

GNU General Public License

From Wikipedia, the free encyclopedia

The **GNU General Public License** (**GNU GPL** or **GPL**) is the most widely used^[5] free software license, which guarantees end users (individuals, organizations, companies) the freedoms to use, study, share (copy), and modify the software. Software that allows these rights is called free software and, if the software is copylefted, requires those rights to be retained. The GPL demands both. The license was originally written by Richard Stallman of the Free Software Foundation (FSF) for the GNU project.

In other words, the GPL grants the recipients of a computer program the rights of the Free Software Definition^[6] and uses copyleft to ensure the freedoms are preserved whenever the work is distributed, even when the work is changed or added to. The GPL is a copyleft license, which means that derived works can only be distributed under the same license terms. This is in distinction to permissive free software licenses, of which the BSD licenses and the MIT License are the standard examples. GPL was the first copyleft license for general use.

As of August 2007, the GPL accounted for nearly 65% of the 43,442 free software projects listed on Freecode,^[7] and as of January 2006, about 68% of the projects listed on SourceForge.net.^[8] Similarly, a 2001 survey of Red Hat Linux 7.1 found that 50% of the source code was licensed under the GPL^[9] and a 1997 survey of MetaLab, then the largest free software archive, showed that the GPL accounted for about half of the software licensed therein.^[10] Prominent free software programs licensed under the GPL include the Linux kernel and the GNU Compiler Collection (GCC). Some other free software programs (MySQL is a prominent example) are dual-licensed under multiple licenses, often with one of the licenses being the GPL.

David A. Wheeler argues that the copyleft provided by the GPL was crucial to the success of Linux-based systems, giving the programmers who contributed to the kernel the assurance that their work would benefit the whole world and remain free, rather than being exploited by software companies that would not have to give anything back to the community.^[11]

On 29 June 2007, the third version of the license (GNU GPLv3) was released to address some perceived problems with the second version (GNU GPLv2) that were discovered during its long-time usage. To keep the license up-to-date, the GPL license includes an optional "any later version" clause, allowing users to choose between the original terms or the terms in new versions as updated

GNU General Public License



Author	Richard Stallman
Latest version	3
Publisher	Free Software Foundation
Published	29 June 2007
DFSG compatible	Yes ^[1]
FSF approved	Yes ^[2]
OSI approved	Yes ^[3]
Copyleft	Yes ^{[2][4]}
Linking from code with a different license	No (except for linking GNU AGPLv3 with GNU GPLv3 – see section)
Website	gnu.org/licenses/gpl.html (https://gnu.org/licenses/gpl.html)

by the FSF. Developers can omit it when licensing their software; for instance the Linux kernel is licensed under GPLv2 without the "any later version" clause.^[12]

Contents

- 1 History
 - 1.1 Version 1
 - 1.2 Version 2
 - 1.3 Version 3
 - 1.4 Adoption
- 2 Terms and conditions
 - 2.1 Use of licensed software
 - 2.2 Copyleft
 - 2.3 License versus contract
- 3 Derivations
- 4 Linking and derived works
 - 4.1 Libraries
 - 4.2 Communicating and bundling with non-GPL programs
- 5 Legal status
- 6 Compatibility and multi-licensing
 - 6.1 Use for text and other media
- 7 Reception
 - 7.1 Legal barrier to app stores
 - 7.2 Microsoft
 - 7.3 Barrier to commercialization
 - 7.4 GPLv3 separates community further
- 8 See also
- 9 References
- 10 External links

History

The GPL was written by Richard Stallman in 1989 for use with programs released as part of the GNU project. The original GPL was based on a unification of similar licenses used for early versions of GNU Emacs, the GNU Debugger and the GNU C Compiler.^[13] These licenses contained similar provisions to the modern GPL, but were specific to each program, rendering them incompatible, despite being the same license.^[14] Stallman's goal was to produce one license that could be used for

any project, thus making it possible for many projects to share code.

The second version of the license, version 2, was released in 1991. Over the following 15 years, members of the FOSS (Free and Open Source Software) community became concerned over problems in the GPLv2 license which allowed GPL-licensed software to be exploited in ways that were contrary to the intentions of the license.^[15] These problems included tivoization (the inclusion of GPL-licensed software in hardware that will refuse to run modified versions of its software); incompatibility issues like with the Affero General Public License; and patent deals between Microsoft and distributors of free and open source software, which were viewed as an attempt to use patents as a weapon against the FOSS community.

Version 3 was developed to attempt to address these concerns and was officially released on 29 June 2007.^[16]

Version 1

Version 1 of the GNU GPL,^[17] released on 25 February 1989,^[18] prevented what were then the two main ways that software distributors restricted the freedoms that define free software. The first problem was that distributors may publish binary files only—executable, but not readable or modifiable by humans. To prevent this, GPLv1 said that any vendor distributing binaries must also make the human-readable source code available under the same licensing terms (Sections 3a and 3b of the license).

The second problem was the distributors might add additional restrictions, either by adding restrictions to the license, or by combining the software with other software which had other restrictions on its distribution. If this was done, then the union of the two sets of restrictions would apply to the combined work, thus unacceptable restrictions could be added. To prevent this, GPLv1 said that modified versions, as a whole, had to be distributed under the terms in GPLv1 (Sections 2b and 4 of the license). Therefore, software distributed under the terms of GPLv1 could be combined with software under more permissive terms, as this would not change the terms under which the whole could be distributed, but software distributed under GPLv1 could not be combined with software distributed under a more restrictive license, as this would conflict with the requirement that the whole be distributable under the terms of GPLv1.

Version 2

According to Richard Stallman, the major change in GPLv2 was the "Liberty or Death" clause, as he calls it — Section 7.^[14] This section says that if somebody has restrictions imposed that *prevent* him or her from distributing GPL-covered software in a way that respects other users' freedom (for example, if a legal ruling states that he or she can only distribute the software in binary form), he or she cannot distribute it at all. The hope is, that this will make it less tempting for companies to use patent threats to require a fee from the free software developers.

By 1990, it was becoming apparent that a less restrictive license would be strategically useful for the C library and for software libraries that essentially did the job of existing proprietary ones;^[19] when version 2 of the GPL (GPLv2) was released in June 1991, therefore, a second license — the Library General Public License — was introduced at the same time and numbered with version 2 to show that both were complementary. The version numbers diverged in 1999 when version 2.1 of the LGPL

was released, which renamed it the GNU Lesser General Public License to reflect its place in the philosophy.

Version 3

In late 2005, the Free Software Foundation (FSF) announced work on version 3 of the GPL (GPLv3). On 16 January 2006, the first "discussion draft" of GPLv3 was published, and the public consultation began. The public consultation was originally planned for nine to fifteen months but finally stretched to eighteen months with four drafts being published. The official GPLv3 was released by FSF on 29 June 2007. GPLv3 was written by Richard Stallman, with legal counsel from Eben Moglen and the Software Freedom Law Center.^[20]

According to Stallman, the most important changes are in relation to software patents, free software license compatibility, the definition of "source code", and hardware restrictions on software modification ("tivoization").^{[20][21]} Other changes relate to internationalization, how license violations are handled, and how additional permissions can be granted by the copyright holder.

It also adds a provision that "strips" DRM of its legal value, so people can break the DRM on GPL software without breaking laws like the DMCA.^[22]

The public consultation process was coordinated by the Free Software Foundation with assistance from Software Freedom Law Center, Free Software Foundation Europe,^[23] and other free software groups. Comments were collected from the public via the gplv3.fsf.org web portal.^[24] That portal runs purpose-written software called stet.

During the public consultation process, 962 comments were submitted for the first draft.^[25] By the end, a total of 2,636 comments had been submitted.^{[26][27][28]}

The third draft was released on 28 March 2007.^[29] This draft included language intended to prevent patent-related agreements like the controversial Microsoft-Novell patent agreement and restricts the anti-tivoization clauses to a legal definition of a "User" or "consumer product". It also explicitly removed the section on "Geographical Limitations", whose probable removal had been announced at the launch of the public consultation.

The fourth discussion draft,^[30] which was the last, was released on 31 May 2007. It introduced Apache License compatibility, clarified the role of outside contractors, and made an exception to avoid the perceived problems of a Microsoft–Novell style agreement, saying in Section 11 paragraph 6 that



Richard Stallman at the launch of the first draft of the GNU GPLv3. MIT, Cambridge, Massachusetts, USA. To his right is Columbia Law Professor Eben Moglen, chairman of the Software Freedom Law Center

You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license [...]

