Linux,* has seen its acceptance by users grow steadily from the early 1990s to the point where it now poses the only significant competitor to the Windows operating system.

This success, depending on your point of view, arises either because of, or despite, the fact that the GPL bars any development of software from GPL-licensed software that is not itself GPL licensed. The GPL seems to embody the maxim that "Freedom in a commons brings ruin to all."† By requiring that all contributions to GPL projects be themselves GPL licensed, the GPL ensures not only that these contributions are available to other programmers (or at least those programmers willing to work within the GPL framework) but also encourages contributions from those programmers to whom it is important that their contributions be made, and remain, "free," as that term is used in the GPL.

* For a discussion of this nomenclature, see the essay "What's in a Name?" contained in Richard M. Stallman, *Free Software: Free Society* (Free Software Foundation, 2002) (p. 51 and following).

† Quotation from Garett Hardin is taken from William Vollman, *Rising Up and Rising Down* (McSweeneys 2003) (Vol. III, p. 219). "Freedom," as the term is used in the quotation, is the absence of rules: the GPL itself is an embodiment of the principle that certain types of freedom require rules in order to be preserved.

The existence of such programmers is by now beyond dispute. Based on the success of the GNU/Linux project alone, the free software project has succeeded. Part of this success is due to the fact that the GPL has as important a symbolic purpose as a practical one. The restrictions of the GPL (and the LGPL) have greatly shaped the nature of development of software that is GPL-licensed. Programmers, by and large, respect the GPL and conform their behavior accordingly. The development of projects under the GPL depends on each participant adhering to the terms of the license by making his or her own contribution available to the community of developers. This adherence has resulted in the great success enjoyed by GPL-licensed projects.

But the GPL has an equally important aspirational purpose. Given that the GPL is often viewed as the "purest" form of licensing in nurturing and encouraging open development of software, development under this license has drawn programmers who take seriously the larger concept of open software development. This has had results beyond those caused by the terms of the license itself.

Contrary to the beliefs of some, the GPL does not require that software running on a GPL-licensed operating system be licensed under the GPL. Similarly, the GPL does not require that only GPL-licensed programs be distributed as part of a distribution containing GPL-licensed code. For example, many of the GNU/Linux distributions, including those of Caldera, a significant early GNU/Linux distributor, included both the GPL-licensed Linux operating system and proprietary licensed code. Caldera paid royalties (as required by the terms of the proprietary software) on proprietary software that was distributed on the same CD-ROM as the operating system. Because the proprietary software did not compile with the operating system, this was perfectly consistent with the terms of the GPL. Purchasers of the Caldera distribution were free to install the GNU/Linux operating system and to install the proprietary software from the same CD-ROM on the same computer.

Despite the fact that such distributions are completely consistent with the GPL, some programmers and distributors have disfavored such distributions, on the grounds that such distribution is inconsistent with the spirit of free software development. A well-known distributor of GNU/Linux software, the Debian project, for example, has taken the position that every piece of software distributed as part of a GNU/Linux distribution should be licensed under a free software license.* This view, held by many members of the free software community, has significantly influenced the development of software under the GPL.

Taking just one example, a substantial movement arose to counter the K Desktop Environment (or KDE), a graphic user interface (GUI) frontend on the GNU/Linux operating system, which some perceived as an encroachment on the free software philosophy.† Beginning in 1996, a programmer named Mattias Ettrich started to

* See *http://www.debian.org/social_contract* for more details.

† For a more thorough description of this episode, see Chapter 15, "Trolls Versus Gnomes" in *rebel code: inside linux and the open source revolution*, Glyn Moody (Perseus Publishing, 2001).

develop KDE based on the Qt Toolkit, a non-GPL licensed program written and owned by Trolltech.* While the Qt Toolkit was available without cost, and while its source code did eventually become available,† there was a strong counterreaction to the development of such a potentially critical piece of software under a non-GPL license. As a result, a separate team of programmers began to develop the GNU Network Object Model Environment or GNOME, announced in August, 1997. As a result, both the KDE and the GNOME programs, both well-supported by developers and applications, are growing and thriving today.

Conversely, the development of free software projects is not determined, or even necessarily shaped, by the terms of the GPL. While the GPL encourages a certain type of development, it does not mandate any particular type of development structure. Indeed, it invites many different approaches to development. As described in Chapter 7, the initial components of the GNU/Linux operating system—the GNU C Compiler and Emacs, among others—were developed under a very different model of software development than the Linux kernel. Moreover, as also described in Chapter 7, even for projects licensed under non-GPL licenses, there are significant advantages to maintaining an "open development" model in which code is kept available to the open source community and not developed (as is permitted) under proprietary licenses.

While its terms may provide the foundation for free software development, the GPL is also a potent symbol of a much larger, and more important, idea of how software (or any other work) should be made and maintained. The success of this license has been driven as much by the ideals that it represents as by strict application of its legal terms.

The Mozilla Public License: Circumstances and Opportunities

As already described, the MPL was the result of a decision by Netscape Communications—one of the first Internet companies—to open source license the code to its Netscape Communicator software in January, 1998. At that time, Netscape was in an intense competition with Microsoft, whose rival web browser, Internet Explorer, had the advantage of its close association with Microsoft's dominant Windows operating system. The MPL was an attempt to get some of the benefit of open source development into a program developed under a proprietary license.

The initial announcement of the "opening" of the Communicator code was greeted with great enthusiasm (and certainly boosted public perception of open source software), although Netscape's own economic condition—and its eventual absorption

* The Q Public License is discussed in Chapter 4.

† As noted in Chapter 4, Trolltech eventually cross-licensed the Qt Toolkit under the GPL after receiving substantial pressure to do so from the free software community.

into America Online—caused problems for the project. A thorough rewriting and a focus on standards-compliance have created a strong Mozilla culture, in many ways independent of its roots in Netscape. Open sourcing the project by itself didn't reverse Netscape's fortunes, but it has been a key source of continued innovation in the web browser market. The continued vitality of this project was demonstrated on June 27, 2004, with the release of the FireFox web browser by the Mozilla Foundation.

The MPL itself has thrived as an open source license. The well-constructed, well-written MPL has certainly found a niche: only the BSD, GPL, and LGPL Licenses are associated with substantially more projects than the MPL. The MPL has also been used as the base for a number of other Open Source Initiative-certified licenses, including the Apple Public License, the Nokia Open Source License, and the Sun Public License.

As can be seen from the examples of the GPL and the MPL, the success of licenses is a factor less of the terms or the wording of those licenses than of the ideas that they represent. Powerful, meaningful ideas draw minds, and the success of open source and free software licensing is the result of the minds that such ideas can draw.