Economics of Litigation related to the GNU General Public License (GPL)

Daniel Watrous

Business Law

Northwest Nazarene University

Economics of Litigation related to the GNU General Public License (GPL)

Free Software Foundation

In 1983, Richard Stallman set out to create a UNIX-like operating system that would be given away for free. That novel idea, to create high quality software to give away for free, would lead to the creation of the Free Software Foundation (Stallman, 2013) two years later in 1985. Since then, and especially from the mid 1990s, he has advocated for free software and opposed political measures that would limit free software, such as software patents and extension of copyright laws.

One byproduct of the open source software, or free software market is that its economy exists somewhat apart from the traditional market for software. Software products generally have a high perceived value. Software developers (programmers) are typically highly paid, highly skilled workers. In it's original form, software is written in a language as plain text (also called source code), which is later transformed so that it can run on specific computer systems. It's not necessary for a consumer of software to know about programming languages or possess the source code in order to use the software, but someone with the source code can change the way a program works and create a new runnable version of the software.

Source code benefits from copyright protection under the law. Companies that produce software in order to sell it to a consumer typically keep their source code secret. This is referred to as closed source. One reason for this is that anyone with the source code could make any changes they like, thereby avoiding the need to pay the company for a new version. Naturally, a company with a commercial software product will want to protect it as much as possible.

The advent of open source software has changed the economics of the software industry. Much like television didn't kill radio, open source software has not killed commercial software. The advent of television did, however, change the primary social function and the character of radio. In a similar way, free software has changed the

character of commercial software.

What Richard Stallman proposed in 1983, and what he has subsequently achieved by establishing the Free Software Foundation and advocating for free software, is quite radical. He claims to have created or facilitated a majority of the Linux operating system, with many ancillary components being associated with his organization. The Linux kernel (the core of the Linux operating system) was created by Linus Torvalds and is released for free as open source software. A primary mechanism driving legal considerations related to this free software is a license known as the GNU General Public License, or GPL. The FSF claims that the GPL is the most widely used open source license among open source software.

GNU General Public License, or GPL

The GPL is a contract (Cheeseman, 2013) that binds the recipient of software to abide by certain terms in order to use it. Those terms grant broad privileges, among which are the right to use, copy, distribute and modify the software. All of these privileges come at no cost to the consumer, but do carry the stipulation that the exercise of these privileges must also be extended to other recipients too. The license points out that "When we speak of free software, we are referring to freedom, not price", later on stating that when you distribute it to others you may "charge for them if you wish", provided that you make them aware of the license and their freedoms under the same.

Section 5 of the GPL covers "Conveying Modified Source Versions". It is in this section that derivative works are bound by the GPL. In particular, subsection c, which reads as follows:

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.

This is often referred to as infection. This means that software which integrates with or accesses (modifies) existing GPL software is infected by the GPL and must therefore extend to others the same broad privileges that came with the original software, even though those contributions were made by a later author.

One common misconception about the GNU GPL is that software so licensed becomes part of the public domain. This is not the case. In its preamble, the GPL states that one of the ways in which developers who choose the GPL protect your rights is to "assert copyright on the software". In the How to Apply These Terms to Your New Programs section, developers are instructed, among other things, to include the copyright notice "Copyright (C) <year> <name of author>". Copyright laws and protections (copyright.gov, 2013) apply to software released under the GPL.

If someone does make changes to GPL licensed software, the GPL does not require him to distribute his changes. However, if he does distribute them, he is required to include a copy of the GPL explaining that all the software so distributed, including his modifications, are bound by that license.

Variations of the licenses

There have emerged a few significant variations of the GPL. Two such examples are the Lesser General Public License (LGPL) and the Affero General Public License (AGPL). Both variations were commercially motivated.

LGPL.   The infection referred to above represents an barrier to use for many commercial software developers who are not willing to infect their software with the GPL. To get around this, the Free Software Foundation introduced the LGPL which, as they state it "incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below". One significant

modification of the terms is that you may convey certain versions of the GPL software "under terms of your choice", making it more commercially friendly.