This aims to make future such deals ineffective. The license is also meant to cause Microsoft to extend the patent licenses it grants to Novell customers for the use of GPLv3 software to *all* users of that GPLv3 software; this is possible only if Microsoft is legally a "conveyor" of the GPLv3 software.^{[31][32]}

Also, early drafts of GPLv3 allowed licensors to add an Affero-like requirement which would have plugged the *ASP loophole in the GPL*.^{[33][34]} As there were concerns expressed about the administrative costs of checking code for this additional requirement, it was decided to keep the GPL and the Affero license separated.^[35]

Others, notably some high-profile developers of the Linux kernel, commented to the mass media and made public statements about their objections to parts of discussion drafts 1 and 2.^[36]

GPLv3 improves compatibility with several open source software licenses such as Apache License, version 2.0, and the GNU Affero General Public License which GPLv2 could not be combined with,^[37] but GPLv3 software can only be combined with GPLv2 software if the GPLv2 version has the optional "*or later*" clause of the GPL. "GPLv2 or any later version" is considered by FSF as the most common form of licensing GPLv2 software^[38] while for example Toybox developer Rob Landley described it as a *lifeboat clause*.^[39] Software projects which are licensed with the optional "*or later*" clause include the GNU Project, while a prominent example without the clause is the Linux kernel.^[12]

The final version of the license text was published on 29 June 2007.^[40]

Adoption

In August 2013, according to Black Duck Software, the website's data show that the GPL license family is used by 54% of open-source projects, with a breakdown of the individual licenses shown in the following table.^[41]

License	Usage Percentage
GPLv2	33%
GPLv3	12%
LGPL 2.1	6%
LGPL 3.0	3%

In a blog post by 451 Group analyst Matthew Aslett, Aslett argued that copyleft licenses went into decline and permissive licenses increased, based on statistics from Black Duck Software.^[42]

However, a later study showed that software licensed under the GPL license family has increased, and that even the data from Black Duck Software have shown a total increase of software projects licensed under GPL. The study used public information gathered from repositories of the Debian Project, and the study criticized Black Duck Software for not publishing their methodology used in collecting statistics.^[43] Daniel German, Professor in the Department of Computer Science at the University of Victoria in Canada, presented a talk in 2013 about the methodological challenges in determining which are the most widely used FOSS licenses, and showed how he could not replicate the result from Black Duck Software.^[44]

In 2011, four years after the release of the GPLv3, according to Black Duck Software data, 6.5% of all open-source license projects are GPLv3 while 42.5% are GPLv2.^[45] Google open-source programs office manager Chris DiBona reported that the number of open-source projects licensed software that had moved to GPLv3 from GPLv2 was 50% in 2009, counting the projects hosted at Google Code.^[46]

Terms and conditions

The terms and conditions of the GPL must be made available to anybody receiving a copy of the work that has a GPL applied to it ("the licensee"). Any licensee who adheres to the terms and conditions is given permission to modify the work, as well as to copy and redistribute the work or any derivative version. The licensee is allowed to charge a fee for this service, or do this free of charge. This latter point distinguishes the GPL from software licenses that prohibit commercial redistribution. The FSF argues that free software should not place restrictions on commercial use,^[47] and the GPL explicitly states that GPL works may be sold at any price.

The GPL additionally states that a distributor may not impose "further restrictions on the rights granted by the GPL". This forbids activities such as distributing of the software under a non-disclosure agreement or contract. Distributors under the GPL also grant a license for any of their patents practiced by the software, to practice those patents in GPL software.

The fourth section for version 2 of the license and the seventh section of version 3 require that programs distributed as pre-compiled binaries are accompanied by a copy of the source code, a written offer to distribute the source code via the same mechanism as the pre-compiled binary, or the written offer to obtain the source code that the user got when they received the pre-compiled binary under the GPL. The second section of version 2 and the fifth section of version 3 also require giving "all recipients a copy of this License along with the Program". Version 3 of the license allows making the source code available in additional ways in fulfillment of the seventh section. These include downloading source code from an adjacent network server or by peer-to-peer transmission, provided that is how the compiled code was available and there are "clear directions" on where to find the source code.

The FSF does not hold the copyright for a work released under the GPL, unless an author explicitly assigns copyrights to the FSF (which seldom happens except for programs that are part of the GNU project). Only the individual copyright holders have the authority to sue when a license violation takes place.

Use of licensed software

Software under the GPL may be run for all purposes, including commercial purposes and even as a tool for creating proprietary software, for example when using GPL-licensed compilers.^[48] Users or companies who distribute GPL-licensed works (e.g. software), may charge a fee for copies or give them free of charge. This distinguishes the GPL from shareware software licenses that allow copying for personal use but prohibit commercial distribution, or proprietary licenses where copying is prohibited by copyright law. The FSF argues that freedom-respecting free software should also not restrict commercial use and distribution (including redistribution).^[47] the GPL explicitly states that GPL works may be sold at any price.

In purely private (or internal) use — with no sales and no distribution — the software code may be modified and parts reused without requiring the source code to be released. For sales or distribution, the entire source code need to be made available to end users, including any code changes and additions — in that case, copyleft is applied to ensure that end users retain the freedoms defined above.^[49]

However, software running as an application program under a GPL-licensed operating system such as Linux is not required to be licensed under GPL or to be distributed with source-code availability—the licensing depends only on the used libraries and software components and not on the underlying platform.^[50] For example if a program consists only of own original custom software, or is combined with source code from other software components,^[51] then the own custom software components need not be licensed under GPL and need not make their code available; even if the underlying operating system used is licensed under the GPL, applications running on it are not considered derivative works.^[50] Only if GPLed parts are used in a program (and the program is distributed), then all other source code of the program needs to be made available under the same license terms. The GNU *Lesser* General Public license (LGPL) was created to have a weaker copyleft than the GPL, in that it does not require own custom-developed source code (distinct from the LGPLed parts) to be made available under the same license terms.

Copyleft

The distribution rights granted by the GPL for modified versions of the work are not unconditional. When someone distributes a GPL'd work plus his/her own modifications, the requirements for distributing the whole work cannot be any greater than the requirements that are in the GPL.

This requirement is known as copyleft. It earns its legal power from the use of copyright on software programs. Because a GPL work is copyrighted, a licensee has no right to redistribute it, not even in modified form (barring fair use), except under the terms of the license. One is only required to adhere to the terms of the GPL if one wishes to exercise rights normally restricted by copyright law, such as redistribution. Conversely, if one distributes copies of the work without abiding by the terms of the GPL (for instance, by keeping the source code secret), he or she can be sued by the original author under copyright law.

Copyleft thus uses copyright law to accomplish the opposite of its usual purpose: instead of imposing restrictions, it grants rights to other people, in a way that ensures the rights cannot subsequently be taken away. It also ensures that unlimited redistribution rights are not granted, should any legal flaw be found in the copyleft statement.

Many distributors of GPL'ed programs bundle the source code with the executables. An alternative

method of satisfying the copyleft is to provide a written offer to provide the source code on a physical medium (such as a CD) upon request. In practice, many GPL'ed programs are distributed over the Internet, and the source code is made available over FTP or HTTP. For Internet distribution, this complies with the license.

Copyleft applies only when a person seeks to redistribute the program. One is allowed to make private modified versions, without any obligation to divulge the modifications as long as the modified software is not distributed to anyone else. Note that the copyleft applies only to the software and not to its output (unless that output is itself a derivative work of the program^[52]). For example, a public web portal running a modified derivative of a GPL'ed content management system is not required to distribute its changes to the underlying software because its output is not a derivative.

There has been debate on whether it is a violation of the GPL to release the source code in obfuscated form, such as in cases in which the author is less willing to make the source code available. The general consensus was that while unethical, it was not considered a violation. The issue was clarified when the license was altered to require that the "preferred" version of the source code be made available.^[53]

License versus contract

The GPL was designed as a license, rather than a contract.^{[54][55]} In some Common Law jurisdictions, the legal distinction between a license and a contract is an important one: contracts are enforceable by contract law, whereas licenses are enforced under copyright law. However, this distinction is not useful in the many jurisdictions where there are no differences between contracts and licenses, such as Civil Law systems.^[56]

Those who do not accept the GPL's terms and conditions do not have permission, under copyright law, to copy or distribute GPL licensed software or derivative works. However, if they do not redistribute the GPL'd program, they may still use the software within their organization however they like, and works (including programs) constructed by the use of the program are not required to be covered by this license.

