World Library and Information Congress:
71th IFLA General Conference and Council

"Libraries - A voyage of discovery"

August 14th - 18th 2005, Oslo, Norway

Conference Programme:
http://www.ifla.org/IV/ifla71/Programme.htm

Juni 14, 2005

Code Number: **121-E**
Meeting: **134 SI - Information Technology**

Open Source Software - definition, licensing models and organisational consequences (introduction)

Reinhard Altenhöner
Die Deutsche Bibliothek,
Frankfurt/Main, Germany

Abstract

Increasingly, the international software development is dominated by the discussion on the "magic term" Open Source Software (OSS). The question is adjacent if that concept is also relevant for libraries. This paper, for a start, focusses on the subject in general terms: What was the origin of the Open Source movement, which predecessors did she have? Which conditions are tied to the usage of OSS, which ideas are behind the development of the term? How are the copyright and utilisation rights covered with OSS? And, finally: What are success factors for the usage of OSS especially in libraries, and which concepts already have been developed?

1. Almost no other term has been discussed as intensively as Open Source Software (OSS) over the past years. No later than 1991 when Linus Thorvald put his idea of a freely accessible system software on the Net, "open source" (or, the idea of a "free software") transformed itself from a hobby of computer freaks to a serious alternative to commercially orientated, proprietary "closed source" software. That is shown not only by increasing distribution numbers but also by spreading shares in the market

that firms who are specialised in support and appliance of OSS programs, can achieve. Often, OSS is brought into a connection with open access, and the initial situation is quite similar. In both cases, there is a noticeable changement that is boiling down to the fact that established channels of distribution for software and for (electronic) documents are being questioned and, possibly, are being curtailed in their market position. At the same time, there are major differences we will come back to later. Both cases, however, have in common an enormous dynamic that corresponds to the high motivation of the acting participants.

2. OSS especially finds its acceptance by providers, in commerce, but also in the field of telecommunications and by the civil service. In Germany, the usage of OSS meanwhile is a officially declared principle for all applications; contradictory decisions in favour of proprietary software products are being pressurised to justify themselves. In Great Britain exists OSS Watch, financed by JISC as the Open Source Software Advisory Service[1]. It is said that more than 800.000 programmers use their knowledge for the development of OSS worldwide. There might be about 80.000 OSS projects, and they all attest to the creativity and quality awareness of a well-motivated community of software developers. Resonant names of OSS developments are Apache, Geronimo, Emacs, Tomcat, Samba, Python, Eclipse, Gnome, Wikipedia and Mozilla Firefox, OpenOffice.org, mySQL, Firebird. It is no disfigurement but a principle that behind those names sometimes are companies that make their product available as as source to the community.

3. Closely connected to the notion OSS is the term “free software”. In philological perspective, this is the older and still valid name which has, however, stepped back behind the PR success of the term “open source” since 1998. As a reaction to the commercialisation of the UNIX development, already more than 20 years ago the GNU project came into life (“GNU is Not Unix”). Under the label “free software”, the following guiding conditions applied for further usage and development[2]:

a) Unlimited usage for private purposes

b) Study of functioning to adapt to private purposes

c) Distribution via copy

d) Improvement and its publication

Core condition is the availability of the source code of the software in a re-usable form. In 1985 the Free Software Foundation (FSF) was founded that quasi acted as distribution organisation. Already the mentioned criteria underline that most important is the freedom of the software, not its being free of charge. The FSF even demanded to ask for money, though not for the software but for the accompanying service.

Under the label „open source“, since 1998 the Open Source Initiative (OSI) tried to gain more interest with free software. For that purpose, the „Debian Free Software

[1] http://www.oss-watch.ac.uk/

[2] See Reiter, Bernhard E.: Wandel der IT: Mehr als 20 Jahre Freie Software. In: HMD, issue 238, August 2008, p. 83 – 91. Cited by the Web text, see http://intevation.de/~bernhard/publications/200408-hmd/200408-wandel_der_it_20j_fs.html. Also see O'Reilly & Associates: Open Source - kurz & gut. German translation by Snoopy & Martin Müller. 1. ed. April 1999 http://www.oreilly.de/german/freebooks/os_tb .

