

# Boundaries and Diversity in Free and Open Source Software

Maurizio Teli, Stefano De Paoli

Department of Sociology and Social Research

University of Trento - Italy

maurizio.teli@soc.unitn.it, stefano.depaoli@soc.unitn.it

June 10th, 2006

OSS2006 - The Second International Conference on Open Source  
Systems

Workshop: What Does Diversity Really Mean for F/LOSS?

## Abstract

How<sup>1</sup> artefacts and discussions about them trace boundaries in the participation to FLOSS development projects? With this paper we want to answer this question, giving a contribution to the understanding of the processes of inclusion and exclusion taking place during the development practices; in other words, how artefacts act in shaping the diverse characteristics of participants. We will discuss this topic focusing on the legal artefacts called software licenses and on the debate which took place around them in two different communities.

The academic debate on FLOSS has been influenced by homogeneous views of development communities, both underlining the participation to a shared culture and claiming the existence of stable social norms. Our view is dynamic: both the developers culture and the norms regulating the communities life, are relational effects of heterogeneous networks built up through negotiations. The main concept we are referring is that of «boundary object», coming from the Science, Technology and Society field (STS). Starting from the ecological view brought by this concept, we show how the processes of construction of technology are deeply connected with conflicts and diversity. In this sense, the concept of boundary object is taken in a double meaning: not only it acts as a boundary between

---

<sup>1</sup>This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

participants, allowing agreements; but also it traces boundaries around the diverse actors allowed to participate, both humans and non humans.

We studied two communities: the geographical information system GRASS and the operating system OpenSolaris<sup>TM2</sup>, following a cyberethnographic approach, concentrating mainly on the projects mailing lists, and analyzing data in a grounded theory way. In the two cases, the licenses act as boundary objects in different moments: in the first case we followed the process of choice of the license, the GNU General Public License (GNU GPL); in the second case, the debates about the incompatibility between the project license, the Common Development and Distribution License (CDDL), and the GNU GPL.

In conclusion, talking about both cases, we showed how the views of the world inscribed in the licenses and the discussion about them are continuous process of tracing boundaries, allowing legitimate participation and excluding diverse potential participant.

## 1 Introduction: licenses as boundary objects of FLOSS

The debate about Free/Libre and Open Source Software communities has been highly influenced by the ideas expressed by Eric S. Raymond. In his principal essays (1999), the American hacker tried to give a description of Open Source software interpreted by academic researchers through a romantic concept of community composed by ideally cooperative people (Bezroukov, 1999). Despite the rich description of Open Source development given by Raymond, many scholars have adopted a reductionist perspective of his arguments. Sociology in particular has adopted a perspective on FLOSS communities where no conflict takes place and people share the same way-of-thinking and doing (i.e. Kelty, 2001; Himanen, 2001), assuming the free/open character of FLOSS communities as something given. Considering a certain limited set of social elements as the explaining factor of FLOSS we loose the complexity of the development process and the sociological research appear reductionist in two assumptions: a restricted set of values or norms are the explanatory forces of FLOSS development; and the real innovation of FLOSS is only a social one, technology is completely left out from the analysis.

Agreeing with Lanzara and Morner (2005), we consider artefacts relevant in the understanding of FLOSS organizing processes, especially licenses, mailing lists and source code. Yuwei Lin (2004) underlined the opportunity of focusing on the boundary objects of FLOSS, which can be considered as *mediators*, which

---

<sup>2</sup>OpenSolaris and Solaris are trademarks of Sun Microsystems, Inc.

«transform, translate, distort, and modify the meaning or the elements they are supposed to carry» (Latour, 2005: 39). Considering that, we will treat licenses as boundary objects.

According to Star and Griesemer (1989: 393), boundary objects are «scientific objects which both inhabit several intersecting social worlds and satisfy the informal requirements of each of them. Boundary objects are objects which are both plastic enough to adapt to local needs and the constraints of the several parties employing them, yet robust enough to maintain a common identity across sites». So, licenses are material artefacts of particular interest: as textual artefacts, it is possible to analyze what practices and political vision are inscribed in a license (*robustness*), before to describe what happen when a license enters in a specific community (*plasticity*). Our analysis will be informed by this theoretical framework, part of the STS tradition represented by Actor - Network Theory (Callon, 1986; Latour, 1993), and so we will discuss both the robustness of the licenses, in the form of what is inscribed in them, and their plasticity, through the debates surrounding them.