Allison Randal argued that the GPLv3 as a license is unnecessarily confusing for lay readers, and could be simplified while retaining the same conditions and legal force.^[57]

Derivations

The text of the GPL is itself copyrighted, and the copyright is held by the Free Software Foundation.

The FSF permits people to create new licenses based on the GPL, as long as the derived licenses do not use the GPL preamble without permission. This is discouraged, however, since such a license might be incompatible with the GPL^[58] and causes a perceived license proliferation.

Other licenses created by the GNU project include the GNU Lesser General Public License and the GNU Free Documentation License.

The text of the GPL is not itself under the GPL. The license's copyright disallows modification of the

license. Copying and distributing the license is allowed since the GPL requires recipients to get "a copy of this License along with the Program".^[59] According to the GPL FAQ, anyone can make a new license using a modified version of the GPL as long as he or she uses a different name for the license, does not mention "GNU", and removes the preamble, though the preamble can be used in a modified license if permission to use it is obtained from the Free Software Foundation (FSF).

Linking and derived works

Libraries

According to the FSF, "The GPL does not require you to release your modified version, or any part of it. You are free to make modifications and use them privately, without ever releasing them."^[60] However if one releases a GPL-licensed entity to the public, there is an issue regarding linking: namely, if a proprietary program uses a GPL library, is the proprietary program in violation of the GPL?

This key dispute is whether or not non-GPL software can legally statically link or dynamically link to GPL libraries. Different opinions exist on this issue. The GPL is clear in requiring that all derivative works of code under the GPL must themselves be under the GPL. Ambiguity arises with regards to using GPL libraries, and bundling GPL software into a larger package (perhaps mixed into a binary via static linking). This is ultimately a question not of the GPL *per se*, but of how copyright law defines derivative works. The following points of view exist:

Point of view: dynamic and static linking violate GPL

The Free Software Foundation (which holds the copyright of several notable GPL-licensed software products and of the license text itself) asserts that an executable which uses a dynamically linked library is indeed a derivative work. This does not however apply to separate programs communicating with one another.^[61]

The Free Software Foundation also created the LGPL, which is nearly identical to the GPL, but with additional permissions to allow linking for the purposes of "using the library".

Richard Stallman and the FSF specifically encourage library-writers to license under the GPL so that proprietary programs cannot use the libraries, in an effort to protect the free-software world by giving it more tools than the proprietary world.^[62]

Point of view: static linking violates GPL but unclear as of dynamic linking

Some people believe that while static linking produces derivative works, it is not clear whether an executable that dynamically links to a GPL code should be considered a derivative work (see Weak Copyleft). Linux author Linus Torvalds agrees that dynamic linking can create derived works but disagrees over the circumstances.^[63]

A Novell lawyer has written that dynamic linking not being derivative "makes sense" but is not "clear-cut", and that evidence for good-intentioned dynamic linking can be seen by the existence of proprietary Linux kernel drivers.^[64]

In *Galoob v. Nintendo* the United States Ninth Circuit Court of Appeals defined a derivative work as having "'form' or permanence" and noted that "the infringing work must incorporate a portion of the copyrighted work in some form",^[65] but there have been no clear court decisions to resolve this particular conflict.

Point of view: linking is irrelevant

According to an article in the Linux Journal, Lawrence Rosen (OSI general counsel) argues that the method of linking is mostly irrelevant to the question about whether a piece of software is a derivative work; more important is the question about whether the software was intended to interface with client software and/or libraries.^[66] He states, "The primary indication of whether a new program is a derivative work is whether the source code of the original program was used [in a copy-paste sense], modified, translated or otherwise changed in any way to create the new program. If not, then I would argue that it is not a derivative work,"^[66] and lists numerous other points regarding intent, bundling, and linkage mechanism. He further argues on his firm's website^[67] that such "market-based" factors are more important than the linking technique.

There is also the specific issue of whether a plugin or module (such as the NVidia or ATI graphics card kernel modules) must also be GPL, if it could reasonably be considered its own work. This point of view suggests that reasonably separate plugins, or plugins for software designed to use plugins, could be licensed under an arbitrary license if the work is GPLv2. Of particular interest is the GPLv2 paragraph:

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:

...

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.

...

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

It should be noted that the GPLv3 has a different clause:

You may convey a work based on the Program, or the modifications to produce it from

the Program, in the form of source code under the terms of Section 4, provided that you also meet all of these conditions: c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable Section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

...

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

As a case study, some supposedly proprietary plugins and themes/skins for GPLv2 CMS software such as Drupal and WordPress have come under fire, with both sides of the argument taken.^{[68][69]}

The FSF differentiates on how the plug-in is being invoked. If the Plug-in is invoked through dynamic linkage and it performs function calls to the GPL program then it is most likely a derivative work.^[70]

Communicating and bundling with non-GPL programs

The mere act of communicating with other programs does not, by itself, require all software to be GPL; nor does distributing GPL software with non-GPL software. However, minor conditions must be followed that ensures the rights of GPL software is not restricted. The following is a quote from the gnu.org GPL FAQ, which describes to what extent software is allowed to communicate with and be-bundled-with GPL programs:

'What is the difference between an "aggregate" and other kinds of "modified versions"?

An "aggregate" consists of a number of separate programs, distributed together on the same CD-ROM or other media. The GPL permits you to create and distribute an aggregate, even when the licenses of the other software are non-free or GPL-incompatible. The only condition is that you cannot release the aggregate under a license that prohibits users from exercising rights that each program's individual license would grant them.

Where's the line between two separate programs, and one program with two parts? This is a legal question, which ultimately judges will decide. We believe that a proper criterion depends both on the mechanism of communication (exec, pipes, rpc, function calls within a shared address space, etc.) and the semantics of the communication (what kinds of information are interchanged).

If the modules are included in the same executable file, they are definitely combined in one program. If modules are designed to run linked together in a shared address space, that almost surely means combining them into one program.

By contrast, pipes, sockets and command-line arguments are communication mechanisms normally used between two separate programs. So when they are used for communication, the modules normally are separate programs. But if the semantics of the communication are intimate enough, exchanging complex internal data structures, that too could be a basis to consider the two parts as combined into a larger program.

The FSF thus draws the line between "library" and "other program" via 1) "complexity" and "intimacy" of information exchange, and 2) mechanism (rather than semantics), but resigns that the question is not clear-cut and that in complex situations, case law will need to decide.

Legal status

The first known violation of the GPL was in 1989 when NeXT extended the GCC compiler to support Objective-C, but did not publicly release the changes.^[71] After an inquiry they created a public patch. There was no lawsuit filed for this violation.

In 2002, MySQL AB sued Progress NuSphere for copyright and trademark infringement in United States district court. NuSphere had allegedly violated MySQL's copyright by linking MySQL's GPL'ed code with NuSphere Gemini table without being in compliance with the license. After a preliminary hearing before Judge Patti Saris on 27 February 2002, the parties entered settlement talks and eventually settled.^[72] After the hearing, FSF commented that "Judge Saris made clear that she sees the GNU GPL to be an enforceable and binding license."^[73]

In August 2003, the SCO Group stated that they believed the GPL to have no legal validity, and that they intended to pursue lawsuits over sections of code supposedly copied from SCO Unix into the Linux kernel. This was a problematic stand for them, as they had distributed Linux and other GPL'ed code in their Caldera OpenLinux distribution, and there is little evidence that they had any legal right to do so except under the terms of the GPL. For more information, see SCO-Linux controversies and SCO v. IBM.

In April 2004, the netfilter/iptables project was granted a preliminary injunction against Sitecom Germany by Munich District Court after Sitecom refused to desist from distributing Netfilter's GPL'ed software in violation of the terms of the GPL. Harald Welte, of Netfilter, was represented by ifrOSS co-founder Till Jaeger. On July 2004, the German court confirmed this injunction as a final ruling against Sitecom.^[74] The court's justification was that:

Defendant has infringed on the copyright of plaintiff by offering the software 'netfilter/iptables' for download and by advertising its distribution, without adhering to the license conditions of the GPL. Said actions would only be permissible if defendant had a license grant... This is independent of the questions whether the licensing conditions of the GPL have been effectively agreed upon between plaintiff and defendant or not. If the GPL were not agreed upon by the parties, defendant would notwithstanding lack the necessary rights to copy,

distribute, and make the software 'netfilter/iptables' publicly available.