Guidelines“ were adapted that explained by technical aspects the core of „free software“.

4. Now, what is the reason for the success[3] of OSS that brought analysts to the proposition that every 4th company would use OSS by 2007[4]:

- Openness. The generally open conducted development corresponds to the academic tradition to directly exchange results of (scientific) work, provide research data etc. So far, that “rule” attaches itself to the experiences of many people, uses established communication channels and co-operation methods.

- Flexibility. Many OSS projects integrate a large number of developers with very different emphasis and background. That facilitates the understanding for special requirements and, at the same time, offers the potential to quickly implement needed adjustments.

- Speed. The speed by which there is reacted to problems, errors or security leaks of the software is legendary. A large group of people that want to make a product successful and immediately undertake the tasks and test new versions, mostly is significantly faster and more successful than the proprietary competitors.

- Motivation. OSS developers are (experience themselves) as part of a community that works on a collaborative success(-product). Often they are, by the way, no part-time hobby developers but professional developers that create OSS full-time[5]. Also, the standard for governmental financed projects to provide the resulting software cost-free for others (that doesn’t necessarily mean OSS or free software), supports the motivation.

- Standards. The superimposition on mostly international based (Internet) standards generally ensures a greater independence from the single suppliers. The accessibility of a code allows, at least in theory, to hand over oncoming tasks to others – though that is, in practise, mostly less realistic. In a broader perspective that also applies for the long-term readability and usability of the software, because the open, to standards related approach acknowledges the needed sustainability on the developer’s side from the beginning.

5. The author of a software decides on its usage[6]. Even if all mentioned re-uses are allowed, a licence exists that has to be acknowledged. Only “public domain” software openly renounces a copyright and even the conventional legal restrictions do not

---

[3] Also see here the basic paper of Eric S. Raymond: The Cathedral and the Bazaar, http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/index.html, where he analyses the movement by anthropological aspects.

[4] According to Soreon Research, see Kassensturz: Open-Source und proprietäre Software im Vergleich (2003), http://www.soreon.de/site1/index.php/german/soreon_studien/software_hardware/kassensturz_open_source_und_propriet_re_software_im_vergleich and the Update 2004 via http://www.soreon.de/site1/index.php/german/soreon_studien/software_hardware/kassensturz_open_source_und_propriet_re_software_im_vergleich_update_2004_95_seiten_31_abbildungen_und_tabellen .

[5] According to a study of the Boston Consulting Group, 40% of the existing free software are so being developed, see Reiter, ann. 2.

[6] See also here St.Laurent, Andrew M.: Understanding Open Source and Free Software Licensing. Sebastopol, CA (O’Reilly): 2004. Online-Version via http://www.oreilly.com/catalog/osfreesoft/book/ .

apply[7]. However, even in that case the “license” helps to clarify the legal situation around usage and distribution of a software product; it helps to avoid uncertainties and equivocalities.

The discussion primarily is dominated by debates on the definitions of the Free Software Foundation (FSF) and the Open Source Initiative (OSI), that was deposed in the Open Source Definition (OSD)[8]. Though the certain wordings are very near each other, the FSF with its anti-commercial drive applies stronger benchmarks that the OSD[9]. That is also related to the fact that the FSF and its senior Richard Stallman also include basic thoughts on social effects of the software development and its methods, whereas the OSD places in the foreground technical and methodical aspects, thus dealing clearly more pragmatic with the subject.[10]. Core aspects of the OSD are:

- Free re-distribution
- Accessibility of the quell code
- Changeability of the code and re-use in new software
- Inviolability of the original code[11]
- No discrimination of certain persons or groups
- No restrictions for certain areas of usage (especially restrictions to commercial sectors)
- Distribution of the license (no distribution with new rules!)
- License must not be valid for a certain product (e.g., as part of a software distribution)
- License must not compromise other software (that, e.g., is also included at the same data storage; disclosure agreements)

