

# **An Intellectual Property Angle on Free Software Development**

**Maureen O'Sullivan**

## *Introduction*

According to *Wired* magazine, Argentina was the first country in the world to consider mandating the use of free software by all government offices.<sup>i</sup> After tightening up its anti-piracy legislation in 1998, it came under persistent pressure from groups such as Software Legal, due to the alleged repeat piracy offences by government departments, which apparently failed to observe these stricter copyright laws. From the copyright industry's point of view, it may have been wiser to turn a blind eye to such abuses. Their dogged pursuit of this matter worldwide reduced piracy rates but also had the effect of driving both the private and public sectors to seek alternatives which have been satisfied by the renaissance and spread of free software around the globe.

This paper talks about the free-proprietary-free software development cycle over the past 30 years, dealing with the introduction of copyright law and the dubious economics of software piracy. It shows that free software continued to exist, despite the onslaught of copyright's strictures. Free software and open source are defined and distinguished and their various licensing schemes discussed. The most popular free and open source software licence, the GNU GPL is analysed, both from legal and socio-legal perspectives. Latin America's initiative on free software, initiated by groups such as Fundación Vía Libre, Software Livre and the Free Software Consortium is at a crucial point – while several proposals to promote free software in governments in the region have been made, so far they have only been passed at state level in Brazil. Success in other jurisdictions, however, is likely to follow in time due to the commitment and persistence of the aforementioned groups. An outstanding issue which also needs to be addressed in Latin America and, indeed the world over,

is how best to legislate so that the licensing schemes are protected or preferably enshrined in statutory form.

#### *A history of software development*

Software development began principally in the US several decades ago and it was not initially a subject of intellectual property protection as the value in computers was in the hardware. The US telephone monopoly, AT&T developed the Unix operating system, which was rewritten in C some time later, making it portable. Around the same time, the Defense Department released the ARPANet to communities of computer hackers<sup>ii</sup> and Unix became the operating system of choice at many academic institutions in the US. Networks grew up among hacker communities and this facilitated code sharing and an expansion of the software commons.

#### *Copyright coverage of computer programs*

In 1980, after a long study, copyright laws were extended to cover software in the US. This prohibited the practice of software sharing and prevented access to the source code, outlawing it as “piracy”. The business of software piracy prevention involved lobbying governments and penalising countries, whose laws did not conform with those of the US, also squashing any attempts to introduce *sui generis* intellectual property regimes which had been considered in Australia, for instance. Even when countries complied by extending their copyright laws, rigid enforcement measures were also expected to be put in place.

Claims that piracy cost the software industry billions of dollars never took into account the fact that software was developed in research centres, largely funded by tax payers’ money and was therefore a public resource in the first place. Neither does this

calculation count the unpaid work which hackers undertook in their spare time, which did not gain them any intellectual property windfalls. Proprietary software companies benefit from the network externalities, which occur when a program is pirated. An obvious but often ignored economic plus of piracy is that when upgrades or manuals are needed, they tend to be purchased from the manufacturer of the pirated software, thus providing an unexpected source of revenue, free of marketing costs.

Nonetheless, proprietary software licensing creates a very restrictive intellectual property rights regime, often denying rights normally granted under copyright law.

*Some software remained free*

Free versions of many software programs survived, an example being offshoots of Unix developed at the University of California at Berkeley. The growth of the Internet facilitated a new model of software development in which it could be built collaboratively, fixed, enhanced and made more secure. Yet, the intellectual property rights regime of licensing twinned with copyright prohibited such activities. The restrictions led more software developers to seek free alternatives to proprietary software. This resulted in the proliferation of programs such as GNU software, the Apache Web server, Linux and other free or open source programs such as BIND and Sendmail, to name but a few.

Free Unix has long been used in the Argentinian state sector and the ongoing interest in open systems has clearly continued.<sup>iii</sup>

*“Free” or “open source” software?*

Hackers involved in the development of non-proprietary software are not a homogeneous group. Those with an ideological approach towards this phenomenon like to refer to this software as “free”. This refers to the freedom to copy, modify and share the source code rather than to cost. In fact, many software companies such as Red Hat sell packaged versions of free software and make most of their profits from associated services.