This exactly mirrored the predictions given previously by the FSF's Eben Moglen. This ruling was important because it was the first time that a court had confirmed that violating terms of the GPL could be copyright violation and established case law over the enforceability of the GPL version 2 under German law.^[75]

In May 2005, Daniel Wallace filed suit against the Free Software Foundation in the Southern District of Indiana, contending that the GPL is an illegal attempt to fix prices (at zero). The suit was dismissed in March 2006, on the grounds that Wallace had failed to state a valid anti-trust claim; the court noted that "the GPL encourages, rather than discourages, free competition and the distribution of computer operating systems, the benefits of which directly pass to consumers".^[76] Wallace was denied the possibility of further amending his complaint, and was ordered to pay the FSF's legal expenses.

On 8 September 2005, the Seoul Central District Court ruled that the GPL was not material to a case dealing with trade secrets derived from GPL-licensed work.^[77] Defendants argued that since it is impossible to maintain trade secrets while being compliant with GPL and distributing the work, they are not in breach of trade secrets. This argument was considered without ground.

On 6 September 2006, the gpl-violations.org project prevailed in court litigation against D-Link Germany GmbH regarding D-Link's copyright-infringing use of parts of the Linux Kernel in storage devices they distributed.^[78] The judgment provided legal precedent that the GPL is valid, legally binding, and stands in German court.^[79]

In late 2007, the BusyBox developers and the Software Freedom Law Center embarked upon a program to gain GPL compliance from distributors of BusyBox in embedded systems, suing those who would not comply. These were claimed to be the first US uses of courts for enforcement of GPL obligations. *See* BusyBox GPL lawsuits.

On 11 December 2008, the Free Software Foundation sued Cisco Systems, Inc. for copyright violations by its Linksys division, of the FSF's GPL-licensed coreutils, readline, Parted, Wget, GNU Compiler Collection, binutils, and GNU Debugger software packages, which Linksys distributes in the Linux firmware^[80] of its WRT54G wireless routers, as well as numerous other devices including DSL and Cable modems, Network Attached Storage devices, Voice-Over-IP gateways, Virtual Private Network devices and a home theater/media player device.^[81]

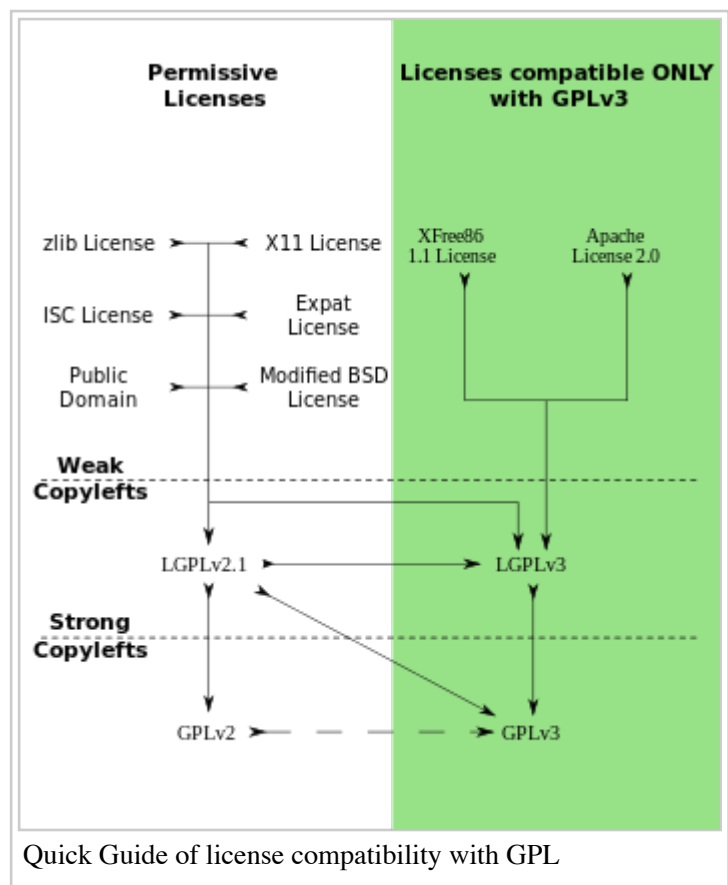
After six years of repeated complaints to Cisco by the FSF, claims by Cisco that they would correct, or were correcting, their compliance problems (not providing complete copies of all source code and their modifications), of repeated new violations being discovered and reported with more products, and lack of action by Linksys (a process described on the FSF blog as a "five-years-running game of Whack-a-Mole"^[81]) the FSF took them to court.

Cisco settled the case six months later by agreeing "to appoint a Free Software Director for Linksys" to ensure compliance, "to notify previous recipients of Linksys products containing FSF programs of their rights under the GPL," to make source code of FSF programs freely available on its website, and to make a monetary contribution to the FSF.^[82]

Compatibility and multi-licensing

Code licensed under several other licenses can be combined with a program under the GPL without conflict, as long as the combination of restrictions on the work as a whole does not put any additional restrictions beyond what GPL allows.^[83] In addition to the regular terms of the GPL, there are additional restrictions and permissions one can apply:

1. If a user wants to combine code licensed under different versions of GPL, then this is only allowed if the code with the older GPL version includes an "any later version" statement.^[84] For instance, the GPLv3 licensed GNU LibreDWG library can't be used anymore by LibreCAD and FreeCAD who have GPLv2 only dependencies.^{[85][86]}



2. Code licensed under LGPL is permitted to be linked with any other code no matter what license that code has,^[87] though the LGPL does add additional requirements for the combined work. LGPLv3 and GPLv2-only can thus commonly not be linked, as the combined Code work would add additional LGPLv3 requirements on top of the GPLv2-only licensed software. Code licensed under LGPLv2.x without the "any later version" statement can be relicensed if the whole combined work is licensed to GPLv2 or GPLv3.^[88]

FSF maintains a list^[89] of GPL-compatible free software licenses^{[90][91]} with many of the most common free software licenses, such as the original MIT/X license, the BSD license (in its current 3-clause form) and the Artistic License 2.0.^[92]

David A. Wheeler has advocated that free/open source software developers use only GPL-compatible licenses, because doing otherwise makes it difficult for others to participate and contribute code.^[93] As a specific example of license incompatibility, Sun Microsystems' ZFS cannot be included in the GPL-licensed Linux kernel, because it is licensed under the GPL-incompatible CDDL. Furthermore, ZFS is protected by patents, so distributing an independently developed GPL-ed implementation would still require Oracle's permission.^[94]

A number of businesses use multi-licensing to distribute a GPL version and sell a proprietary license to companies wishing to combine the package with proprietary code, using dynamic linking or not. Examples of such companies include MySQL AB, Digia PLC (Qt framework, before 2011 from Nokia), Red Hat (Cygwin) and Riverbank Computing (PyQt). Other companies, like the Mozilla Foundation (products include Mozilla Application Suite, Mozilla Thunderbird and Mozilla Firefox), used multi-licensing to distribute versions under the GPL and some other open-source licenses.