The OSI has been trying to accredit licenses based onto these principles; the acceptance, however, of that model has been considerably lower as its creators had hoped for. Thus, the level of distribution of the GNU license still is overwhelming[12]. If

[7] Not especially mentioned is the aspect of the legal liability of the license here, though that is a major aspect, and one of the first licencses, the Berkeley Software Distribution (BSD), especially insists of the liability law. In practise, corresponding liability rules for proprietary software often do not have the desired result; in the free software area, on the other hand, often emerge very quick solutions for bug fixes or change requests; see ann. 4.

[8] http://opensource.berlios.de/docs/definition.php. See also the article Open Source Definition, http://de.wikipedia.org/wiki/Open_Source_Definition.

[9] See here the discussion in relevant panels and in Wikipedia on open source.

[10] A good example certainly is Linus Torvalds, who deals absolutely informal with commercial users of the LINUX development, as long as the developing community profits from the development results. See also an interview with L.T. by Hiro Yamagata by the title „Der Pragmatiker der freien Software“ in O'Reilly & Associates: Open Source - kurz & gut , a.a.O.

[11] This mainly means corrections / changes of original software. The user should know exactly who is responsible for / requires which version.

[12] In numbers: GPL, LGPL and directly related: over 50.000 projects, see Sourceforge.net, http://sourceforge.net/softwaremap/trove_list.php?form_cat=14

one tries to categorise the different types of licenses for free software, four groups can be named[13]:

a) GNU General Public License (Version 1: 1989) Core aspect is that, if a modified software is distributed to a third party, it has to be provided in the source code with an GNU GPL as well.

b) GNU Lesser Public License (GNU LGPL) is a lesser issue, because the software can be used as a module for proprietary software; thus, the module remains free but not the resulting software product.

c) X11 and similar licenses offer no protection. The distribution of a developed software can be made without source code and rights; the software can be used as a basis for proprietary software.

d) A further category form the „other“ licenses that are incompatibel to GNU GPL: A company distributes a software with a proprietary license; external developers do not get that right but have to cede it if their source code should be integrated.

6. OSS in a concrete situation offers a basis that can/has to be adjusted. This would mean that only big institutions that have an own development department could employ OSS successfully. In reality, meanwhile there is a broad spectrum of companies (see, e.g., the LINUX distributions) that offer services and additional developments on OSS. Only in that combination the employment of OSS becomes successful. Even big companies such as IBM or Novell are getting along with that; on the one hand, they offer important software products as OSS, on the other hand furthermore develop commercially orientated proprietary software which they understand to be additional components to the “free” basic product. There is a controversial discussion on possible cost advantages, but, as of now, few concrete studies. It is becoming apparent, however, that over time the employment of OSS needs a higher demand for consulting and adjustment (against considerably lesser introduction costs in comparison to proprietary software) that invariably leads to higher costs. But on a medium timescale (in excess of 3 years), the cost level considerably lowers under that of proprietary software.

7. Libraries in numerous areas employ IT, and thus are potential users of OSS – there are sufficient examples of that. But even special OSS for libraries has been developed by now, the most well-known being Koha and Greenstone whose mutual point of origin is New Zealand. Further examples are being named in the talks of this session and don’t need to be explained here. Anyway, the fact that there exists already an own Website on the subject, shows the very speed of the development of the subject[14].

8. Meanwhile, often “open access” is being discussed parallel to the “open source” movement. It has become clear by now what potential lies within both approaches. Especially important is the licensing, i.e. the deliberate assignement of (open) utilisation rights to a software or to a document; the question of the changeability of a text has to be attached special importance to. Only those clear regulations produce the needed acceptance that is already more prevalent in the software development context that in the scientific publishing context. The movement is also important to

[13] See Reiter, ann. 2.
[14] http://www.oss4lib.org.

libraries, although up to now, amongst them rather the idea of open access is known that the of OSS.