In other words, software that simply links to LGPL is not required to fall under either the GPL or LGPL. Any changes to the LGPL software itself are still bound by the LGPL.

The obvious aim of this variation of the GPL is to enable commercial software developers to leverage LGPL licensed software without losing the ability to keep the source code of their software closed. While the FSF does make this license available, it's is still quite invasive when compared to other more liberal licenses. The FSF has even published an argument against using the LGPL, hoping to force more software developers, even commercial developers, to release their software under the GPL.

AGPL.   Software is increasingly delivered as a service, which means we don't download the software, we simply access it over a network. Since the GPL deals primarily with distribution, a person could modify and run GPL licensed software on his own server and provide access to that software without being required to distribute his modified source code.

The AGPL was introduced to address this and includes provisions requiring modified versions accessed over a network to be made available to anyone that accesses it. Here is some of the relevant wording from section 13.

> "your modified version must prominently offer all users interacting with it remotely through a computer network (if your version supports such interaction) an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge"

Dual licensing

Some software developers will release software under a dual license. In other words, they will release the same source code under different licenses. For those that wish to use it

freely, they will provide it under the terms of the GPL. However, they reserve the right to also sell licenses to others to include the software in closed source software.

The GPL doesn't prevent the author and copyright holder from pursuing this dual licensing, although it does prevent him from rescinding the rights of others under the GPL for software already distributed under that license. The GPL does not require an author and copyright holder to release future versions under the same license. While non-copyright holders must abide the terms of the GPL, the copyright holder is free to issue his copyrighted work in any way he chooses.

Alternatives to the GPL

The GPL is only one of many open source licenses under which software is published. Many other licenses are considered more commercially friendly. In general these impose fewer restrictions on the modification and distribution of software, including for commercial purposes. Some are very simple, like the MIT and BSD licenses, while others are more involved, like the Apache or Mozilla licenses.

The Open Source Initiative (OSI, 2013) provides resources and copies of popular open source licenses. Enterprises that are evaluating open source softwares should familiarize themselves with the details of the various open source licenses. This is especially true for software vendors that intend to sell a product that may incorporate open source components.

Economics of Open Source Software

Open source software can be viewed as a type of Software Cooperative. In that capacity there are both community benefits and commercial objectives. The economy of open source software has its roots in a type of barter system. For example, Linux is a combination of hundreds of different pieces of software. Each separate piece of software has one or more contributors. The contribution of one developer is synergistic with the contribution of another, and so there is mutual benefit.

It's like a barter system in the sense that the developers are trading in lines of software, not cash for software. As a result, many open source licenses aim to secure the freedom of the software so licensed and encourage continued bartering. In other words, they effectively communicate "If you contribute to this software, all of your work will be available to you and to others. When others contribute, you'll get their changes too".

As more contributors come forward, the quality and functionality of the software increases. This in turn increases the benefit and the motivation to both consume and contribute to available open source software. The economy grows. Today there are many thousands of open source programs. Not only are some open source software projects the highest quality, but they also power and make possible significant amounts of traditional commerce, such as Internet based commerce.

Commercial Applications of Open Source Software

Software is inherently a high value product. It is expensive to produce and has a high perceived value from the consumer. The fact that open source software was created within the barter-centric economy described above doesn't reduce it's perceived value due to the fact that the alternative is often to pay a high fee for commercial software to fill the same purpose.

As a result, many have tried to find a way to market open source software. Some popular approaches to this include packaging several pieces of software together, (re)branding the open source software with a commercially catchy name and logo or simply selling support.

The GPL places some obvious legal restrictions on packaging and distribution as well as branding efforts. As a result, when it comes to GPL licensed software, the most common commercial model is referred to as Professional Open Source. In this model the software is still available for free under the terms of the GPL, but a company with expertise in the software will provide an option to purchase a support contract.

The support contract may include custom integrations, troubleshooting, installation, training, etc. It may also include additional components or services that augment the functionality of the GPL software, but don't link directly to it. In some cases these contracts can be quite expensive.