Use for text and other media

It is possible to use the GPL for text documents instead of computer programs, or more generally for all kinds of media, if it is clear what constitutes the source code (defined as "the preferred form of the work for making changes in it").^[95] For manuals and textbooks, though, the FSF recommends the GNU Free Documentation License (GFDL) instead, which it created for this purpose.^[96] Nevertheless, the Debian developers recommended (in a resolution adopted in 2006) to license documentation for their project under the GPL, because of the incompatibility of the GFDL with the GPL (text licensed under the GFDL cannot be incorporated into GPL software).^[97] Also, the FLOSS Manuals foundation, an organization devoted to creating manuals for free software, decided to eschew the GFDL in favor of the GPL for its texts in 2007.^[98]

If the GPL is used for fonts, any documents or images made with such fonts might also have to be distributed under the terms of the GPL. This is not the case in countries like the US and Canada where copyright law is inapplicable to the appearance of fonts, though program code inside a font file may still be covered which can complicate font embedding (since the document could be considered 'linked' to the font). FSF provides an exception for cases where this is not desired.^{[99][100]}

Reception

Legal barrier to app stores

The GPL License is incompatible with many application-distribution-systems, like the Mac App Store, and certain other software distribution platforms (on smartphones as well as PCs). The problem lies in the right "To make a copy for your neighbour", as this right is violated by the integrated DRM-Systems made to prevent copying of paid software. Even if the application is free-as-in-beer in the App Store in question, it might result in a violation of that app store's terms.^[101]

Note that there is a distinction between an *app store* which sells DRM-restricted software under proprietary licenses, and the more general concept of digital distribution via some form of online software repository: various UNIX-like distributions provide app repositories, including Fedora, RHEL, CentOS, Ubuntu, Debian, FreeBSD, OpenBSD and so on. These specific app repos all contain GPL-licensed software apps, in some cases even when the core project does not permit GPL-licensed code in the base system (for instance OpenBSD [1] (<http://www.openbsd.org/policy.html>)). In other cases, such as the Ubuntu App Store, proprietary commercial software applications *and* GPL-licensed applications are both available via the same system; the reason that the Mac App Store (and similar projects) is incompatible with GPL-licensed apps is not inherent in the concept of an app store, but is rather specifically due to Apple's terms-of-use requirement^[101] that all apps in the store utilize Apple DRM-restrictions. Ubuntu's app store does not demand any

such requirement: "These terms do not limit or restrict your rights under any applicable open source software licenses."^[102]

Microsoft

In 2001, Microsoft CEO Steve Ballmer referred to Linux as "a cancer that attaches itself in an intellectual property sense to everything it touches".^{[103][104]} In response to Microsoft's attacks on the GPL, several prominent Free Software developers and advocates released a joint statement supporting the license.^[105] Microsoft has released Microsoft Windows Services for UNIX which contains GPL-licensed code. In July 2009, Microsoft itself released a body of around 20,000 lines of Linux driver code under the GPL.^[106] The Hyper-V code that is part of the submitted code used open-source components licensed under the GPL and was originally statically linked to proprietary binary parts, the latter being inadmissible in GPL-licensed software.^[107]

"Viral" nature

Craig Mundie, Microsoft Senior Vice President has described the GPL as being "viral" in 2001.^[108] Mundie argues that the GPL has a "viral" effect in that it only allows the conveyance of whole programs, which means programs that link to GPL libraries must themselves be under a GPL-compatible license, else they cannot be combined and distributed.

In the views of Richard Stallman, Mundie's metaphor of a "virus" is wrong as software under the GPL does not "attack" or "infect" other software. Stallman believes that comparing the GPL to a virus is an extremely unfriendly thing to say, and that a better metaphor for software under the GPL would be a spider plant: If one takes a piece of it and puts it somewhere else, it grows there too.^[109]
^{[110][111]}

Barrier to commercialization

The FreeBSD project has stated that "a less publicized and unintended use of the GPL is that it is very favorable to large companies that want to undercut software companies. In other words, the GPL is well suited for use as a marketing weapon, potentially reducing overall economic benefit and contributing to monopolistic behavior" and that the GPL can "present a real problem for those wishing to commercialize and profit from software".^[112]

Richard Stallman wrote about the practice of selling license exceptions to free software licenses as an example of ethically acceptable commercialization practice. Selling exceptions here means that the copyright holder of a given software releases it (along with the corresponding source code) to the public under a free software license, "then lets customers pay for permission to use the same code under different terms, for instance allowing its inclusion in proprietary applications". Stallman considered selling exceptions "acceptable since the 1990s, and on occasion I've suggested it to companies. Sometimes this approach has made it possible for important programs to become free software". Despite that the FSF doesn't practice selling exceptions, a comparison with the X11 license (which is a noncopyleft free software license) is proposed for suggesting that this commercialization technique should be regarded as ethically acceptable. Releasing a given program under a noncopyleft free software license would permit embedding the code in proprietary software. Stallman comments that "either we have to conclude that it's wrong to release anything under the

X11 license—a conclusion I find unacceptably extreme—or reject this implication. Using a noncopyleft license is weak, and usually an inferior choice, but it's not wrong. In other words, selling exceptions permits some embedding in proprietary software, and the X11 license permits even more embedding. If this doesn't make the X11 license unacceptable, it doesn't make selling exceptions unacceptable".^[113]

GPLv3 separates community further

Some journalists^{[45][114][115]} and Toybox developer Rob Landley^[39] criticized that with the introduction of the GPLv3 the split between the open source and free software community is wider than ever. As the significantly extended GPLv3 is essentially incompatible with the GPLv2,^[84] compatibility between both is only given under the optional "*or later*" clause of the GPL, which was not taken for instance by the Linux kernel.^[12] Bruce Byfield noted that before the release of the GPLv3, the GPLv2 was a unifying element between the open-source and the free software community.^[45]

Lawrence Rosen, attorney and computer specialist, praised how the community using the Apache license were now able to work together with the GPL community in a compatible manner, as the problems of GPLv2 compatibility with Apache licensed software were resolved with the GPLv3. He said, "I predict that one of the biggest success stories of GPLv3 will be the realization that the entire universe of free and open source software can thus be combined into comprehensive open source solutions for customers worldwide."^[116]

See also

- Anti-copyright
- Dual-licensing
- European Union Public Licence (EURL)
- GPL font exception
- List of software licenses
- Permissive and copyleft licenses

References

- ¹ [^] "Debian – License information" (<http://www.debian.org/legal/licenses/>). Software in the Public Interest, Inc. Retrieved 10 December 2009.
- ² [^] ^{*a*} ^{*b*} "Licenses – Free Software Foundation" (http://www.fsf.org/licensing/licenses/index_html#GPLCompatibleLicenses). Free Software Foundation. Retrieved 10 December 2009.
- ³ [^] "Licenses by Name" (<http://opensource.org/licenses/alphabetical>). Open Source Initiative. Retrieved 10 December 2009.
- ⁴ [^] "Copyleft: Pragmatic Idealism – Free Software Foundation" (<http://www.fsf.org/licensing/essays/pragmatic.html>). Free Software Foundation. Retrieved 10 December 2009.

5. ^ "Top 20 Open Source Licenses" (<http://www.blackducksoftware.com/resources/data/top-20-open-source-licenses>). Black Duck Software. Retrieved 19 March 2014.
6. ^ GPL FAQ: Does using the GPL for a program make it GNU Software? (<https://www.gnu.org/licenses/gpl-faq.html#DoesUsingTheGPLForAProgramMakeItGNUSoftware>)
7. ^ "Freecode's statistics page" (<http://Freecode.com/stats/>).
8. ^ "SourceForge.net: Software Map" (<http://www.dwheeler.com/frozen/sourceforge-stats-20031110.html>). D Wheeler.com. Retrieved 17 November 2008.
9. ^ David A. Wheeler. "Estimating Linux's Size" (<http://www.dwheeler.com/sloc/redhat62-v1/redhat62sloc.html>).
10. ^ David A. Wheeler. "Make Your Open Source Software GPL-Compatible. Or Else" (<http://www.dwheeler.com/essays/gpl-compatible.html>).
11. ^ "Why the GPL rocketed Linux to success" (<http://www.dwheeler.com/blog/2006/09/01/#gpl-bsd>). "So while the BSDs have lost energy every time a company gets involved, the GPL'ed programs gain every time a company gets involved."
12. ^ ^a ^b ^c Torvalds, Linus. "COPYING" (<http://git.kernel.org/cgiit/linux/kernel/git/stable/linux-stable.git/tree/COPYING>). kernel.org. Retrieved 13 August 2013. *"Also note that the only valid version of the GPL as far as the kernel is concerned is this particular version of the license (ie v2, not v2.2 or v3.x or whatever), unless explicitly otherwise stated."*
13. ^ "The History of the GPL" (http://www.free-soft.org/gpl_history/). Retrieved 24 November 2011.
14. ^ ^a ^b Stallman, Richard (21 April 2006). "Presentation at the second international GPLv3 conference, held in Porto Alegre" (<http://fsfeurope.org/projects/gplv3/fisl-rms-transcript.en.html#before-gnu-gpl>).
15. ^ "Why Upgrade to GPL Version 3 --GPLv3" (<http://gplv3.fsf.org/rms-why.html>). Fsf.org. Retrieved 17 March 2011.
16. ^ "FSF releases the GNU General Public License, version 3 – Free Software Foundation – working together for free software" (http://www.fsf.org/news/gplv3_launched). Fsf.org. Retrieved 15 January 2011.
17. ^ "GNU General Public License, version 1" (<https://www.gnu.org/licenses/old-licenses/gpl-1.0.html>).
18. ^ "New General Public License" (<http://groups.google.com/group/gnu.announce/msg/bf254a45c6f512f3>).
19. ^ For the reasoning see The GNU project (<https://www.gnu.org/gnu/thegnuproject.html>).
20. ^ ^a ^b Stallman, Richard (25 February 2006). "Presentation in Brussels, Belgium—the first day of that year's FOSDEM conference." (<http://www.ifso.ie/documents/rms-gplv3-2006-02-25.html>).
21. ^ Interview with Richard Stallman (http://www.freesoftwaremagazine.com/articles/interview_with_richard_stallman), Free Software Magazine, 23 January 2008.
22. ^ "A Quick Guide to GPLv3 – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/licenses/quick-guide-gplv3.html#neutralizing-laws-that-prohibit-free-software-but-not-forbidding-drm>). Free Software Foundation.
23. ^ "GPLv3: Drafting version 3 of the GNU General Public License" (<http://fsfeurope.org/projects/gplv3/>). Free Software Foundation Europe.
24. ^ "gplv3.fsf.org comments for discussion draft 4" (<http://gplv3.fsf.org/comments/gplv3-draft-4.html>).

