Encyclopedia of Virtual Communities and Technologies

Subhasish Dasgupta
George Washington University, USA

Intellectual Property Rights in Open Source Software Communities

Chitu Okoli
Concordia University, Canada

Kevin Carillo
Concordia University, Canada

INTRODUCTION

Intellectual property is an old concept, with the first recorded instances of patents (1449) and copyrights (1662) both occurring in England ("Intellectual property", Wikipedia, 2004). The first piece of software was submitted for copyright to the United States Copyright Office in 1961, and was accepted as copyrightable under existing copyright law (Hollaar, 2002).

The open source movement has relied upon controversial intellectual property rights that are rooted in the overall history of software development (Lerner & Tirole, 2002; von Hippel & von Krogh, 2003). By defining specific legal mechanisms and designing various software licenses, the open source phenomenon has successfully proposed an alternative software development model whose approach to the concept of intellectual property is quite different from that taken by traditional proprietary software.

A separate article in this encyclopedia treats open source software communities in general as a type of virtual community. This article takes a historical approach to examining how the intellectual property rights that have protected free/open source software have contributed towards the formation and evolution of virtual communities whose central focus is software projects based on the open source model.

TOWARD THE OPEN SOURCE CONCEPT

With the advent of computer technology since the 1960s, communities of computer programmers began to form, starting with the MIT Artificial Intelligence Lab (AIL) in 1959. These "hackers", as they called themselves, had a common culture that valued creative software solutions to various scientific and computing problems, and enshrined a culture where sharing ideas and programming code was expected and valued, with the most capable and inventive programmers receiving high respect.

However, things began to change in the early 1980s as computers became more ubiquitous, as physical sizes shrank and prices dropped while computing power simultaneously increased dramatically. Computing-based enterprises and even not-for-profit shops such as AIL began to realize the commercial value of software, and they started to enforce their copyrights and began to restrict sharing of software code strictly to their own organizations. Richard Stallman, a hacker at MIT's AIL, opposed these moves to no avail. He finally quit in 1984 in protest against the restrictions on sharing among computer programmers, which he considered inimical to the hacker culture. He founded the Free Software Foundation (FSF) and with legal consultation created the concept of the "copyleft", proclaimed in the GNU Manifesto (FSF, 1985) and legally enshrined in 1989 in the GPL (FSF, 1991).

Copyleft as expressed by the GPL has had a critical effect on shaping the very existence of open source software virtual communities. Open source software uses copyright law to preserve certain freedoms (hence the name, "free software") regarding the creation, modification, and sharing of software. Specifically, all open source software grants users the following key rights:

1. **The right to full access to the source code:** When a computer programmer sees how a piece of software actually works, as specified in the source code, they can fully understand the inner workings and can intelligently modify the software as they deem appropriate.

2. **The right for anyone to run the program for any purpose without restriction:** There are no restrictions against commercial, military, foreign, or any other use, and discrimination against users for any reason is expressly forbidden.

3. **The right to modify the source code:** This includes absorbing the software, in whole or in part, into other pieces of software created by other developers.

4. **The right to distribute both the original software and the modified software:** A key difference be-

tween "free software" and "freeware" is that while freeware generally permits and encourages free distribution of the software, it does not permit sale of the distributed software beyond reasonable distribution costs.

5. **The right to know about their open source rights:** The open source license must be prominently displayed and distributed to users, so that they are aware of their rights (including access to the source code).

The GPL, the first legal document to license open source software, grants users and developers these rights with the intention that developers would modify the software and share it with others with similar liberality. This is a distinct concept beyond simple "open source" that the FSF calls "copyleft". To guarantee this goal, the GPL grants the privileges mentioned above as long as a key condition is observed: *The obligation to distribute derivatives under copyleft.* Any software modified under the GPL can be redistributed for sale, but it must be licensed under a copyleft license; that is, modified derivative works must also be made available under an open source license. While it does not have to be licensed under the GPL itself, the chosen license may not restrict any of the five rights listed above.

