

From Open Source Software to Open Source Hardware

Robert Viseur

UMONS, Faculté Polytechnique,
Service d'Economie et de Management de l'Innovation
robert.viseur@umons.ac.be
CETIC
robert.viseur@cetic.be

Abstract. The open source software principles progressively give rise to new initiatives for culture (free culture), data (open data) or hardware (open hardware). The open hardware is experiencing a significant growth but the business models and legal aspects are not well known. This paper is dedicated to the economics of open hardware. We define the open hardware concept and determine intellectual property tools we can apply to open hardware, with a strong focus on open source licenses and practices. We next conduct some case studies in order to determine which licenses and business models are used by open hardware companies. We show some strong similarities with open source software and propose new opportunities for future works.

1 Introduction

The open hardware concept covers new practices for hardware creation inspired by open source licenses and development models. The concept is not totally new. Bruce Perens (co-author of the Open Source Definition) already discussed the concept in the book “Open sources: voices from the open source revolution” published in 1999, and describes open hardware from open source model. Open source later gives rise to new initiatives such as open access (in the scientific field), open data (for data owned by companies, produced by research or published by public sector) or free culture [3, 4].

Open, or open source, hardware is known by popular projects such as LEON3 (free processor compatible with SPARC v8 specifications) or Arduino (free electronic board for prototyping and do-it-yourself works). There also were some recent articles in popular newspapers or conferences for professionals. In 2010, the sector was estimated at \$50 millions, for 13 main companies and 200 active projects [10]. The potential growth was estimated at one billions dollars in 2015. However, business models and legal aspects of open hardware have not been studied in detail.

The paper is organized as follow. We define the open hardware concept and determine intellectual property tools that we can apply to hardware. We then conduct eleven case studies in order to determine business models and licenses that companies use for their projects. Finally we discuss our results and propose future works about open source hardware.

2 Definitions

The concepts of “free software” and “open source” were defined by two well-known organisms: the Free Software Foundation (fsf.org) and the Open Source Initiative (opensource.org). Such organisms do not exist in the open hardware field, so several definitions can be found. Fortunately a more complete definition was recently created by the participants of the Open Hardware Summit (openhardwaresummit.org). The definition is named “open source hardware definition” and can be considered as reference. It defines 12 criteria and clearly draws on the Open Source Definition.

Note that two kinds of open hardware projects can be found: the “open source IP” and the “open source designs”. The projects of “open source IP” are electronic components such as cores (DSP, cryptography, etc.), controllers (Ethernet, I2C, VGA, etc.) or processors (LEON3, OpenRISC, etc.). The projects of “open source designs” are more or less complex designs such as specialized boards (OGP1), prototyping boards (Arduino, Beagleboard, etc.), electronic devices (Ben Nanonote, OpenMoko, etc.) or machines (Makerbot, RepRap).

3 Intellectual Property and Business Models

The open source hardware consists of some items to protect: (1) the source code for electronics (e.g.: VHDL or Verilog source codes), (2) the source code for associated softwares (e.g.: development tools, SDK, etc.), (3) the schematics, the design files and the technical drawings (what we named “hardware design”), (4) the aesthetic value, (5) the documentations, and (6) the brands.

Several intellectual property tools can be applied (see inpi.fr and [6]). The source codes are protected by copyrights and sometimes by patents. The technical innovations on machines can also be protected by patents. The aesthetic value of a machine can be protected by industrial design rights. The documentation are covered by copyrights. The name can be protected as a trademark. Note that it exist a legal protection for the topographies of semiconductor products (europa.eu). Some similarities with open source softwares can be found, for example for the protection of source codes (electronics and associated softwares).

The open source software licenses are based on copyright. They allow the authors to fix the softwares user's rights and obligations. Patents can sometimes cover the software [6]. The free and open source licenses can include clauses about patents (such as documentation or automatic license) and trademarks.