25. ^ "gplv3.fsf.org comments for draft 1" (<http://gplv3.fsf.org/comments/gplv3-draft-1>). "Showing comments in file 'gplv3-draft-1' [...] found 962"
26. ^ "gplv3.fsf.org comments for draft 2" (<http://gplv3.fsf.org/comments/gplv3-draft-2.html>). "Showing comments in file 'gplv3-draft-1' [...] found 727"
27. ^ "gplv3.fsf.org comments for draft 3" (<http://gplv3.fsf.org/comments/gplv3-draft-3>). "Showing comments in file 'gplv3-draft-3' [...] found 649"
28. ^ "gplv3.fsf.org comments for draft 4" (<http://gplv3.fsf.org/comments/gplv3-draft-4.html>). "Showing comments in file 'gplv3-draft-4' [...] found 298"
29. ^ "Guide to the third draft of GPLv3" (<http://gplv3.fsf.org/gpl3-dd3-guide>).
30. ^ "Final Discussion Draft" (<http://gplv3.fsf.org/gpl-draft-2007-05-31.html>). Retrieved 4 June 2007.
31. ^ "GPL version 3 FAQ" (<http://gplv3.fsf.org/dd3-faq>). Retrieved 4 June 2007.
32. ^ "Fourth Discussion Draft Rationale" (<http://gplv3.fsf.org/gpl3-dd4-rationale.pdf>). Retrieved 4 June 2007.
33. ^ Tiemann, Michael (7 June 2007). "GNU Affero GPL version 3 and the "ASP loophole" " (<http://opensource.org/node/152>). OSI. Retrieved 19 August 2013.
34. ^ List of free-software licences on the FSF website (<http://www.fsf.org/licensing/licenses/index.html>):
"We recommend that developers consider using the GNU AGPL for any software which will commonly be run over a network".
35. ^ Why did you decide to write the GNU Affero GPLv3 as a separate license? (<https://www.gnu.org/licenses/gpl-faq.html#SeparateAffero>) on gnu.org
36. ^ "Kernel developers' position on GPLv3" (<http://lwn.net/Articles/200422/>). LWN.net. Retrieved 4 June 2007.
37. ^ "GPL 3 Overview" (<http://www.techlawforum.net/post.cfm/gpl-3-overview>). Tech LawForum. 29 June 2007. Retrieved 2 September 2013.
38. ^ "A Quick Guide to GPLv3 – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/licenses/quick-guide-gplv3.html#new-compatible-licenses>). Free Software Foundation.
39. ^ ^a ^b Landley, Rob. "CELF 2013 Toybox talk - <http://landley.net/talks/celf-2013.txt>" (<http://landley.net/talks/celf-2013.txt>). landley.net. Retrieved 21 August 2013. "GPLv3 broke "the" GPL into incompatible forks that can't share code. [...] FSF expected universal compliance, but hijacked lifeboat clause when boat wasn't sinking.[...]"
40. ^ "GNU General Public License" (<https://www.gnu.org/licenses/gpl.html>). Retrieved 15 June 2012.
41. ^ "Top 20 licenses" (<http://www.blackducksoftware.com/resources/data/top-20-licenses>). Black Duck Software. 23 August 2013. Retrieved 23 August 2013. "1. GNU General Public License (GPL) 2.0 33%, 2. Apache License 13%, 3. GNU General Public License (GPL) 3.0 12%"
42. ^ Proffitt, Brian (16 December 2011). "GPL, copyleft use declining faster than ever - Data suggests a sharper rate of decline, which raises the question: why?" (<http://www.itworld.com/it-managementstrategy/233753/gpl-copyleft-use-declining-faster-ever>). IT world. Retrieved 23 August 2013.
43. ^ "GPL use in Debian on the rise: study" (<http://www.itwire.com/business-it-news/open-source/52838-gpl-use-in-debian-on-the-rise-study>). Itwire.com. Retrieved 2 September 2013.
44. ^ "Surveying open source licenses" (<https://lwn.net/Articles/547400/>). Lwn.net. Retrieved 2 September 2013.

45. ^{a b c} Byfield, Bruce (22 November 2011). "7 Reasons Why Free Software Is Losing Influence: Page 2" (http://www.datamation.com/open-source/7-reasons-why-free-software-is-losing-influence_2.html). Datamation.com. Retrieved 23 August 2013. "*At the time, the decision seemed sensible in the face of a deadlock. But now, GPLv2 is used for 42.5% of free software, and GPLv3 for less than 6.5%, according to Black Duck Software.*"
46. ^a Asay, Matt (23 July 2009). "GPLv3 hits 50 percent adoption | The Open Road - CNET News" (http://news.cnet.com/8301-13505_3-10294452-16.html). News.cnet.com. Retrieved 2 September 2013.
47. ^{a b} "Selling Free Software" (<https://www.gnu.org/philosophy/selling.html>). Free Software Foundation.
48. ^a GPL FAQ: Use GPL Tools to develop non-free programs (<https://www.gnu.org/licenses/gpl-faq.html#CanIUseGPLToolsForNF>)
49. ^a GPL FAQ: GPL require source posted to public (<https://www.gnu.org/licenses/gpl-faq.html#GPLRequireSourcePostedPublic>), Unreleased modifications (<https://www.gnu.org/licenses/gpl-faq.html#UnreleasedMods>), Internal Distribution (<https://www.gnu.org/licenses/gpl-faq.html#InternalDistribution>)
50. ^{a b} GPL FAQ: Port program to GNU/Linux (<https://www.gnu.org/licenses/gpl-faq.html#PortProgramToGL>)
51. ^a example: if *only* GNU Lesser General Public License- (LGPL-) libraries, LGPL-*software-components* and components with permissive free software licenses are used (thus not GPL itself), then *only* the source code of LGPL parts has to be made available— for the developer's own self-developed software components this is not required (even when the underlying operating system used is licensed under GPL, as is the case with Linux).
52. ^a A counter example is the GPL'ed GNU Bison: the parsers it outputs *do* contain parts of itself and are therefore derivatives which would fall under the GPL, if it were not for a special exception granted by GNU Bison: "Conditions for Using Bison" (https://www.gnu.org/software/bison/manual/html_node/Conditions.html). Retrieved 11 December 2008.
53. ^a "Reasoning behind the "preferred form" language in the GPL" (<http://lwn.net/Articles/431651/>). LWN.net. 7 March 2011.
54. ^a "Essay by Stallman explaining why a license is more suitable than a contract" (<https://www.gnu.org/philosophy/no-ip-ethos.html>).
55. ^a "Eben Moglen explaining why the GPL is a license and why it matters" (<http://fsfeurope.org/projects/gplv3/barcelona-moglen-transcript.en.html#q7-a-contract>).
56. ^a Guadamuz-Gonzalez, Andres (2004). "Viral contracts or unenforceable documents? Contractual validity of copyleft licenses" (http://papers.ssrn.com/sol3/papers.cfm?abstract_id=569101). *European Intellectual Property Review* **26** (8): 331–339.
57. ^a Allison Randal (14 May 2007). "GPLv3, Clarity and Simplicity" (http://web.archive.org/web/20081015194558/http://radar.oreilly.com/archives/2007/05/gplv3_clarity_a.html). Archived from the original (http://radar.oreilly.com/archives/2007/05/gplv3_clarity_a.html) on 15 October 2008.
58. ^a "GPL FAQ: Can I modify the GPL and make a modified license?" (<http://www.fsf.org/licensing/licenses/gpl-faq.html#ModifyGPL>).
59. ^a "The GNU General Public License Version 3" (<https://www.gnu.org/copyleft/gpl.html>). Free Software Foundation. 29 June 2007. Retrieved 21 July 2009.