These copyleft terms are critical to the very existence of OSS virtual communities. When Richard Stallman posted his manifesto and invited software developers to join him in his crusade for free software, there was no lack of sympathetic and willing hackers who wanted a return to the days of free sharing. However, there was a grave concern that, corporate interests could easily take these programs, add their proprietary extensions, and withdraw the software from public access. With its copyleft mechanism, the GPL guaranteed that any person or corporation who wanted to benefit from the liberal efforts of computer programmers would be legally bound to share their work in the same spirit of camaraderie. Considering the climate in which the free software movement was founded, it is unlikely that the movement could have gotten off the ground without such a radical clarion call to mobilize devoted followers in the first place.

IMPORTANT OPEN SOURCE SOFTWARE LICENSES, AND THEIR EFFECTS ON OPEN SOURCE SOFTWARE COMMUNITY LIFE

As detailed earlier, the GNU GPL was the first open source software license, and with its strong copyleft provisions, it enabled the possibility of open source software commu-

nities to form. One particularly strong feature of the GPL is its requirement that not only must derivates of licensed software be copylefted (that is, made available under GPL-like terms), but all software programmatically linked together with GPL-licensed software must also be copylefted. This requirement, inspired by the Free Software Foundation's stated goal of eventually ridding the world of proprietary software, has been widely considered excessive. In fact, no other organization has issued such restrictive open source software licenses. However, in spite of its strictness, the GPL remains the most popular licenses for open source software.

Based largely on the GPL, open source development communities such as SourceForge.net have flourished, protected by open source licenses that permit free creation and sharing of open source software. The most important addition to the GPL camp was Linux, which provided a long-sought kernel for the operating system being built by the GNU Project and that has now being proven to be powerful, fast, efficient, stable, reliable, and scalable (Edwards, 1998).

Loosening Up: Open Source Becomes More Commercial

In the 1990s, largely resulting from the phenomenal success of Linux, many of the organizations who had gradually commercialized their software in the 1970s and 80s came to appreciate the quality and quantity of work that could be done with their software when released to open source communities under the protection of appropriate licensing structures (West & Dedrick, 2001). However, few of these organizations felt comfortable with according rights as broad-sweeping as the GPL, and so gradually a wide variety of licenses were developed as various large software developers, both commercial and academic, began to experiment with releasing their source code for free development. These licenses avoided imposing the requirement of sharing such software under such rules; that is, they generally permitted developers to make proprietary derivatives from the selected source code they released.

Although the University of California already widely licensed their proprietary version of Unix, the Berkeley Software Distribution (BSD), they re-licensed it with the open source BSD License in the early 1990s ("Berkeley Software Distribution", Wikipedia, 2004). The BSD license gives users the rights to run programs, to view and modify the source, and to distribute their modifications, including for commercial purposes. However, unlike the GPL, the BSD license does not require licensees to release the modifications by copyleft—they are free to make their modifications proprietary. Popular programs that use this

license include a number of variations of the BSD Unix operating system, the JGraph graphing tool, and the PostgreSQL database management system.

Similarly, MIT released various programs under the simple MIT license in the same period under terms very similar to the BSD license. Programs that use this license include the X Windows Unix graphical user interface and the BitTorrent file downloading system.

In 1991 the Free Software Foundation released the Library GPL (later renamed the "Lesser GPL"), which retains the requirement of derivatives being copylefted, but without imposing the same restriction on programmatically linked software (FSF, 1999). This permits the distribution of dynamically linked libraries that are attached to large pieces of software; in particular, the Lesser GPL permits proprietary software to use open source software modules without having to be entirely released as open source.