Two families of open source licenses exist: the academic licenses and the copyleft licenses. The first one (e.g.: BSD, MIT, etc.) allows the user to change the license of the software (a proprietary license can be applied). The second one (e.g.: LGPL, GPL, etc.) requires the conservation of the license. There are a lot of open source licenses, and some of these licenses are incompatible between them [7, 9]. That is a problem in the software developments based on reusable components and, by extension, with open source hardwares (e.g.: IP cores sharing).

The licenses strongly influence the business models. The license determines the possibilities for creating value and capturing revenues by modulating the appropriability conditions [8]. Free software companies generally create revenues with the distribution of softwares, the services and the software edition [2, 13]. The software editor sells add-ons for its open source software or applies dual licensing model [8, 12, 13]. The dual licensing model implies the software is published under both copyleft and proprietary licenses. The proprietary version benefits from its license, simplifying the reuse in other proprietary developments, or technical differentiation (more features) [13].

Some licenses are specific to open hardware field, for example the TAPR Open Hardware License and the CERN Open Hardware License (www.ohwr.org; www.ohwr.org). TAPR OSH also exists in a Non Commercial version.

4 Methodology

We aim to identify emergent practices and determine which business models and licenses are used by companies. We conducted eleven case studies by analyzing organizations with commercial activities in the open hardware field. We presented a draft with partial results in an international conference about free softwares in order to give feedback from more specialized audience.

We mainly used information from the projects Web site or information from project's owners (presentations, interviews, etc.). We then searched documents published or relayed by newspapers, specialized in computer science or not. We also collected public documents highlighting particular aspects of projects such as relations between companies, relations between companies and communities or important events (conflict, license change, etc.).

We refer to Troxler about the business models of fab labs, Malinen, Mikkonen, Tienvieri et Vadén about open source hardware developers' motivations, and Baldwin, Hienerth and von Hippel about the way from innovations by users to commercial products [1, 5, 11].

5 Results

Most of the business models in open source hardware have an equivalent in open source software, except for the distribution of manufactured third party products, and the manufacture and the sell of products, which are inapplicable for softwares.

We find, as we would expect, some open source software licenses: BSD, GPL and LGPL. They are used for source codes (electronics and softwares). They sometimes cover hardware design (it is not the intended use for that kind of license).

Hardware design is also covered by Creative Commons licenses (creativecommons.org). The chosen license is often CC-BY-SA. That one allows to copy, to distribute and to modify the work, with a commercial goal or not. It requires to name the original author (BY, for Attribution) and to keep the same license (SA,

for Share Alike). That version of Creative Commons applies the copyleft principle (as GPL does with open source software).

Table 1. Examples of business models for open source softwares and hardwares

Business model	Example (OSH)	Example (OSS)
Distribution of designs	Opencores (OpenTech)	Red Hat, Novell (SuSe)
Dual licensing edition	Gaisler (Leon3)	Trolltech (QT), MySQL
Services (support, porting,...)	Gaisler	IBM, Novell, Linagora
Online services (Cloud, SaaS,...)	Bug Labs	Nexedi (ERP5 Free SaaS), OpenERP (OpenERP Online)
Distribution of manufactured third party products	Farnell	-
Manufacture and sell of products	Smart Projects (Arduino)	-

One project use a license written for open source hardware (TAPR OSH, for OGP1). The use of licenses forbidding commercial use is uncommon. No one reference to protection of the topology of semiconductor products was found.

Note that the license is not always clearly indicated. When the source code is available, it is often hosted on shared platforms such as Github (github.com), Google Code (code.google.com) or Sourceforge (sourceforge.net).

6 Discussion

The use of Creative Commons licenses for protecting hardware design raises the question of the protection which is really provided. Indeed, it is not the intended use for that kind of license: Creative Commons licenses were written to cover cultural contents (musics, books, movies, etc.). Moreover a CC-BY-SA license brings freedom to users but that freedom can be canceled by a patent covering the product. The free software and open hardware licenses often prevent or limit this by including clauses about patent, and requiring documentation or automatic patent license.