## 2 The licenses views of the world: cutting off potential participants

In this paragraph we focus on the views of the world inscribed in the two licenses studied. With *inscription*, we mean the act of inscribing in an artefact a framework of action which define «actors with specific tastes, competences, motives, aspirations, political prejudices, and the rest, and assumes that morality, technology, science and economy will evolve in particular ways» (Akrich, 1992: p. 208). So, our discussion of diversity can be considered as focused not on diverse people, but on diverse moral, technological, scientific and economical behaviors which can be considered included or excluded by the boundaries traced by and through the licenses.

Methodologically, we will refer to the so called ethnography of cyberspace or cyberethnography (Hakken, 1999), and consider it as a method assemblage, «practices that can cope with an hinterland of pre-existing social and material realities [...] they detect, resonate with, and amplify particular patterns of relations in the excessive and overwhelming fluxes of real» (Law, 2004: 13 - 14). The collection of data was followed by grounded theory analysis (Glaser and Strauss, 1967), recursively moving between the data and the relations traced and codified by our analysis.

## 2.1 The GNU GPL

In this paragraph we will analyze the GNU General Public License (Free Software Foundation, 1991) assuming the spokesman problem to exclude dangerous entities. In 1983, MIT's programmer Richard M. Stallman (RMS) launched the project for the writing of an entire UNIX compatible operating system, known as GNU (GNU's Not Unix). According to Stallman plans within GNU «Users will no longer be at the mercy of one programmer or company which owns the sources and is in sole position to make changes» (Stallman, 1985). Stallman wanted in fact a new definition of users, sources, programmers and companies. GNU shall then be seen as an attempt to redefine the entities-of-the-world-of-computer-programming (entities) in a different way: to symmetrically construct a sharing-community of programmers and companies based on an operating system where «Complete system sources will be available to everyone» (Stallman, 1985).

The GNU General Public License was born as «the method» for ensuring the goals of this project. The preamble of the GPL indicates who are the most important entities to exclude:

The licenses for most software are designed to take away your freedom to share and change it.

Most part of the software licenses clashes with FSF plans, because they could be used to «owns the sources». Thus the FSF had to exclude them from the GNU project. The FSF had to convince the entities-of-computer-programming to accept this exclusion. According to Stallman (1986):

I do this by copyrighting the programs and putting on a notice giving people explicit permission to copy the programs and change them but only on the condition that they distribute under the same terms that I used, if at all.

Stallman conceived the «hack» called Copyleft, which uses copyright laws against its usual purpose. Within Copyleft, authors give to everyone permission to run, copy, modify their program and to distribute modifications. But the Copyleft imposes some restriction on the use of GNU and software in general: the GPL will prevent GNU software from being turned into proprietary software. In other words to enroll the entities the GPL needs to the exclude the use of «the licenses for most software» which take away the users freedom «to share and change the software». So, two different behaviors are in play: people who allow sharing and changing are included, people who do not allow them are excluded.

## 2.2 CDDL

In this paragraph, we will give a description of the view of the world inscribed by Sun on the Common Development and Distribution License - CDDL (Sun Microsystems, 2005b) in order to draw a preliminary sketch of the boundaries defined by this license. We will focus only on two aspects: the presence of licenses from «the various geological eras of software history» and the «patent peace provision».

The first issue is based on the fact that part of the code of the Solaris Operating System is regulated by agreement between Sun and other software producers, making impossible to release the entire system under a program - based license, like for example the GPL, or under a license allowing only dynamic links, like the GNU Lesser GPL (LGPL). So, the choice was a file - based license. We have to notice that the license, as a file - based license, covers the files part of the system and not the system as a whole. This is explicitated at Term 1.3, which states that

Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of *files* containing Original Software with *files* containing Modifications, in each case including portions thereof. (emphasis added).

How to connect this Term with the legacy licenses phenomenon? In order to answer this question we need to refer to another Term of Section 1: Term 1.6, which defines «Larger Work» as «a work which combines Covered Software or portions thereof with code not governed by the terms of this License.» In this case, the file - based protection allows Sun to combine source files they can release under CDDL license with files protected by different licenses. At the same time, this clause, connected with 3.5, «Distribution of Executable Versions», and 3.6, «Larger Works», acts in shaping the possible participant to the commons, a boundary is traced, and people who wants to release their larger work under a different license are included.