The best-known proponent of free software is Richard Stallman, founder of the Free Software Foundation (FSF), creator of much of its software and author of the most commonly used free software licence, the GNU General Public Licence (GNU GPL).<sup>iv</sup>

Free software is often referred to by an alternative name of “open source” and open source hackers are commercially, rather than ideologically, driven. Many of them believed that the ambiguity of the term “free” in English would deter commercial interest in this software when it began to take hold in the late 1990s. Some of the most notable members of this camp include Eric Raymond and Linus Torvalds. The best known open source software is Linux, or GNU/Linux, which more accurately reflects the marriage or cohabitation between the Linux kernel and GNU software included in the operating system. GNU/Linux has overshadowed an earlier moniker proposed by the FSF: *Linux*.

The European Union, a keen user of free software recently suggested drawing these two banners together under one heading: “libre” software. The word “libre” means “free” as in freedom, rather than price in both Spanish and French but this new name has led to a burgeoning terminology for this software, the most recent denomination being “FLOSS”, which stands for Free/Libre/Open Source Software.

The clash of ideological and mercantile motives, whilst creating different licensing parameters does not prevent the use of Richard Stallman's GNU GPL in the majority of free and open source software projects. GNU/Linux for instance, is licensed under the GNU GPL.

### *Different licensing styles*

The FSF sanctions a number of licences<sup>v</sup>, many of which are also approved by guidelines drawn up by the Open Source Initiative (OSI)<sup>vi</sup>, which is akin to an open source version of the FSF. Some OSI approved licences fall outside the ambit of those approved by the FSF: this is due to the fact that many of their terms are less hostile, or more friendly towards proprietary software.

The FSF divides its licences into a number of categories: the preferred model being a "copyleft" licence. Copyleft, although often jokingly described as "reversing all rights", is actually a combination of copyright and a licensing scheme which obliges users of the software to contribute the source code of their published modifications back to the free software commons. If the licence is violated, the author's copyright is relied upon to hold the offender accountable. In theory, this works in the same way as proprietary licensing except rather than reserving more rights for the copyright holder, it guards fewer, provided that users are prepared to release their published modifications under the same licensing terms. Copyleft thus subverts copyright.

Other licences approved by the FSF include the modified Berkeley Software Distribution (BSD) licence, which is a more recent version of the original BSD Unix

licence, and the Apache licence. These licences, not being copyleft licences, do not oblige the user to give anything back.

The FSF also has a category of unapproved, “non-free” licences, deemed to restrict freedoms unduly.

OSI-sanctioned licences allow programmers to certify their software as “OSI Certified Open Source Software”. OSI has also approved the GNU GPL, as well as BSD licences and over 40 free and open source software licences now exist. New licences tend to get written to suit a particular software developmental model and differences between licences are often miniscule.

BSD style licences, although recognising the contribution of the developer who has gone before, allows the newcomer to do what he or she pleases with the software. Modifications can be copyrighted and this creates the problem of multiple, mutually incompatible versions of software, such as proprietary Unixes, or proprietary applications built on free Unixes. It also allows the imposition of restrictions on access, potentially rendering the free commons obsolete.

The GNU GPL is the most popular and also the most controversial free software licence, given that other licences do not seek to restrict to the same extent the actions of free software users. It stipulates that you may use and copy the software as you choose. When you modify software released under this licence and publish the modifications, you may not subsequently privatise these modifications: instead, you must make the source code available. Furthermore, if you combine a proprietary program with free software, the entire program must be released under the GPL.

### *Legal and quasi-legal facets of the GNU GPL*

The GNU GPL's legality may be dubious in certain jurisdictions. Several sections, such as those dealing with an absence of warranties<sup>vii</sup> or endeavouring to prevent users from enforcing their copyright by obliging them to surrender the source code of their modifications. For instance, under Section 14 of the German Copyright Act, copyright holders would retain the right of integrity to their code.<sup>viii</sup> The FSF's request that software authors surrender their copyright to the organisation may also infringe intellectual property principles in civil law countries. GNU/Linux's reliance on the assertion of individual coders' copyright is unlikely to fare any better in common law jurisdictions due to its difficulty in dealing with the apportionment of copyright in collaborative works. The unauthorised incorporation of free software into a proprietary program may be defensible under copyright law on grounds of fair use, depending on the amount of code used,<sup>ix</sup> although this would violate the licence.

The GNU GPL does not seek to address the issue of proprietary code which has inadvertently been included in a free software distribution and rightly so – this would be outside its scope. However, any free software legislation should endeavour to clarify this issue and absolve any inadvertent infringers from liability in this regard.

However, at present, such points are purely speculative for the GNU GPL has yet to make its court debut. While its ethos appears fair - if you use the software, you abide by the conditions or, in any case, are bound by copyright - courts, accustomed to enforcing increasingly restrictive copyright laws may not look too favourably on what it seeks to achieve. Absent a proper grounding on the complex economics of free

software ecosystems and the established customary practices enshrined in the GNU GPL, judges may dismiss the licence out of hand.