The use of GPL probably causes less difficulties for open source hardware companies. The companies revenues also come from the manufacture and the sell of products developed in a collaborative way and published under free licenses. The manufacture implies the quality control of the industrial process and the ability to manage the distribution of products. Those capacities can be reached with difficulty by a community. However the customers critical mass and the attraction power of popular brands allow to benefit from economies of scale, to offer lower prices and to invest in new products developments. Moreover quality control allows to face poor quality copies (e.g.: Arduino Asian knockoffs).

The forks are well known in the software field [14]. They are a consequence of the four freedoms given by free and open source licenses which allow to create a new project by forking source code and community. Malinen and al. note that forking is harder with open source hardware because of the need of physical copies [15]. However forking is still possible with smaller open source hardware projects. Two studied open hardware projects had been forked: OpenSPARC and Arduino. The OpenSPARC fork is a “friendly” fork. That one was made to simplify product. Several forks was made with Arduino. They were motivated by trademarks issues and by the desire to offer lower price version of products.

7 Future Works

The relation between companies and users (sometimes gathered in communities) is not well known. Some companies, such as Arduino, created business ecosystems, playing the role of leader and receiving positive externalities. Other projects seems more closed. Their owners seem less wanting to exploit the returns from collaborative development than benefit from open source hardware label and capitalize on the commercial attractiveness of documented materials which simplify the developers' work. The interactions between open hardware companies and developers should be further studied. That research could be based on questionnaires sent to the projects leaders or on activity in the collaborative tools.

References

- [1] Baldwin, C.Y., Hienert, C., von Hippel, E.: How user innovations become commercial products: a theoretical investigation and case study. *Research Policy* 35(9), 1291–1313 (2006)
- [2] Elie, F.: *Économie du logiciel libre*. Eyrolles (2006)
- [3] Lessig, L.: *Free Culture - The Nature and Future of Creativity*. Penguin (2005)
- [4] Lindman, J., Tammisto, Y.: Open Source and Open Data: Business Perspectives from the Frontline. In: Hissam, S.A., Russo, B., de Mendonça Neto, M.G., Kon, F. (eds.) *OSS 2011*. IFIP AICT, vol. 365, pp. 330–333. Springer, Heidelberg (2011)
- [5] Malinen, T., Mikkonen, T., Tienvieri, V., Vadén, T.: Community created open source hardware: A case study of “eCars — Now!”. *First Monday* 16(5) (2011)
- [6] Messerschmitt, D.G., Szyperski, C.: *Industrial and Economic Properties of Software: Technology, Processes, and Value* (2000)
- [7] Montero, E., Cool, Y., de Patoul, F., De Roy, D., Haouideg, H., Laurent, P.: Les logiciels libres face au droit. *Cahier du CRID* (25) (2005)
- [8] Muselli, L.: Le rôle des licences dans les modèles économiques des éditeurs de logiciels open source. *Revue Française de Gestion* 34(181), 199–214 (2008)
- [9] St-Laurent, A.M.: *Understanding Open Source and Free Software Licensing*. O'Reilly Media (2004)
- [10] Torrone, P., Fried, L.: Million dollar baby - Businesses designing and selling open source hardware, making millions, O'Reilly's foo camp east 2010, Microsoft's NERD center, MIT campus (2010), <http://www.adafruit.com> (read: March 6, 2012)

- [11] Troxler, P.: Commons-Based Peer-Production of Physical Goods: Is There Room for a Hybrid Innovation Ecology? In: 3rd Free Culture Research Conference, Berlin, October 8-9 (2010)
- [12] Välimäki, M.: Dual licensing in open source software industry. *Systèmes d'Information et Management* 8(1), 63–75 (2003)
- [13] Viseur, R.: La valorisation des logiciels libres en entreprise, *Jeudis du Libre*. Université de Mons (September 15, 2011)
- [14] Viseur, R.: Forks impacts and motivations in free and open source projects. *International Journal of Advanced Computer Science and Applications (IJACSA)* 3(2) (February 2012)
- [15] von Hippel, E.: User toolkits for innovation. *Journal of Product Innovation Management* 18(4), 247–257 (2001)