

What Is Copyleft?

GNU philosophy

This essay was originally published on <http://gnu.org>, in 1996.

This document is part of GNU philosophy, the GNU Project's exhaustive collection of articles and essays about free software and related matters.

Copyright © 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009 Free Software Foundation, Inc.

Verbatim copying and distribution of this entire document are permitted worldwide, without royalty, in any medium, provided this notice is preserved.

What Is Copyleft?

Copyleft is a general method for making a program (or other work) free, and requiring all modified and extended versions of the program to be free as well.

The simplest way to make a program free software is to put it in the public domain, uncopyrighted. This allows people to share the program and their improvements, if they are so minded. But it also allows uncooperative people to convert the program into proprietary software. They can make changes, many or few, and distribute the result as a proprietary product. People who receive the program in that modified form do not have the freedom that the original author gave them; the middleman has stripped it away.

In the GNU Project, our aim is to give *all* users the freedom to redistribute and change GNU software. If middlemen could strip off the freedom, we might have many users, but those users would not have freedom. So instead of putting GNU software in the public domain, we “copyleft” it. Copyleft says that anyone who redistributes the software, with or without changes, must pass along the freedom to further copy and change it. Copyleft guarantees that every user has freedom.

Copyleft also provides an incentive for other programmers to add to free software. Important free programs such as the GNU C++ compiler exist only because of this.

Copyleft also helps programmers who want to contribute improvements to free software get permission to do so. These programmers often work for companies or universities that would do almost anything to get more money. A programmer may want to contribute her changes to the community, but her employer may want to turn the changes into a proprietary software product.

When we explain to the employer that it is illegal to distribute the improved version except as free software, the employer usually decides to release it as free software rather than throw it away.

To copyleft a program, we first state that it is copyrighted; then we add distribution terms, which are a legal instrument that gives everyone the rights to use, modify, and redistribute the program’s code, *or any program derived from it*, but only if the distribution terms are unchanged. Thus, the code and the freedoms become legally inseparable.

Proprietary software developers use copyright to take away the users’ freedom; we use copyright to guarantee their freedom. That’s why we reverse the name, changing “copyright” into “copyleft.”

Copyleft is a way of using of the copyright on the program. It doesn’t mean abandoning the copyright; in fact, doing so would make copyleft impossible. The “left” in “copyleft” is not a reference to the verb “to leave”—only to the direction which is the inverse of “right.”

Copyleft is a general concept, and you can’t use a general concept directly; you can only use a specific implementation of the concept. In the GNU Project, the specific distribution terms that we use for most software are contained in the *GNU General Public License*. The GNU General Public License is often called the GNU GPL for short. There is also a Frequently Asked Questions page about the GNU GPL, at <http://gnu.org/licenses/gpl-faq.html>. You can also read about why the FSF gets copyright assignments from contributors, at <http://gnu.org/copyleft/why-assign.html>.

An alternate form of copyleft, the *GNU Lesser General Public License* (LGPL), applies to a few (but not all) GNU libraries. To learn more about properly using the LGPL, please read the article “Why You Shouldn’t Use the Lesser GPL for Your Next Library,” available at <http://gnu.org/philosophy/why-not-lgpl.html>.

The *GNU Free Documentation License* (FDL) is a form of copyleft intended for use on a manual, textbook or other document to assure everyone the effective freedom to copy and redistribute it, with or without modifications, either commercially or noncommercially.

The appropriate license is included in many manuals and in each GNU source code distribution.

All these licenses are designed so that you can easily apply them to your own works, assuming you are the copyright holder. You don’t have to modify the license to do this, just include a copy of the license in the work, and add notices in the source files that refer properly to the license.

Using the same distribution terms for many different programs makes it easy to copy code between various different programs. When they all have the same distribution terms, there is no problem. The Lesser GPL, version 2, includes a provision that lets you alter the distribution terms to the ordinary GPL, so that you can copy code into another program covered by the GPL. Version 3 of the Lesser GPL is built as an exception added to GPL version 3, making the compatibility automatic.

If you would like to copyleft your program with the GNU GPL or the GNU LGPL, please see the license instructions page, at <http://gnu.org/copyleft/gpl-howto.html>, for advice. Please note that you must use the entire text of the license you choose. Each is an integral whole, and partial copies are not permitted.

If you would like to copyleft your manual with the GNU FDL, please see the instructions at the end of the FDL text, and the GFDL instructions page, at <http://gnu.org/copyleft/fdl-howto.html>. Again, partial copies are not permitted.