60. ^ "GPL FAQ: Does the GPL require that source code of modified versions be posted to the public?" (<https://www.gnu.org/licenses/gpl-faq.html#GPLRequireSourcePostedPublic>).
61. ^ "Frequently Asked Questions about the GNU Licenses – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/licenses/gpl-faq.html>). FSF. Retrieved 15 March 2011.
62. ^ "Why you shouldn't use the Lesser GPL for your next library – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/philosophy/why-not-lgpl.html>). Gnu.org. Retrieved 15 January 2011.
63. ^ Linus Torvalds, GPL only modules (<https://lkml.org/lkml/2006/12/17/79>), *linux-kernel* mailing list (17 December 2006).
64. ^ Matt Asay, The GPL: Understanding the License that Governs Linux (<http://www.novell.com/coolsolutions/feature/1532.html>), *Novell Cool Solutions Feature* (16 Jan 2004).
65. ^ *Lewis Galoob Toys, Inc. v. Nintendo of America, Inc.*, 964 F.2d 965 (<http://law.justia.com/cases/federal/appellate-courts/F2/964/965/341457/>), ¶10 (9th Cir. 21 May 1992).
66. ^ ^a ^b Lawrence Rosen, Derivative Works (<http://www.linuxjournal.com/article/6366>), *Linux Journal* (1 January 2003).
67. ^ Lawrence Rosen, Derivative Works (<http://www.rosenlaw.com/lj19.htm>), *rosenlaw.com* (25 May 2004).
68. ^ "Why They're Wrong: WordPress Plugins Shouldn't Have to be GPL" (<http://www.webmaster-source.com/2009/01/29/why-theyre-wrong-wordpress-plugins-shouldnt-have-to-be-gpl/>). Webmaster-source.com. 29 January 2009. Retrieved 15 January 2011.
69. ^ "Licensing FAQ" (<http://drupal.org/licensing/faq#q7>). Drupal.org. Retrieved 15 January 2011.
70. ^ "Frequently Asked Questions about the GNU Licenses – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/licenses/gpl-faq.html#GPLPluginsInNF>). Gnu.org. Retrieved 15 January 2011.
71. ^ 12 Years of GPL Compliance: A Historical Perspective (<http://ebb.org/bkuhn/talks/LinuxTag-2011/compliance.html>), Slide 10
72. ^ See *Progress Software Corporation v. MySQL AB*, 195 F. Supp. 2d 328 (D. Mass. 2002), on defendant's motion for preliminary injunction.
73. ^ "Judge Saris defers GNU GPL Questions for Trial in MySQL vs. Progress Software" (<https://www.gnu.org/press/2002-03-01-pi-MySQL.html>). gnu.org. Retrieved 24 March 2011.
74. ^ Harald Welte vs. Sitecom, final order, translated from German by Jens Maurer (<http://www.groklaw.net/article.php?story=20040725150736471>)
75. ^ Bird & Bird, A Review of German Case Law on the GNU General Public License (http://www.twobirds.com/English/News/Articles/Pages/2007/Review_German_GNU_General_Public_License.aspx), 17 December 2007, retrieved 1 March 2012
76. ^ Dismissal (<http://www.groklaw.net/pdf/WallaceFSFGrantingDismiss.pdf>) of Wallace v. FSF. From this article (<http://www.groklaw.net/article.php?story=20060320201540127>) on Groklaw.
77. ^ Seoul Central District Court ruling (?) (in Korean) (<https://web.archive.org/web/20071018211119/http://korea.gnu.org/gv/sentence.html>)
78. ^ gpl-violations.org project prevails in court case on GPL violation by D-Link (http://gpl-violations.org/news/20060922-dlink-judgement_frankfurt.html)

79. ^ D-Link Judgment (English translation) (http://www.jbb.de/judgment_dc_frankfurt_gpl.pdf) (**English**)
D-Link Judgement (http://www.jbb.de/fileadmin/download/urteil_lg_muenchen_gpl.pdf) (**German**)
80. ^ Ewing, James (1 August 2004). "Linux on Linksys Wi-Fi Routers" (<http://www.linuxjournal.com/article/7322>). Linux Journal. Retrieved 23 January 2012.
81. ^ ^a ^b "Free Software Foundation Files Suit Against Cisco For GPL Violations" (<https://www.fsf.org/news/2008-12-cisco-suit>) (Press release). Free Software Foundation. 11 December 2008. Retrieved 22 August 2011.
82. ^ "FSF Settles Suit Against Cisco" (<https://www.fsf.org/news/2009-05-cisco-settlement.html>) (Press release). Free Software Foundation. 20 May 2009. Retrieved 22 August 2011.
83. ^ "The GNU General Public License v3.0 – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/licenses/gpl.html>). fsf.org. Retrieved 24 March 2010.
84. ^ ^a ^b "Frequently Asked Questions about the GNU Licenses – Is GPLv3 compatible with GPLv2?" (<https://www.gnu.org/licenses/gpl-faq.html#v2v3Compatibility>). gnu.org. Retrieved 3 June 2014. *"No. Some of the requirements in GPLv3, such as the requirement to provide Installation Information, do not exist in GPLv2. As a result, the licenses are not compatible: if you tried to combine code released under both these licenses, you would violate section 6 of GPLv2. However, if code is released under GPL "version 2 or later," that is compatible with GPLv3 because GPLv3 is one of the options it permits."*
85. ^ Larabel, Michael (24 January 2013). "FSF Wastes Away Another "High Priority" Project" (http://www.phoronix.com/scan.php?page=news_item&px=MTI4Mjc). Phoronix. Retrieved 22 August 2013. *"Both LibreCAD and FreeCAD both want to use LibreDWG and have patches available for supporting the DWG file format library, but can't integrate them. The programs have dependencies on the popular GPLv2 license while the Free Software Foundation will only let LibreDWG be licensed for GPLv3 use, not GPLv2."*
86. ^ Prokoudine, Alexandre (27 December 2012). "LibreDWG drama: the end or the new beginning?" (<http://libregraphicsworld.org/blog/entry/libredwg-drama-the-end-or-the-new-beginning>). libregraphicsworld.org. Retrieved 23 August 2013. *"[...]the unfortunate situation with support for DWG files in free CAD software via LibreDWG. We feel, by now it ought to be closed. We have the final answer from FSF. [...] "We are not going to change the license."*
87. ^ "GNU Lesser General Public License v2.1 – GNU Project – Free Software Foundation (FSF)" (<https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html#SEC2>). fsf.org. Retrieved 26 April 2011.
88. ^ "Frequently Asked Questions about the GNU Licenses – How are the various GNU licenses compatible with each other?" (<https://www.gnu.org/licenses/gpl-faq.html#AllCompatibility>). The official site. Retrieved 13 April 2011.
89. ^ "Various licenses with comments – GPL-Compatible Free Software Licenses" (<https://www.gnu.org/licenses/license-list.html#GPLCompatibleLicenses>). FSF. Retrieved 20 April 2012.
90. ^ "Frequently Asked Questions about the GNU Licenses – What does it mean to say that two licenses are "compatible"?" (<https://www.gnu.org/licenses/gpl-faq.html#WhatIsCompatible>). FSF. Retrieved 14 April 2011.
91. ^ "Frequently Asked Questions about the GNU Licenses – What does it mean to say a license is "compatible with the GPL?" " (<https://www.gnu.org/licenses/gpl-faq.html#WhatDoesCompatMean>). FSF. Retrieved 14 April 2011.