Given the global dimension of software development and usage, the GNU GPL could make its legal inauguration in any jurisdiction around the globe. Certain countries may come under pressure to decide the outcome of a challenge to the GNU GPL in a particular way. Given the politics presently involved in the proposed adoption of free software by many government institutions worldwide and the fierce battle being waged on a number of different fronts by proprietary software companies, a court test would undoubtedly be undesirable and for the free software community, extremely unwise<sup>x</sup>. If not upheld, the effects on the free software community could be devastating, both in terms of the virtual land grab by proprietary software companies, which would ensue and at a human level, on hacker morale.

As it happens, the GNU GPL's enforcement mechanisms have not followed a strictly legal trajectory to date. The licence itself is primarily a customary law and secondly a software licence. Its means of enforcement relies on its moral force backed up by social pressure from the hacker community which developed the free software commons. If a violation of the licence occurs, reports can be made to the FSF via their website at [license-violation@gnu.org](mailto:license-violation@gnu.org). The offender is contacted and advised to desist: most do so at this point. For those who do not, heightened pressure is brought to bear, involving flaming and negative publicity. Being portrayed as a thief of free software is not a particularly attractive proposition.

No offender has yet forced a court hearing but if one does occur, and the licence is not upheld, licence violations may occur on a grand scale, severely affecting the

motivation of those who have chosen to license their programs under the GPL. Furthermore, the “theft” of the communal-style intellectual property, if sanctioned by a court, would be a psychological plus for the “thief” as he or she would be seen to have the weight of the law behind them. Possible outcomes of such a scenario would involve a return to BSD style licensing for some free software but software such as GNU/Linux was not developed under this model and benefited enormously from the contributions of source code, which it received under the GNU GPL. Any failure of the GNU GPL to bind its users could spell GNU/Linux’s and other free software programs’ demise and I believe that this is a matter which lawyers in Argentina must address without delay.

#### *Why appropriate legislation matters*

There has been a growing interest in free software throughout the world over the past few years. Studies, plans for the adoption of free software by public bodies and conversions from proprietary to free software are now all commonplace. Various legislative measures are being considered around the world. A particularly radical bill was introduced in Peru towards the end of 2001 by Congressman, Villanueva Nuñez.

The proposed legislation would have mandated the use of free software by public bodies and the use of proprietary software would only have been allowed if no free software alternative were available. Permission to use proprietary software was to be granted for a maximum of two years at a time, only renewable if no free software alternative were available for that particular task.<sup>xi</sup> The stated justification for this legislative move was economic and relied on Article 14 of the Constitution, which indicates that the State is obliged to encourage Peru’s technological progress.<sup>xii</sup>

However, a hefty donation from Microsoft has scuppered this proposition for now.

The difficulty of what is actually meant by different jurisdictions around the world when they talk about free, open source, libre software or, indeed, FLOSS is in urgent need of a clear and concise legal definition. Do they mean BSD licensed software and, if so, which BSD licence? Are they referring to software released under the GNU GPL or, indeed, any of the other approximately 40 free software licences in existence? The involvement of state agencies in the use or development of this software does not change the fact that at any time, any individual anywhere in the world may violate the licence and threaten the continuation of free software licensed in this way.

Little is mentioned about licences by these agencies but presumably, they endeavour to conform with the licensing provisions of the software they are using. The use of software subject to BSD style licences is largely uncontroversial, given that there are very few terms to violate as users have almost unlimited freedom. However, as far as software subject to the GNU GPL is concerned, the licence may easily be violated and public bodies face two difficulties in this respect. First, they should avoid violating the licence themselves in order not to become the subject of flaming and negative publicity and second, they should endeavour to ensure that the software they produce remains free by enforcing compliance with the licence. This would appear to require that some sort of legislative framework is put in place to enshrine hackers' customary practices in the law. Such a legislative move would not be without precedent: the Law Merchant or *Lex Mercatoria*, the precursor of modern commercial law, grew out of the customs adopted by merchants at trade fairs in Europe in the Middle Ages.

*Legislating free software: A Lex Lignux?<sup>xiii</sup>*

Customary norms have ensured compliance with the GNU GPL for over a decade.

Given that the GNU GPL is used in a very great number of free software projects, any legislation should be based, at least loosely, on this.