Open Source Becomes as Competitive as Commercial Products

The communities formed using these later licenses are different from those that typically use GNU licenses in a few ways. Generally speaking, they do not espouse the free software philosophy as radically as the FSF. The software development under these licenses is generally carried out in smaller, focused projects, and the resulting products are often eventually made proprietary by the single person or company that started the project. Many of these projects can hardly be called "communities", and many of them are not primarily virtual—that is, connected via telecommunications.

However, the success of commercialized open source software demonstrated that this model of software development can create valuable products that can then become proprietary, to the benefit of the corporate founder (Ousterhout, 1999). In 1998, Netscape Corporation released the source code of its ailing Communicator Internet client suite for open source development under the Mozilla project. Mozilla was released under the Netscape Public License (NPL) and the Mozilla Public License (MPL) ("Mozilla", Wikipedia, 2004). These licenses attempted to include a copyleft provision that required modifiers to distribute derivative works under similar licenses, but the copyleft specifications were light in the sense that programmers could use packaging loopholes to distribute proprietary extensions along with NPL/MPL code.

The release of Netscape code was a milestone in that it was the first major attempt for a large corporation to license their core code as free software, with the strategic intention of incorporating the improvements into their commercial products. The development of the Mozilla project was very gradual until 2003 when AOL, who had bought Netscape after its change in strategy in 1998, scaled back their support for the project. However, this led to the formation of the independent Mozilla Foundation that rallied support and steam for the project. In November 2004, Mozilla released the first official version of their Firefox Web browser, widely acclaimed as matching or even superseding the quality of Microsoft's Internet Explorer, the market dominator at that time.

Apple Computer surprisingly followed this model in 2000 when they released the kernal of their Unix-based operating system to the open source community as Darwin 1.0 ("Apple Darwin", Wikipedia, 2004). The original Apple Public Source License (APSL) under which it was released was similar to the Netscape Public License in that it reserved proprietary rights for Apple. In 2002 they helped form the OpenDarwin community, which develops Apple Darwin, the kernal of Apple's flagship Mac OS X. Apple eventually revised the APSL to be fully copyleft, such that it has been approved by the Free Software Foundation, though it is not as strong as the GPL.

The communities formed around these licenses (Netscape/Mozilla and Darwin) are remarkably different from other OSS communities in that they are dedicated to developing products in parallel with commercial products. Netscape Communicator was almost dying, but is being revived by the success of the Mozilla Project. Similarly, the Apple Mac OS X operating system is flourishing as a result of the partnership between commercial sponsors and the supporting OSS communities which has given the community members the opportunity to create high-quality software (Mishra, Prasad, & Raghunathan, 2002; Stamelos, Angelis, Oikonomou, & Bleris, 2002). The different OSS licenses that have been introduced so far and their specific assumptions are presented in Table 1.

The "Open" Concept Extends Beyond Software

There is much more to open source software than just a technical phenomenon and an alternate software development methodology. Open source software projects are virtual communities in which people interact to achieve a common goal (Chengalur-Smith & Sidorova, 2003; Diker & Scholl, 2001; K. Lakhani & von Hippel, 2003; Ljungberg, 2000). Such communities have power structures, community norms, values, and traditions (Bergquist & Ljungberg, 2001). Most open source software communities are somewhat narrow in their scope of contributors, being primarily white males around 30 years old (Ghosh, Glott, Krieger, & Robles, 2002; Hars & Ou, 2001; K. R. Lakhani, Wolf,

Table 1. Major open source software licenses and their associated assumptions

Major open source software licenses			Rights		
Name	Creation Date	Examples of licensed software	Open Source	Copyleft derivatives	Copyleft linked software
GNU GPL	1984	Linux	X	X	X
GNU Free Documentation License	1991	Wikipedia	X	X	X
GNU Lesser GPL	1990	OpenOffice.org	X	X	
MIT License	early 1990s	X Window System	X		
Berkeley Software Distribution License	1998	BSD Unix	X		
Netscape/ Mozilla Public License	1999	Mozilla Project	X	X	
Apple Public Source License	2000	Mac OS X Darwin kernel	X		

Bates, & DiBona, 2002). This corresponds closely to the primary demographic of skilled computer programmers.