92. ^ "Black Duck Open Source Resource Center" (<http://www.blackducksoftware.com/oss/licenses/>). blackducksoftware.com. Retrieved 26 April 2011.
93. ^ "Make Your Open Source Software GPL-Compatible. Or Else." (<http://www.dwheeler.com/essays/gpl-compatible.html>). David A. Wheeler. Retrieved 26 April 2011.
94. ^ Linux: ZFS, Licenses and Patents | KernelTrap (<http://kerneltrap.org/node/8066>)
95. ^ Free Software Foundation: Frequently Asked Questions about the GNU Licenses: Can I use the GPL for something other than software? (<https://www.gnu.org/licenses/gpl-faq.html#GPLOtherThanSoftware>). Retrieved 20 June 2009.
96. ^ GNU project: Frequently Asked Questions about the GNU Licenses: Why don't you use the GPL for manuals? (<https://www.gnu.org/licenses/gpl-faq.html#WhyNotGPLForManuals>) . Retrieved 20 June 2009.
97. ^ Debian Project: Resolution: Why the GNU Free Documentation License is not suitable for Debian (http://www.debian.org/vote/2006/vote_001.en.html#amendmenttexta). Voted February–March 2006. . Retrieved 20 June 2009.
98. ^ FLOSS Manuals foundation: License Change (<http://en.flossmanuals.net/bin/view/Blog/LicenseChange>) 6 June 2007. . Retrieved 20 June 2009.
99. ^ "Font Licensing" (<http://www.fsf.org/blogs/licensing/20050425novalis>).
100. ^ "How does the GPL apply to fonts?" (<https://www.gnu.org/licenses/gpl-faq.html#FontException>).
101. ^ ^a ^b "The GPL, the App Store and You (<http://www.tuaw.com/2011/01/09/the-gpl-the-app-store-and-you/>)"
102. ^ "Ubuntu One : Terms and Conditions" (<https://one.ubuntu.com/terms>). One.ubuntu.com. 29 August 2013. Retrieved 25 September 2013.
103. ^ Newbart, Dave (1 June 2001). "Microsoft CEO takes launch break with the Sun-Times" (<http://web.archive.org/web/20010615205548/http://suntimes.com/output/tech/cst-fin-micro01.html>). *Chicago Sun-Times*. Archived from the original (<http://suntimes.com/output/tech/cst-fin-micro01.html>) on 15 June 2001.(Internet archive link)
104. ^ text of GPL v1 with reference to source code download site at microsoft.com (<http://www.dwheeler.com/frozen/microsoft-interix-gpl.txt>)
105. ^  *Free Software Leaders Stand Together*. Wikisource.
106. ^ Clarke, Gavin (20 July 2009). "Microsoft embraces Linux cancer to sell Windows servers" (http://www.theregister.co.uk/2009/07/20/microsoft_windows_drivers_linux/). The Register.
107. ^ Clarke, Gavin (23 July 2009). "Microsoft opened Linux-driver code after 'violating' GPL" (http://www.theregister.co.uk/2009/07/23/microsoft_hyperv_gpl_violation/). The Register.
108. ^ "Speech Transcript – Craig Mundie, The New York University Stern School of Business" (<http://www.microsoft.com/presspass/exec/craig/05-03sharedsource.msp>), *Prepared Text of Remarks by Craig Mundie*, Microsoft Senior Vice President, The Commercial Software Model The New York University Stern School of Business 3 May 2001
109. ^ Poynder, Richard (21 March 2006). "The Basement Interviews: Freeing the Code" (http://www.archive.org/stream/The_Basement_Interviews/Richard_Stallman_Interview_djvu.txt). Retrieved 5 February 2010.

110. ^ Chopra, Samir; Dexter, Scott (14 August 2007). *Decoding liberation: the promise of free and open source software* (<http://books.google.com/?id=c7ppFih2mSwC&lpg=PT74>). Routledge. p. 56. ISBN 0-415-97893-9.
111. ^ Williams, Sam (March 2002). *Free as in Freedom: Richard Stallman's Crusade for Free Software* (<http://oreilly.com/openbook/freedom/ch02.html>). O'Reilly Media. ISBN 0-596-00287-4.
112. ^ GPL Advantages and Disadvantages (http://www.freebsd.org/doc/en_US.ISO8859-1/articles/bsdl-gpl/article.html#GPL-ADVANTAGES), FreeBSD: [doc/en_US.ISO8859-1/articles/bsdl-gpl/article.sgml,v1.5 2006/10/16 12:35:23 keramida Exp](http://www.freebsd.org/doc/en_US.ISO8859-1/articles/bsdl-gpl/article.sgml,v1.5%2006/10/16%2012:35:23%20keramida%20Exp)
113. ^ Richard Stallman (2010). On Selling Exceptions to the GNU GPL (<http://www.fsf.org/blogs/rms/selling-exceptions>). Free Software Foundation.
114. ^ Linux Creator Calls GPLv3 Authors 'Hypocrites' As Open Source Debate Turns Nasty (https://web.archive.org/web/20080413091038/http://www.informationweek.com/blog/main/archives/2007/07/linux_creator_c.html) [...] *the latest sign of a growing schism in the open source community between business-minded developers like Torvalds and free software purists.*
115. ^ Torvalds Still Keen On GPLv2 (<http://www.internetnews.com/dev-news/article.php/3720371/Torvalds+Still+Keen+On+GPLv2.htm>) *"In some ways, Linux was the project that really made the split clear between what the FSF is pushing which is very different from what open source and Linux has always been about, which is more of a technical superiority instead of a -- this religious belief in freedom," Torvalds told Zemlin. So, the GPL Version 3 reflects the FSF's goals and the GPL Version 2 pretty closely matches what I think a license should do and so right now, Version 2 is where the kernel is."*
116. ^ "Comments on GPLv3" (<http://www.rosenlaw.com/GPLv3-Comments.htm>). Rosenlaw.com. 2007. Retrieved 22 August 2014.

External links

- GNU General Public License (<https://gnu.org/licenses/gpl.html>) (version 3)
- History of the GPL (http://www.free-soft.org/gpl_history/)
- GNU General Public License v1.0 (<https://gnu.org/licenses/old-licenses/gpl-1.0.txt>)—This version is deprecated by the FSF.
- GNU General Public License v2.0 (<https://gnu.org/licenses/old-licenses/gpl-2.0.txt>)—This version is deprecated by the FSF but is still used by many software projects, including Linux and GNU packages
- The Emacs General Public License (http://www.free-soft.org/gpl_history/emacs_gpl.html), a February 1988 version, a direct predecessor of the GNU GPL
- A Practical Guide to GPL Compliance (<https://softwarefreedom.org/resources/2008/compliance-guide.html>) (Covers GPLv2 and v3)—from the Software Freedom Law



Wikibooks has a book on the topic of: **FOSS Licensing**

Center

- A paper on enforcing the GPL (http://sapnakumar.org/Publications_files/EnfGPL.pdf)
- Frequently Asked Questions about the GPL (<https://gnu.org/licenses/gpl-faq.html>)
- GPL, BSD, and NetBSD – why the GPL rocketed Linux to success (<http://www.dwheeler.com/blog/2006/09/01/#gpl-bsd>) by David A. Wheeler
- GNU General Public License and Commentaries (<http://www.rattlesnake.com/software-law/GNU-GPL-and-Commentaries.html>), edited by Robert Chassell
- List of presentation transcripts about the GPL and free software licenses (https://wiki.fsfe.org/Transcripts#Licences_and_the_GNU_GPL)
- Make Your Open Source Software GPL-Compatible. Or Else. (<http://www.dwheeler.com/essays/gpl-compatible.html>) (David A. Wheeler, 7 April 2004)—why a GPL-compatible license is important to the health of a project
- The Labyrinth of Software Freedom (http://www.softpanorama.org/Copyright/License_classification/index.shtml) (BSD vs GPL and social aspects of free licensing debate), by Dr. Nikolai Bezroukov

Retrieved from "http://en.wikipedia.org/w/index.php?title=GNU_General_Public_License&oldid=627152462"

Categories: [Computer law](#) | [Free and open-source software licenses](#) | [GNU Project](#) | [Copyleft](#)

- This page was last modified on 26 September 2014 at 12:56.
- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.