The boundary traced not only includes someone, it also excludes two developer groups. The presence of the copyleft clause (see 3.1. «Availability of Source Code», and 3.2. «Modifications»), chosen in order to provide «the protections and freedoms necessary for true open source» (Sun Microsystems, 2005a), exploits the same «hack» as the GPL, paradoxically excluding the possibility to combine CDDL - covered software with code covered by the GPL. Sun seems to share with FSF the idea that copyleft maintains the vitality of the software common, but this choice divides CDDL developers from GPL developers. The second group excluded is that of people who want to distribute in a proprietary form their Modifications to the covered files.

The second issue is very interesting. In Section 2 and Section 6 there are different and complementary statements related to the «software patent» issue: Section 2 indicates that both the Initial Developer and the Contributors grant to other users of the software the opportunity to use the patent they have in relation to the code they contribute with; Section 6 states that if someone «assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor [...] alleging that the Participant Software [...] directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, [...] shall, [...] terminate prospectively and automatically». So, a mechanism of reciprocity about patents is inscribed in the license, excluding the «patent terrorists» (Wilder, 2005) from the participation to the software common and regulating the related behavior in the community. This is a clear and declared tentative to create a «patent-safe developer commons around OpenSolaris» (Phipps, 2005), and it is also considered one of the reason excluding the GPL as the possible license for the project.

In conclusion of this paragraph, we have just seen as the history of the system as well as general issues in software development, like software patents, have been taken into account in the license writing. They also shape the boundary around the diverse possible participants to the projects, in both the human and non-human sides, as well as the community behavior.

### 3 Boundaries shaping in debates

In this part we will describe how the *plastic* part of the licenses, their interpretations in debates around them act in giving shape to the possible participants characteristics. As well as in the previous section, we will start with the GRASS case and conclude with the OpenSolaris one.

#### 3.1 GRASS: choosing interests

The Geographical Information System GRASS (Geographic Resources Analysis Support System<sup>1</sup>) was born at the beginning of the '80 as a small development project of the United States Army Corp of Engineering Research Laboratory (USACerl). In 1996 USACerl decided, for a number of reasons, to stop the development of the system (GRASS version 4.1) and invited the users to migrate toward proprietary GIS and commercial version of GRASS (US Army Cerl, 1996).

In the 1998 a new GRASS development team (GDT) came up with the aim to relaunch the GRASS development and community. The new GDT was

composed by an international group of volunteers researchers affiliated to different institutions; the new team assumed also a structure very close to the «town council» model (Cox, 1998), characterized by a restricted group of programmers leading the development of a large project.

In October 1999, a discussion on the possibility to release GRASS under a FLOSS license took place in the GRASS users mailing lists. After some negotiations inside the new community, the GDT adopted the GPL (V.2.0) as the copyright license for the new GRASS software (version 5.0 and subsequently).

The GRASS community found an agreement choosing a well know FLOSS license. The GPL has served as then boundary object between the different institutions and individuals participating in the new course of GRASS. Moreover, within the release under the GPL, GRASS and their community moved from an uncertain public domain situations toward the copyright protection of the GPL (GRASS Development Team, 1999).

The GPL Copyleft «method» represents a form of controversy and also a form of exclusion within GRASS community. The restrictive character of the Copyleft closes the way to applications with incompatible licenses and to their developers, becoming in fact what could exclude technical and human actors from the community of GRASS.

«Why GPL» is the thread of a discussion happened within the GRASS Developers Mailing List in March 2001. With the start of this discussion some aspects of the release of GRASS under the GPL are questioned by a newcomer of the list we will call Pippo (true name initials are SA):

«It is possible for the authors of the original code to re-release their code under the LGPL or another license.»

In fact, the re-release of GRASS under the LGPL would change the positions of participants in GRASS community, allowing non-GPL and proprietary plugins to link directly with the system. The LGPL would place the copyleft «method» on the program itself but would not apply any restrictions to other software linking with GRASS. LGPL would require only the changes to GRASS to be LGPL licensed.

Pippo's proposal is in fact an attempt to modify some rules inside the GRASS framework. However Pippo is thus forced to defend his own proposal in front of the previous agreement of the GRASS community «to release GRASS under the GPL»:

«> The only difference between the GPL and LGPL is software linked to the GRASS library would  
> not have to be GPL. It offers the same protection of the software, but doesn't scare away people

> who do not want to release GPL software.