Nonetheless, the open source approach has inadvertently provided concepts that are not only restricted to software. In 2000, the FSF created the Free Documentation License, which was designed to license the text documentation that accompanies free software under terms similar to the GPL, only adapted for text content. This new type of license created a legal instrument for the existence of the textual counterpart of open source software, sometimes called "open source" or "open content" text. The primary exponent of this model has been Wikipedia, the free encyclopedia ("free" in the same sense as free software—www.wikipedia.org). The encyclopedia is maintained by a community of over 100,000 contributors—far larger than any open source software community—because anyone is permitted to contribute to articles.

A comparable phenomenon is being created by the Creative Commons (CC, www.creativecommons.org), a resource that creates licenses on demand for literary, audio, and video works—the more traditional media for copyright licensing (CC refers people to the GNU GPL for software licensing). Created in 2002, CC lets creators of content choose among several rights patterns they want to give users of their works; the community aspect arises from their hosting a directory of CC-licensed works on the Internet. Even though the CC community is not very cohesive as a virtual community, it does provide a legal vehicle by which non-software virtual communities that create shared content could license their works.

CONCLUSION

With the history outlined here, open source software communities have established themselves as an important type of virtual community, secured by the legal

framework of open source licenses. To date, out of thousands of software works licensed under the GPL since 1984, we are aware of only one challenge to its legality. As of December 2004, the SCO Group, who currently owns the copyrights to the Unix operating system, is suing IBM for copyright violations on its distribution of Unix-based operating systems, including Linux ("SCO-Linux Controversies", Wikipedia, 2004). Among other allegations, SCO claims that IBM's distribution of Linux violates their copyright because, they claim, the GPL is invalid. This first legal test of the GPL is widely considered frivolous, but as of the writing of this article, this case is still being tried in the United States. If SCO should win and then require Linux distributors to obtain special licenses from them, such a development could seriously hinder the development of this operating system. However, if SCO should lose, as most observers expect, this legal test could serve to boost and incontrovertibly establish the place of this type of virtual community with its important role in the software industry.

REFERENCES

Bergquist, M., & Ljungberg, J. (2001). The power of gifts: Organizing social relationships in open source communities. *Information Systems Journal, 11*(4), 305-320.

Chengalur-Smith, S., & Sidorova, A. (2003, December 14-17). Survival of open-source projects: A population ecology perspective. Paper presented at the *24th International Conference on Information Systems 2003*, Seattle, Washington.

Diker, V.G., & Scholl, H.J. (2001, January 3-6). The art of leveraging: How powerful nonlinear feedback processes can restructure rapidly growing technology and knowledge industries. Paper presented at the *34th Hawaii Inter-*

national Conference on System Sciences, Island of Maui, Hawaii.

Edwards, J. (1998). The changing face of freeware. *IEEE Computer, 31*(10), 11-13.

FSF. (1985, May 17). *The GNU Manifesto.* Retrieved December 2004, from *http://www.gnu.org/gnu/manifesto.html*

FSF. (1991, November 8). *GNU General Public License.* Retrieved December, 2004, from *http://www.gnu.org/copyleft/gpl.html*

FSF. (1999, August 13). *GNU Lesser General Public License.* Retrieved December 2004, from *http://www.gnu.org/licenses/lgpl.html*

Ghosh, R.A., Glott, R., Krieger, B., & Robles, G. (2002). Free/Libre and Open source software: Survey and study. Brussels: *Report of the FLOSS Workshop on Advancing the Research Agenda on Free/Open Source Software,* European Commission.

Hars, A., & Ou, S. (2001, January 3-6). Working for free? Motivations of participating in open source projects. Paper presented at the 34*th* *Hawaii International Conference on System Sciences,* Island of Maui, Hawaii.

