Open Source Intellectual Property Rights

O

Stewart T. Fleming
University of Otago, New Zealand

INTRODUCTION

The open source software movement exists as a loose collection of individuals, organizations, and philosophies roughly grouped under the intent of making software source code as widely available as possible (Raymond, 1998). While the movement as such can trace its roots back more than 30 years to the development of academic software, the Internet, the World Wide Web, and so forth, the popularization of the movement grew significantly from the mid-80s (Naughton, 2000).

The free software movement takes open source one step further, asserting that in addition to freedom of availability through publication, there should be legally-enforceable rights to ensure that it stays freely available and that such protections should extend to derived works (Stallman, 2002).

The impetus of both movements has resulted in the widespread distribution of a significant amount of free software, particularly GNU/Linux and Apache Web server. The nature of this software and the scale of installation appear to be an emerging concern for closed software vendors. At this time, we are seeing the emergence of legal challenges to the open source movement and a clash with the changing landscape of intellectual property and copyright protection.

There is spirited debate within and between both movements regarding the nature of open source software and the concerns over the extent to which software should remain free or become proprietary. This article concentrates on the issues directly relating to open source licenses, their impact on copyright and intellectual property rights, and the legal risks that may arise. For more general reference, the reader is directed to the Web sites of the Free Software Foundation (http://www.fsf.org), Open Source Initiative (http://www.opensource.org), and the excellent bibliography maintained by Stefan Koch (http://wwwai.wu-wien.ac.at/~koch/forschung/sw-eng/oss_list.html).

BACKGROUND

Motivations for Participation

The open source software movement is motivated by the desire to make software widely available in order to stimulate creative activity (either in the development of derivative software or in the use of that software in other endeavors). Free software requires open source software and goes further by pursuing the protection of ideas, ensuring that the intellectual basis for a software development can never be controlled exclusively or exploited.

Why would an individual decide to participate in a movement for which they might not accrue any direct financial benefit? Boyle (2003) discusses individual motivations with the notion of a reserve price, a level at which any individual decides to become an active participant rather than a consumer and to engage in some voluntary activity. It might be out of altruistic motives; it might be for the intellectual challenge; or it might be to solve a personal problem by making use of collaborative resources (and the entry level to collaboration is participation).

Gacek and Arief (2004) identify two additional motivations for open source participation: "developers are users" (p.35) and "knowledge shown through contributions increases the contributor's merit, which in turn leads to power" (p.37). This indicates a powerful motivation through self-interest and enhanced reputation with wide recognition of contributions, especially in large projects.

The presence of large industrial consortia in the open source movement and broad participation across many software development companies indicate that many commercial organizations also are motivated to participate. Table 1 lists broad categories to explain individual, academic, institution, and commercial motivations to participate in open source production activity.

Dempsey et al (2002) liken participation in open source software development to peer review in scientific research. By releasing one's software and using the software of others, continual innovations and improvements are made.

The large-scale collaborative nature of open source software development makes it important that the contributions of individuals are recognized, and the resulting situation is that ownership of any piece of open source software is jointly held. The solution that has evolved in the development of free software and open source movements has been the development of a variety of licensing models to ensure recognition and availability of contributions.

Open Source Definition

The Open Source Initiative (OSI) was begun in 1998 to make the case for open source software development to be more accessible to the commercial world. It provides samples of open source licenses and ratifies many of the licenses that cover various open source software developments. The Open Source Definition (Perens, 1999) is a useful description of the characteristics of what constitutes open source software (Table 2).

Software licenses that meet this definition can be considered as open source licenses, and the OSI provides certification for conforming licenses. While there have been many open source and free software licenses that have been created to suit various purposes, there are three main influences that will be considered in this article: GNU General Public License models, BSD license models, and Mozilla Public License (MPL) models.

MAIN FOCUS: OPEN SOURCE LICENSE MODELS

One of the most important developments to come out of the open source movement has been the proliferation and deep consideration of various licensing models to grant various rights to users of software. The collaborative depth of the movement is neatly illustrated by the spirited debate that surrounds issues that affect the community as a whole, and the diversity of the community provides broad viewpoints that cover all aspects, from the deeply technical to the legal.

There is a wide range of different licenses (http://www.fsf.org/licenses/license-list.html), some free software, some not-free, and others incompatible with the General Public License (GPL). Table 2 summarizes the restrictions on various development activities applied by the three common classes of license.

BSD

The modified Berkeley Systems Development (BSD) license is an open source license with few restrictions and no impact on derived work. It requires only that attribution of copyright be made in source code and binary distribution of software. It specifically excludes any software warranties and disallows the use of the original organization in any advertising or promotion of derived works.

MIT(X11)

MIT(X11) is another open source license with very few restrictions and no impact on derived works. It requires only that a copyright notice be included with copies or substantial extracts of the software and excludes warranties.

The risk with unrestricted licenses such as BSD and MIT models is that a licensee can produce a derived work and not release improvements or enhancements, which might be useful to the wider community (Behlendorf, 1999).

Mozilla Public License

The modified version of the Mozilla Public License (MPL) (http://www.opensource.org/licenses/mozilla1.1.php) is a free software license that meets the OSI definition and is compatible with the GPL. It contains a number of complex provisions, but the inclusion of a multiple licensing clause allows it to be considered compatible with the GPL. The license is the controlling license for the Netscape Mozilla Web browser and associated software. It was developed specifically for the business situation at Netscape at the time of release but has since been used in many open source developments. The MPL/GPL/LGPL tri-license (http://www.mozilla.org/MPL/boilerplate-1.1/mpl-tri-license-txt) provides the mechanism for maintaining compatibility with the GPL.

Related Content

Static Signature Verification Based on Texture Analysis Using Support Vector Machine
Subhash Chandraand Sushila Maheshkar (2017). *International Journal of Multimedia Data Engineering and Management (pp. 22-32).*
www.irma-international.org/article/static-signature-verification-based-on-texture-analysis-using-support-vector-machine/178931

Enhancing Rating Prediction by Discovering and Incorporating Hidden User Associations and Behaviors
Ligaj Pradhan (2019). *International Journal of Multimedia Data Engineering and Management (pp. 40-59).*
www.irma-international.org/article/enhancing-rating-prediction-by-discovering-and-incorporating-hidden-user-associations-and-behaviors/232181

Bare Nothingness: Situated Subjects in Embodied Artists' Systems
Eleanor Dareand Elena Papadaki (2016). *Experimental Multimedia Systems for Interactivity and Strategic Innovation (pp. 16-48).*
www.irma-international.org/chapter/bare-nothingness/135122

A Visual Framework for an IoT-Based Healthcare System Based on Cloud Computing
Sanjana Chelat Menon, Umayal V. R., Gulshan Soni, Aymaan Baigand Amit Kumar Tyagi (2023). *Using Multimedia Systems, Tools, and Technologies for Smart Healthcare Services (pp. 83-95).*
www.irma-international.org/chapter/a-visual-framework-for-an-iot-based-healthcare-system-based-on-cloud-computing/314927

Learning Networks
Albert A. Angehrnand Michael Gibbert (2005). *Encyclopedia of Multimedia Technology and Networking (pp. 526-531).*
www.irma-international.org/chapter/learning-networks/17294