And that's a big difference. If people are scared off because they can't make money using the freely given contributions of other, so be it.»

[EM, GRASS DevML, discussion dated 22 March, 2001 (the quoted part is Pippo's message)]

«If end users get accustomed to the proprietary enhancements, the owner of the proprietary rights to get power over the GRASS development. The GPL is the license which protects against this and thus most firmly ensures the long term freedom of the software.»

[BR, GRASS DevML, discussion dated 26 March, 2001]

Here it emerges a protection of the GDT agreement on the GPL and a clear exclusion of those developers which want to link proprietary application to GRASS. These members (true name initials are EM and BR) assume that through a different license (also others FLOSS licenses) the applications owners can bring some power on the development of GRASS.

Pippo's position is not only contradicted. Pippo himself is clearly invited not to join or contribute to GRASS community if he doesn't agree with the GPL choice:

«GRASS is protected under GPL by our legal team who know the intricacies of that license.

[...] If you don't like that, you don't have to use GRASS or contribute.»

[BB, GRASS DevML, discussion dated 27 March, 2001]

The Copyleft boundary of the GPL is evident: application owners, with non compatible GPL licenses are separated and excluded by the development and by the GRASS community. We have here a clear separation from the GRASS framework of both the developer and his code.

### **3.2 OpenSolaris: discussions on license, discussions on organizing**

In this paragraph, we will concentrate only on part of the long debate which took place on the osol - discuss mailing lists, the main project mailing - list for number of messages. We will see how, during this discussion, the positioning of the participants in the FLOSS social world has been taken into account and how this can be considered harmful by others participants, more interested in the development phase, bringing to a consideration of the same mailing lists organization.

The debate (the thread was entitled «CDDL & GPL incompatible, what does it mean?») involved different topics, one of which was the belonging of the project to the FLOSS social world. Look at the following posts:

«the bottom line is that opensource developers and users want their software to be GPL. if it is not then these people will be turned off by opensolaris.»

[MW, osol - discuss, discussion dated 20 August, 2005]

«No, \*some\* users and developers want their software to be GPL. And just as those users will be turned off by OpenSolaris because it is not, there will be many that will be turned off if it becomes GPL. [...] It would make it more attractive those developers that actually care about the license, or are GPL zealots. The majority of users don't care what license a program is under. They just like good software.»

[SW, osol - discuss, discussion dated 20 August, 2005]

In our point of view, these posts are very interesting, because: they give meanings to what an open source developer and user is; they suggest how to increase the participation to the project, through the mediation of the license; they describe the FLOSS social world as composed of almost two groups, «GPL zealots» and not - «GPL zealots». They also involve a group silent, «the majority of users», who «don't care what license a program is under». The participation of this second group is the focus of other posts, like the two following:

«Irrelevant discussion on an open source project mailing list is a cancer. It must be cut out to prevent those of us with high email loads from unsubscribing. Pretty soon, the only people left on the mailing list will be the ones not working on the project. BTW, that is also why Apache project lists are called "dev", not "discuss", since it narrows the acceptable discussion to something that active engineers can keep up with and still do their work.»

[RF, osol - discuss, discussion dated 7 September, 2005]

«Some engineers and managers have, indeed, unsubscribed – primarily due to flames, unfocused discussion, and volume. [...] Around mid-pilot we started creating more lists to try to distribute the load so the technical conversations would not get lost. The most substantive technical conversations are happening on code, DTrace, etc. However, we do need a venue for newcomers and general issues that are project-wide. Although this is certainly developer program, we do want to grow and get more diverse over time.»

[JG, osol - discuss, discussion dated 7 September, 2005]

In this case, the focus is on the way the discussions on license seen before act on the participation of technical people, who «do their work». They can be far from the project due to the «high email loads». For the first person, the solution can be to build up mailing lists called «dev», instead of «discuss», in order to focus on the work to be done. The second one answered clearly: this is a known problem, but to increase the number of participants, the presence of a project-wide mailing lists is fundamental. Two different way of considering participation emerge, thanks to the presence of political disputes about the license.

So, we have seen how the license not only acts in shaping the boundaries around diverse possible participants, but also stimulates debates that re-shape that boundaries, both involving the political ecology of FLOSS and the organizations of other mediators participating in the life of a developers community.