Hollaar, L.A. (2002). *Legal protection of digital information.* Washington, DC: BNA Books.

Lakhani, K., & von Hippel, E. (2003). How open source software works: "Free," user-to-user assistance. *Research Policy, 32,* 923-943.

Lakhani, K.R., Wolf, B., Bates, J., & DiBona, C. (2002). *The Boston Consulting Group hacker survey.* Boston Consulting Group and Open Source Developers Network.

Lerner, J., & Tirole, J. (2002). Some simple economics of open source. *Journal of Industrial Economics, 50*(2), 197-234.

Ljungberg, J. (2000). Open source movements as a model for organising. *European Journal of Information Systems, 9*(4), 208-216.

Mishra, B., Prasad, A., & Raghunathan, S. (2002, December 15 - 18). Quality and profits under open source versus closed source. Paper presented at the *23rd International Conference on Information Systems,* Barcelona, Spain.

Ousterhout, J. (1999). Free software needs profit. *Communications of the ACM, 42*(4), 44-45

Stamelos, I., Angelis, L., Oikonomou, A., & Bleris, G.L. (2002). Code quality analysis in open source software development. *Information Systems Journal, 12*(1), 43-60.

von Hippel, E., & von Krogh, G. (2003). Open source software and the "private-collective" innovation model: Issues for organization science. *Organization Science, 14*(2), 209-223.

West, J., & Dedrick, J. (2001, January 3-6). Proprietary vs. open standards in the network era: an examination of the Linux phenomenon. Paper presented at the *34th Hawaii International Conference on System Sciences,* Island of Maui, Hawaii.

Wikipedia. (2004). *Wikipedia, the free encyclopedia.* Retrieved December 2004, from *http://www.wikipedia.org*

KEY TERMS

Copyleft: A non-exclusive, publicly-accorded legal license backed by copyright law that permits derivative works from the copyright holder's licensed works, on the condition that licensees relicense their works to the public under a similarly liberal copyleft.

Copyright: The exclusive right given to the creator of an intellectual work of text, audio, video, or software, to restrict and control how their work and its derivatives are distributed or how they are exploited for financial or other benefit.

Free Software: An earlier name for open source software, emphasizing the liberties given to end users and developers of derivative works. Particularly used for copylefted open source software. There is no requirement that the software be distributed at no charge; thus, distinct from freeware.

Free Software Foundation: Founded by Richard Stallman in 1985 to promote free software, especially the copyleft concept. Produced the GNU Manifesto (1985), the GNU General Public License (1989, 1991), the GNU Lesser General Public License (1991, 1999), and the GNU Free Documentation License (2000, 2001, 2002).

GNU General Public License: The first and still the most radical open source software license, created for the GNU Project. Requires that all derivative works be equally free (in the open source sense); that is, all derivative works must provide the full source code and must permit free use, modification, and redistribution.

GNU Project: (Stands for, "Gnu's Not Unix") Established by Richard Stallman in 1983 under the auspices of the Free Software Foundation. Its goal was, and still is, to create an open source Unix-based operating system. This goal was realized in 1991 by Linus Torvald's creation of Linux.

Intellectual Property Rights: Exclusive rights accorded by a state to legal persons based on intangible knowledge, permitting them to control how the knowledge is distributed or exploited for financial or other benefit. Consists of copyrights, patents, trademarks, and trade secrets.

Linux: A Unix-based open source operating system designed for Intel-based microcomputers. The kernel was created in 1991 by Linus Torvalds, and it was added on to the GNU Project to form what is properly called the GNU/Linux operating system.

Mozilla Project: A project formed in 1998 when Netscape released its Internet tools suite for open source development. Released its flagship Firefox Web browser and Thunderbird e-mail client in late 2004. The Sunbird calendar component is currently under development.

Open Source Software: Software whose source code is liberally made available for use, modification, creation of derivative works, and redistribution. Not necessarily copylefted.