Countries considering the adoption of free software, rather than first focusing on obliging the use of free over proprietary software by public bodies, need to ensure the survival of this software development model by protecting a functioning if somewhat unconventional *quasi* legal regime. This would be an entirely separate endeavour from mandating its use.

The creation of some type of *Lex Lignux* should follow fast on the heels of adoption of free software. It could take the form of a *sui generis* law and it would be imperative that uncertainties and areas of doubt that exist in the licence are ironed out. This would not have to disturb copyright laws in any way and has a precedent. Under EU law databases, for instance, are protected for 15 years only: a much shorter time than their computer program counterparts.

A group of Brazilian lawyers involved with the Free Software Consortium which is based in Costa Rica are currently studying making the GNU GPL compatible with their copyright law. We are exchanging data on that at the moment. I am also a member of the Free Software Consortium and I sit on its Ethical Committee and coordinate its Cyber Tribunal. We are currently involved in a project to draft a law which will be put before the Costa Rican Congress, endeavouring to protect free software true to the spirit of the GNU GPL. In order to achieve this, I have to deepen

my knowledge of the civil law system of copyright and that is one of my many reasons for visiting Latin America. I hope that my trip will be productive.

### *Conclusion*

The most popular free software licence has effectively bound the hacker community and outsiders for many years, creating a virtual space where generosity is compulsory. Its ethos has also spread to areas outside of software. Free software's importance has increased with the interest and participation of external agents, such as government agencies and commercial firms. While enhancing the reputation of this software these new bodies could also pose a threat: if the licence is not upheld a free software free-for-all could ensue. It is vital, therefore, to draw up a legislative measure which obliges respect for this regime, if the benefits of free software development are to be preserved. The most effective way in which this could be carried out would be to put in place a legislative framework, which resolves any legal uncertainties. It should also aim to enshrine hacker customs in the law so that if tested in court, judges will have very little leeway to misinterpret or reinterpret the terms under which free software is distributed.

- <sup>i</sup> Scheeres, J., “Argentina Mulls Open-Source Move,” *Wired News* <http://www.wired.com/news/business/0,1367,43529,00.html> (accessed May 4, 2003).
- <sup>ii</sup> The term “hacker” is used in the sense of “computer programmer” throughout this paper.
- <sup>iii</sup> “Ambito de Software Libre en el Estado Nacional – Declarado de Interes Nacional,” <http://www.softwarelibre.gov.ar> (accessed May 20, 2003).
- <sup>iv</sup> A comprehensive legal analysis of the GNU GPL has been carried out and can be accessed at O’Sullivan, M., “Making Copyright Ambidextrous: An Expose of Copyleft”, *The Journal of Information, Law and Technology (JILT)* 2002 (3) <http://elj.warwick.ac.uk/jilt/-2-3/osullivan.html>.
- <sup>v</sup> Free software licences can be viewed at [www.gnu.org/licenses/license-list.html](http://www.gnu.org/licenses/license-list.html).
- <sup>vi</sup> Open source software licences are to be found at [www.opensource.org/licenses](http://www.opensource.org/licenses).
- <sup>vii</sup> In Germany, disclaimers of warranties would not be upheld, especially if the software were for sale: <http://www.gnu.org/brave-gnu-world/issue-15.en.htm> (accessed March 10, 2003). This may also fall foul of the US Uniform Computer Information Transaction Act (UCITA).
- <sup>viii</sup> Metzger, A. and Jaeger, T., “Open Source Software and German Copyright Law”, [http://www.ifross.de/ifross\\_html/publikation.html](http://www.ifross.de/ifross_html/publikation.html) (accessed November 24, 2002).
- <sup>ix</sup> McGowan, D., “Legal Implications of Open Source Software”, *Social Science Research Network Electronic Paper Collection*: [http://papers.ssrn.com/paper.taf?abstract\\_id=243237](http://papers.ssrn.com/paper.taf?abstract_id=243237) (accessed February 20, 2003).
- <sup>x</sup> Some hackers would, apparently, relish the challenge.
- <sup>xi</sup> Proyecto de Ley Nro. 1609, 2001.
- <sup>xii</sup> <http://www.asesor.com.pe/teleley/biblioteca/constitucional/5000b.htm> (accessed May 26, 2003).
- <sup>xiii</sup> “Lignux” was an alternative name once proposed by the FSF for Linux: the intention was to incorporate the word “GNU” into Linux in some manner. The author believes this to be a more elegant legal term for any proposed “lex”: *Lex GNU/Linux* would surely be a bit too much of a mouthful.