Social Impact

Opinions about the GPL can be sharply divided. For those participating in the bartering system and who derive direct benefit from the GPL, they are mostly in favor of it. For those that are interested in making money directly from GPL software or wish to leverage GPL software to round out their own commercial software offering, they can be very vocally against it. There are even those who argue that the existence of GPL software interferes with commerce by reducing sales of their existing commercial solutions (see page 11) (Gatto, 2007).

One of the most common complaints about the GPL is that it infects any source code that it touches. In the case of the Affero GPL this infection even affects consumers of the software that do not distribute it. Some enterprises worry about how interactions with other commercial software throughout their organization may impact their obligations under the GPL. As a result, some businesses opt to avoid any association with the GPL which is not absolutely necessary.

Some software developers have released their software under multiple licenses, including variations of the GPL and other open source licenses (such as Apache) to reassure enterprise customers that other software in the enterprise won't be affected adversely by the GPL (Inc., 2013).

Compliance with the GPL can be akward and is often overlooked or ignored. The justification some use to overlook or ignore the requirements of the GPL comes down to enforceability.

Enforceability

For many years there has been an ongoing debate about whether the GPL was enforceable in the courts. One of the subtle issues at play is whether it's legal for the GPL to retain licensing control over the contributions of subsequent developers. In other words, secondary contributors may dislike the idea that their contribution is bound by the GPL. Perhaps they would favor a different license for their own contribution, but the GPL claims that any derivative work must be bound by the terms of the GPL.

Another argument relates to linking. When someone else creates an independent and separate software program and simply links it to GPL software, they argue that it isn't fair that their separate software should become bound by the GPL simply because it leverages GPL software.

These issues and others have been considered in court proceedings, but there is very little case law and almost no precedents as a result. In general, there is very little money directly associated with GPL software, which reduces the motivation and resources to pursue remedy through the courts.

Insufficient Funds

As described above, the economy open source software is more of a barter system than a monetary one. The purveyors of open source software are often individuals or small organizations. As a result, they infrequently have retained legal counsel, let alone sufficient funds to pursue legal action against someone who violates the terms of the GPL. Even if they did have resources to pursue legal action, the outcome would be to enforce the GPL, which is free distribution of the software so licensed.

Any court win in favor of the GPL would be a financial loss for the person who had retained copyright and distributed his software using the license. It may be possible to seek damages from the person violating the GPL to recoup legal fees, but that still makes it a zero sum outcome. Needless to say, that's not a very popular position for many lawyers or

software creators who use the GPL.

Litigation

There have been a few instances where there was enough financial motivation to pursue litigation. The cases below deal with some of the issues shown above.

PROGRESS SOFTWARE CORPORATION, et al., Plaintiffs, v. MySQL AB, et al., Defendants.

In this case, Progress Software Corporation (hereafter Progress) was seeking legal remedy to protect the market value of a software product that integrated with MySQL. MySQL, who owns the registered trademark MySQL ^{TM}, requested a preliminary injunction against Progress for, among other things, sublicensing or distributing MySQL's program and from using the the MySQL trademark.

The trademark issue was rather straight forward, and the plaintiff was enjoined from using the owner's mark. The motion was denied with respect to enjoining Progress from distributing its Gemini software. Among the rationale used by the court was that "With respect to the General Public License ("GPL"), MySQL has not demonstrated a substantial likelihood of success on the merits or irreparable harm." This due to the fact that there was "factual dispute concerning whether the Gemini [**4] program is a derivative or an independent and separate work under GPL".

If it had been a derivative work, the language of the GPL might be interpreted to curtail distribution for commercial purposes, while if it was found to be separate an independent, distribution rights may have gone the other way. The balance of harms tipped in favor of Progress in this case and the Judge chose to deny the injunction request.

This dispute was settled by the parties before going to trial. As a result, this case did not establish precedent and provides no definitive resolution as to the enforceability of the GPL.

DANIEL WALLACE, Plaintiff, v. FREE SOFTWARE FOUNDATION, INC., Defendant.