## 4 Conclusions

We told a story of legal artefacts as mediators, of licenses as shaping boundaries, and of boundaries re-shaped by discussions. We told a story of people and objects connected in allowing participation by different actors in FLOSS communities. We told a story of boundary objects, of history of licenses, and of communities living.

We learned that the connections considered in the licenses written text restart from it. They connect diverse actors, diverse social elements and diverse objects. We learn that, considering diversity of participation in FLOSS communities, we have to take care of objects and mediators, because they are able to shape new connections, new mediations, and new participants. We suggests that more cyberethnographic research on FLOSS mediators can be able to open perspective both to understanding FLOSS communities life and to make actions in order to increase the participation of diverse entities, both human and non - human.

## References

- Akrich, M. (1992). The de-scription of technical objects. In Bijker, W. and Law, J., editors, *Shaping Technology, Building Society: Studies in Sociotechnical Change*. MIT Press, Cambridge, MA.
- Bezroukov, N. (1999). Open source software development as a special type of academic research (critique of vulgar Raymondism). *First Monday*, 4(10).
- Callon, M. (1986). Some elements of a sociology of translation: domestication of the scallops and the fishermen of Saint Brieuc Bay. In Law, J., editor, *Action*

- and Belief: a new Sociology of Knowledge?*, pages 196 – 233. Routledge and Kegan, London.
- Cox, A. (1998). Cathedrals, bazaars and the town council. <http://slashdot.org/features/98/10/13/1423253.shtml>. [01/06/2006].
- Free Software Foundation (1991). Gnu general public license 2.0. <http://www.gnu.org/copyleft/gpl.html>. [01/06/2006].
- Glaser, B. G. and Strauss, A. L. (1967). *The discovery of grounded theory :strategies for qualitative research*. Weidenfeld and Nicolson, London.
- GRASS Development Team (1999). Press release, grass gis released under gnu-gpl. <http://grass.itc.it/announces/gnu-release.html>. [01/06/2006].
- Hakken, D. (1999). *Cyborg@Cyberspace. An Ethnographer Looks to the Future*. Routledge, New York.
- Himanen, P. (2001). *The Hacker Ethic and the Spirit of the Information Age*. Random House, New York.
- Kelty, C. M. (2001). Free software/free science. *First Monday*, 6(12).
- Lanzara, G. F. and Morner, M. (2005). Artifacts rule! how organizing happens in open software projects. In Czarniawska, B. and Hernes, T., editors, *Actor Network Theory and Organizing*. Copenhagen Business School Press, Copenhagen.
- Latour, B. (1993). *We Have Never Been Modern*. Harvard University Press, Cambridge.
- Latour, B. (2005). *Reassembling the Social. An Introduction to Actor-Network-Theory*. Oxford University Press, New York.
- Law, J. (2004). *After Method. Mess in Social Science Research*. Routledge, London and New York.
- Lin, Y. (2004). *Hacking Practices and Software Development: A Social Worlds Analysis of ICT Innovation and the Role of Free/Libre Open Source Software*. PhD thesis, University of York, Science and Technology Studies Unit, Department of Sociology.
- Phipps, S. (2005). Cddl reflections. <http://blogs.sun.com/roller/page/webmink>. [01/06/2006].

- Raymond, E. S. (1999). *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly and Associates, Sebastopol, California.
- Stallman, R. M. (1985). The gnu manifesto. <http://www.fsf.org/gnu/manifesto.html>. [01/06/2006].
- Stallman, R. M. (1986). Byte interview with stallman. <http://www.gnu.org/gnu/byte-interview.html>. [01/06/2006].
- Star, S. L. and Griesemer, J. R. (1989). Institutional ecology, 'translations' and boundary objects: Amateurs and professionals in Berkeley's museum of vertebrate zoology, 1907/39. *Social Studies of Science*, 19(3):387 – 420.
- Sun Microsystems (2005a). Cddl faq. <http://www.opensolaris.org/os/about/faq>. [01/06/2006].
- Sun Microsystems (2005b). Common development and distribution license. <http://www.opensolaris.org/os/licensing/>. [01/06/2006].
- US Army Cerl (1996). Announcements. <http://web.archive.org/web/19970619195255/www.cecer.army.mil/announcements/grass.html>. [01/06/2006].
- Wilder, R. (2005). How to fight against patent terrorism. *News.com*. <http://news.com.com/How+to+fight+against+patent+terrorism/> [01/06/2006].