Open Source Software and Electronic Markets — Preface to the Special Section

ARNOLD PICOT, MARINA FIEDLER AND JOHANNES HUMMEL

INTRODUCTION

Open source software is a fascinating phenomenon in many respects. It enables the fusing of competition and collective ownership (Achtenhagen et al. 2002); the integration of a product’s user into the generation process of the product (Lakhani and von Hippel 2000; Kuan 2000; Hertel et al. 2003; Franke and Shah 2003); the participation of individual programmers as well as the commitment of large corporations (Brügge et al. 2004; Franck and Jungwirth 2002); the simultaneous pursuit of competition and cooperation (Krogh 2003) as well as the pooling of altruistic behaviour with calculated economic benefits (Lerner and Tirole 2002).

Open source software is mainly known for its new way of software development through communities of programmers, who collaborate voluntarily. Eric Raymond describes his oss developing experience in the following way:

I also believed there was a certain critical complexity above which a more centralized, a priori approach was required. I believed that the most important software ... needed to be built like cathedrals, carefully crafted by individual wizards or small bands of mages working in splendid isolation, with no beta to be released before its time. Linus Torvalds’s style of development — release early and often, delegate everything you can, be open to the point of promiscuity — came as a surprise. No quiet, reverent cathedral-building here — rather, the Linux community seemed to resemble a great babbling bazaar of differing agendas and approaches ... out of which a coherent and stable system could seemingly emerge only by a succession of miracles. The fact that this bazaar style seemed to work, and work well, came as a distinct shock. (Raymond 1999, p. 27)

Nevertheless OSS has become a serious commercial factor. Just Hewlett Packard on its own reports for 2003 more than US$2.5 billion revenue in Linux-based products (HP 2004).

Naturally OSS development poses a lot of different research questions, like ‘Who is doing the work?’, ‘Why do people contribute software code?’, ‘What exactly do OSS participants contribute?’, ‘How much do they contribute?’, ‘How efficient is OSS development?’, ‘How do OSS developers resolve their conflicts?’, ‘How does patent litigation influence OSS development?’.

All of these questions will be addressed in the following papers, based on observation, interviews and survey results. To get a better
understanding about the definition of OSS, we explain the concept in the following section.

**THE CONCEPT OF OPEN SOURCE SOFTWARE**

While the roots of open source software lead back to the 1960s the concept itself only exists in an official fashion since 1998 when the Open Source Initiative (OSI) was founded. According to OSI software can only be designated as OSS if it is protected by a licence that is acknowledged by the OSI. In order to become acknowledged by the OSI as an OSS licence a licence must fulfil the ten requirements listed in Figure 1.

Up to November 2003, 47 licenses had been certified by OSI (see http://www.opensource.org/licenses/index.php). Among the most popular of these OSS licences are GNU General Public Licence (GPL), GNU Lesser/Library General Public Licence (LGPL), Berkeley Software Distribution (BSD) and Artistic Licence.

All these licences entitle the rights holder or the user of the software respectively to read, use, change and distribute the source code of the software without any restriction. Among others, this implies that every programmer processing an OSS can make this software available for distribution (not necessarily using the same software name). Furthermore this concept involves that other software that is dynamically connected with OSS programs without being an integral part of the OSS program may remain proprietary.

However, OSS licences differ with respect to regulations whether they can be integrated with proprietary software, whether modifications must be revealed in case of distribution and whether they grant certain rights to

---

**Figure 1. Criteria of open source software according to the Open Source Initiative**

*Source: www.opensource.org*
the original developer. While GPL prohibits the integration with proprietary software and further distribution of the new software package as a proprietary good, BSD type licences allow this kind of licence design. GPL, for example, does not permit that modifications of the OSS source code might remain proprietary in case of further distribution; licences of the BSD or of the Artistic licence kind, however, open up such possibilities. Netscape and Sun have developed specific OSS-licences MPL (Mozilla Public Licence) and SPL (Sun Public Licence). These licences reserve particular rights to the original right holder, in this case commercial enterprises, whereas other OSS-licences exclude such possibilities.

Table 1 depicts the main differences between the most often applied OSS licenses.

<table>
<thead>
<tr>
<th>Unspecified</th>
<th>GPL</th>
<th>LGPL</th>
<th>MPL</th>
<th>Artistic</th>
<th>BSD type (x 11), Apache, w3c, Python, Zope</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unlimited reading, use, modification and distribution of source code</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Software can be connected and distributed with proprietary software without OSS licence</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Modifications of OSS source code may remain proprietary in case of redistribution</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Special privileges for the copyright holder regarding modifications of other contributors</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
</tbody>
</table>

Source: Adapted from Perens (1999)

Figure 2 shows that the frequency of adoption of the various types of OSS licences varies remarkably. This analysis is based on the largest Open Source Portal ‘SourceForge.net’, which on November 4, 2003 contained 45,287 OSI-approved-projects. Please note that this information on frequency does not involve any weight regarding the economic or technical relevance of the projects. Projects like Linux and Apache have their own websites and are not included in SourceForge.net.

The frequency of use of the various licence models may be explained historically (regarding the historic diffusion of the licence models), but it may also be explained by the rights that are granted with the type of licence. In 1989 Richard Stallman created the GNU GPL in order to protect software that had been created within

The key idea of GPL above all is to foreclose a later ‘making proprietary’ of a once open software code. Different from BSD-type licences, which allow redistribution of software (after a possible modification) in a proprietary fashion without revealing the source code, GPL protected software does not permit such a procedure. If a GPL protected software is integrated with a non-GPL-protected software, the entire new software package must be fully handled under GPL-conditions. This characteristic is often referred to as ‘viral’.

Lesser/Library General Public Licence (LGPL) and Berkeley Software Distribution (BSD) rank second and third in frequency after GPL (11% and 7% respectively). LGPL originally was developed in order to grant libraries in the GNU-project a less rigorous protection when using proprietary programs in connection with GNU programs. Some significance can also be attributed to Artistic-, MIT- and Apache-Licences with 2% frequency each and to MPL-Licences with 1% frequency. The cumulative share of the remaining 39 licenses is lower than 1% which means between 0 and 219 protected projects.

**RELEVANCE FOR ELECTRONIC MARKETS**

Why do we deal with OSS in EM? There are at least two possible answers to this seemingly obvious question:

1. The process of generating and distributing OSS can in itself be regarded as a specific, electronically supported market process which brings about its own institutions and cultures. There seems to be an implicit, market like arrangement for equitable exchanges within OSS communities that builds the basis for a flexible, quality oriented and coordinated evolution of software systems. In OSS development, especially in the popular projects that attract global attention and participation, almost all communication and contributions are handled via electronic channels. The papers from Zhao and Fadi; Stefan Koch and Ruben van Wendel de Joode in this issue address aspects of this organization of a very specific electronic market.

2. OSS communities can provide high quality open systems and open standards that could facilitate the building up and the diffusion of electronic market infrastructure and transactions. OSS philosophy even seems to overturn traditional market structures and strategies of software firms as the corporate world adopts the OSS approach more and more as an instrument for attacking so far more or less monopolized proprietary software markets. IBM’s investment of more than 1 billion US$ into OSS-Linux, for example, marks a strategic move to implement new rules for software markets and for the corresponding business models. This new competition could not only lead to lower prices for software and software services but could also affect the speed of diffusion of e-business. This aspect may turn out to have high relevance; however, it must be left to future in-depth research and is only indirectly dealt with in this special issue. One aspect going along with this development of commercialization — the increase of patents, is described in Vermuri’s paper.

**References**