In Wallace v. Free Software Foundation (FSF), Daniel Wallace accused the FSF of conspiring with various software developers and software organizations to "pool and cross-license their copyrighted intellectual property in a predatory price fixing scheme in violation of §1 of the Sherman Act, 15 U.S.C.S. §1". In other words, he observed that the outcome of the open source community was the release of a free operating system that effectively undercut his market, and so appeared to fix the price for operating system software across the market. This he blamed on the FSF by means of the GPL.

This case was also dismissed without reaching trial on the basis that Daniel Wallace was unable to show in his complaint that there was injury to competition as a whole. In his opinion, the judge observed that, contrary to Wallace's claim that the GPL was foreclosing competition in the market, it was actually increasing the amount of competition in the market, some of which was free of charge.

The court further articulated its understanding of the GPL by stating that it merely "acts as a means by which certain software may be copied, modified and redistributed without violating the software's copyright protection". While this case still does not establish a precedent, it does contain an opinion which seems to give credence to the GPLs purported purpose.

PLANETARY MOTION, INC., Plaintiff-Counter-Defendant-Appellee, v. TECHPLOSION, INC., Michael Gay a.k.a. Michael Carson, Defendants-Counter-Claimants-Appellants.

In this case we hear an argument that software released under the GPL waives any rights to trademark. The court's interpretation of the GPL did not support that conclusion.

Software distributed without charge to users pursuant to a GNU General Public License is not necessarily ceded to the public domain, and the licensor

purports to retain ownership rights, which may or may not include rights to a mark.

The court thus held that distribution under the GPL was not tantamount to surrendering ownership, copyright or present or future rights to trademark.

Conclusion

Open source licenses, such as the GNU General Public License (GPL), have created a barter-like economy, where high quality software can be created without requiring direct monetary investment. The GPL, in particular, asserts that software so licensed and distributed may be used, modified and redistributed, so long as it is done without charge and secures copyright protection to the original author.

There is little case law or established legal precedent related to the GPL. This is due in part to the high cost of litigation and the void of financial value present with much open source software. Court issued opinions in cases that do involve the GPL acknowledge the primary objectives of the GPL. These include copyright and other intellectual property ownership rights and distribution rights. Another opinion suggests that the GPL is not in conflict with key antitrust laws.

For the time being, the open source community, and particularly those who license their software under the GPL, may proceed with confidence that they can release the software they produce, for free, without giving up important copyright protection under the law. They may even find that they can sell ancillary services related to GPL software, even if it was released by another party, provided that they do so within the limits established by the GPL.

References

Cheeseman, H. (2013). *Business law* (Eighth Edition ed.). unknown: Pearson.

copyright.gov, U. S. (2013). *United states copyright office.* Available from
    `http://www.copyright.gov/`

Foundation, F. S. (2013). *General public license (gpl).* Available from
    `http://www.gnu.org/licenses/gpl.html`

Gatto, J. G. (2007). Doubts wane over gpl enforceability. *Managing Intellectual Property.*
    Available from
    `http://www.pillsburylaw.com/siteFiles/Publications/A9A22185D029BBE6EAA4332F1A7249E2`

Inc. 10gen. (2013). *Mongodb licensing.* Available from
    `http://www.mongodb.org/about/licensing/`

John Daniel Tinder, J. (2006). *Daniel wallace, plaintiff, v. free software foundation, inc.,*
    defendant. Available from `http://tinyurl.com/wallacevfsf`

OSI. (2013). *Open source initiative.* Available from `http://opensource.org/`

Patti B. Saris, U. S. D. J. (2002). *Progress software corporation, et al., plaintiffs, v. mysql*
    ab, et al., defendants. Available from `http://tinyurl.com/progressvmysql`

Restani, J. A. (2001). *Planetary motion, inc., plaintiff-counter-defendant-appellee, v.*
    techplosion, inc., michael gay a.k.a. michael carson,
    defendants-counter-claimants-appellants. Available from
    `http://tinyurl.com/planetaryvtechplosion`

Stallman, R. M. (2013). *Free software foundation.* Available from `http://www.fsf.org/`