

The GNU project and Free Software

You only have the right to use most software under licence

Whether they compose music, paint pictures or write software, those who create cultural artefacts retain the copyright to their works. Other people only have permission to “use” these artefacts under licence. An item's creator can stipulate any licence conditions on this use, within the law of the land. Works for which nobody can claim ownership, and which anyone can therefore use, reside in the “public domain”. Works automatically fall into the public domain 70 years after the death of their creator.

Richard Stallman enjoyed a community of shared ideas

Companies have only relatively recently treated software as a commodity. In the earlier days of computing, companies made most of their profit on hardware sales, and happily allowed people to modify and share the software that ran on this hardware. A computer scientist, named Richard M Stallman (often referred to as “*rms*” as per his computer login) worked in the 1970s at *MIT* labs in America. He found the academic atmosphere of shared learning attractive, and useful as a way to improve software.

Companies began to “privatise” software, so *RMS* reacted

In the early eighties, companies started to realise that they could sell software licenses apart from hardware. They therefore became less willing to allow people to share the software, or its source-code blueprints. Stallman found this divisive; he thought that it could become a menace to a society driven by information. He believed that such a society could not afford to have the mechanisms of its information distribution obscured and held to ransom. He decided, therefore, to write an operating system where everyone had the guaranteed permission to use, share and modify every aspect of the software. He felt such a platform necessary to ensure freedom in a world dominated by computers.

Copyleft emerged to protect users' and programmers' freedom

Stallman decided not to place his operating system in the public domain, as he felt that companies could then take his code, change it a bit, and then release their proprietary version, thereby “privatising” his work and defeating what he had wished to achieve. He decided, therefore, to retain his copyright, but to licence the code in a way that anyone could use it as they wished; this included sharing it, modifying it and even selling it, so long as they did not prevent others from doing the same, even with the subsequently modified versions. He termed this turning of copyright on its head “copyleft”.

With Free Software, one means “free” as in freedom

In deciding to write an operating system specifically to ensure the freedom of its users, Stallman launched the idea of consciously Free software. The “Free” construes the same meaning as the free in “freedom”, and not necessarily the “free” in “free gift”. Indeed, *RMS* had no problem with the notion of selling Free Software, so long as the seller did not accompany this with restrictions in sharing and modifying it. English confuses the two meanings of free by subsuming them into one word. Other languages tend to provide two words: French’s “*libre*” and “*gratuit*”, for example.

Stallman decided to create a Free *Unix*-like system

Stallman had no particular love for *Unix*. He had enjoyed the more obscure operating systems of a previous era. Nevertheless, he realised that *Unix* had gained popularity, and that his Free operating system needed to feel familiar to those who used it. He decided, therefore, to make his operating system look and feel like *Unix* in all but name. He planned to reconstruct a free version of each little program that made up the *Unix* universe until, eventually, he could provide a total replacement. He hoped any *Unix* user could feel at home in this *Unix* replacement, but without the encumbrance of *Unix*'s increasingly onerous licence restrictions.

***GNU*: a whimsical acronym for a serious project**

In an Internet posting in 1983, Stallman announced his intention to program his new Free operating system. He dubbed his endeavour the *GNU* project, which stood for “*GNU*'s Not *Unix*”. Note that the first *G* (pronounced) leads to the acronym's standing for itself: a typical programmer's whimsy. After publishing his “*GNU Manifesto*”, he worked for many years on the gargantuan project. He created a Free C compiler, a powerful text editor and versions of many of the tools one found in a typical *Unix* distribution. Despite this partial success, commentators wondered whether Stallman could pull off the task of creating a complete operating system: he had no kernel, the master program which controls all the others. People still had to rely on the proprietary *Unix* kernel for the moment, and it seemed as if *GNU*'s Free kernel lay far on the horizon.

The *General Public Licence* enshrined copyleft in legalese

To keep things on a legal footing, Stallman co-wrote the *GNU General Public Licence* (the *GPL*). He released his software under this licence, which provided a legal basis for copyleft. Other programmers began to use the licence too, and Stallman established the *Free Software Foundation* to marshal such development. Without a Free *GNU* kernel, however, people had to use these *GPL* programs upon an otherwise proprietary system.

Other Free software projects have also thrived

While Stallman's *GNU* project had a pioneering scope, others began their own, independent Free software projects. These included the *X Window System*, to provide graphical interfaces to *Unix* and *Unix*-like systems. Later groups worked on alternative Free *Unix*-like operating systems based upon the *BSD* flavour of *Unix*. Many of these projects used Free software licences other than the *GPL*. Often, these licences, whilst giving to users all the freedoms necessary for the definition of Free Software, did not guarantee *copyleft*. This meant that companies could “privatise” the software into a proprietary branch – the very reason why Stallman had not placed his software into the public domain in the first place. Some claim that this complete lack of restrictions makes such licences even “freer” than the *GPL*. Others warn that they do not protect the very freedom they provide. Whatever the case, one finds, on the average Free operating system, a collection of different licences, some copyleft, and others not.

Open Source means Free Software minus its philosophy

In the late 90s, some worried that the term “Free Software” had an ambiguity about it which needed resolving. They also felt that conservative businesses could find the philosophical aspects of the Free Software movement unpalatable. They wished to focus more upon the shared and open methodology that helped to create good quality code; they reinterpreted Free Software as a useful engineering paradigm, with unfettered access to the source code at its core. They termed such software “Open Source”. In their definition of the term, it became clear that, beneath the marketing hype, all Open Source software has necessarily also to fit the definition of Free Software. There still, however, emerged a split between the Free Software movement, that wished to emphasise the philosophical importance of freedom, and Open Sourcers, who wished to focus on more pragmatic benefits of the same freedoms.

Further reading

All about Free Software and the *GPL* from the organisation that spawned it
<http://www.gnu.org/philosophy/philosophy.html>

Everything “Open Source” from the horses' mouths
<http://www.opensource.org/>

A fascinating application of the Free methodology to the rest of culture
<http://www.creativecommons.org/>

An interesting printed collection of Stallman's writings
 Free Software, Free Society: Selected Essays of Richard M. Stallman
 Richard M Stallman 2002, GNU Press, ISBN: 1882114981