Palmira Granados Moreno

What is Free Software?

Free Software is a movement that is concerned about the way information is being distributed. It is mainly focused on software and the way the current intellectual property system tries to "protect" it. This movement is defined in the Philosophy of the Gnu Project as "matter of liberty, not of price... a matter of the users' freedom to run, copy, distribute, study, change and improve the software."[1]

In the same vein exists the Open Source movement, which refers to projects, and specially computer software, that are open to the general public, and promotes the availability of the source code, and the rights to modify, redistribute, package and sell of the software.

Nowadays, even though these two movements have slight differences in principles, they have one purpose in common: promotion of free/libre/open knowledge for everyone by proposing a change in the protection granted to software and by the implementation of licenses such as the GPL, the BSD and the Creative Commons licenses.

When did it start?

The Free Software movement began in 1983 when Richard Stallman conceived and announced what he called the GNU Project. His purpose: to restore the cooperation that was prevalent in the early days of the computer industry. This project started as a proposal to create an operating system which by the 1990's was considered complete after including the kernel created by Linus Torvals called Linux. Since then, several applications have been developed under the same philosophy: free distribution of computer software.

The Open Source Initiative began in 1998 when Eric S. Raymond and Bruce Perens decided to form an initiative that promoted the practical benefits of sharing source code and therefore, attracting software houses and other high-tech industry companies into the concept proposed. This divergence came as a consequence for the confusion caused by the term "free." The term of "open source" was created by Christine Peterson of the Foresight Institute.

1 http://www.gnu.org/philosophy/free-sw.html

Why did it start?

The people within the movement accurately assert that the spirit of free collaboration they uphold has always been present in the process of intellectual creation. Referring to the history and origins of intellectual property, according to Arnold Hauser,[2] during the times of the ancient Greece, authors and performers saw their works as part of the culture to which they belonged and therefore, they considered the act of others and of themselves to publicly perform the works of the collectivity as the best way to distribute those works, and as the best way to preserve them. According to the same author, the same happened during the height of Christianity, when intellectual works were considered an expression of the deity through the artist.

This model of collective cooperation and collaboration has been kept throughout the years, and nowadays there are several industries and fields that are based on it. Software in particular is one of them. Supporters of this movement, as well as many people involved in the computer field, assert that the way in which software is created and developed requires collaboration from a great number of persons. Thus, there should be a group of people working on the creation of the compiler, a group of people working on the kernel, a group of people working on the editors, many groups of people working on the many different applications, and last but not least, a group of people working on the testing, scrutiny, experimentation and maintenance of the work as a whole. In the words of Eric Raymond “Given enough eyeballs, al bugs are shallow” which refers to the fact that the more people collaborating, the better the outcome.

In my opinion, the current intellectual property system was created to meet the needs of an industrial era, to measure up with the demands of the people devoted to copying, publishing and distributing books in the industrial era (referred to as stationers in Hauser's book). However, I agree with the people from the aforementioned movements that the current digital and technological era calls for a different system of protection. An era whose Economy is based on information and that depends on the speed in which this

2 Arnold Hauser. *The Social History of Art.* (New York, United States: Alfred A. Knopf, 1952) cited by Ronald V. Bettig. *Copyright Culture.* (United States: West View Press, 1996) at p. 11

information is distributed needs a system that promotes efficiency, that recognizes the importance of cooperation, of freedom, and of democratic knowledge.

How does it work?

Richard Stallman defines free software as "a matter of liberty, not price." To illustrate the meaning he wants to point out with his definition of free software, he refers to free as "in free speech, not as in free beer."[3] He refers to five types of freedom:

- The freedom to run the program for any purpose
- The freedom to study the program, and to adapt it to one's needs
- The freedom to redistribute copies
- The freedom to improve the program and to distribute every improvement.

In order to accomplish and observe these freedoms, access to the source code[4] is required. He asserts that a program, in order to be free, does not have to be gratis or free of charge, but he emphasizes as well that the freedom he is referring to is the one where the user does not have to ask for permission, and that these freedoms must be kept in every derived work.

The open source initiative, on the other hand, claims a users' right to have open access to the source code and to change and/or share it, and the requirement to preserve the author's names and copyright statement in the code. In the open source definition it is also prohibited any kind of discrimination against persons, groups or endeavors.

The main difference between the free software movement and the open source initiative is the status of the derived works. Under the GPL (main free software license) every work based on free software code must be distributed under the terms and conditions provided by the GPL including every single freedom stated in the free software manifesto, independently of the case where a work is based on software distributed under the GPL and, at the same time, on proprietary software. On the other hand, open

---

3 *Supra*1

4 According to Jonathan Zittrain in his article "Normative Principles for Evaluating Free and Proprietary Software." (2004) 71 University of Chicago Law Review 265at p. 5, source code is what programmers write by using different languages such as C, Java, Python, etc., and object code is what computers run, is what is in binary code.

source licenses, require that the source code is open and that the terms and conditions of every license under which any piece of code used in the derived work is distributed, are kept and observed. In other words, the open source license of the derived work must not insist that all the other programs distributed and used on the same medium must be open source. It is important to highlight that the licenses of both movements guarantee anyone's right to commercialize the outgoing products, which is one common misconception.

Hence, they recognize the need for investment, for money to implement this industry, but they have a different model of business. Their main purpose is to give consumers what they want. They know that consumers want control over their software, so they decide to make this characteristic their advantage and what defines their product. They also give stability, reliability and security while sharing the costs with many developers. They aim at building a great product which can stand for quality, just as every other proprietary system, but, given the characteristic of making available the source code and granting the right to modify and adapt it, their product also has the advantage of being extremely flexible and highly customizable.

Bob Young[5] cites two examples referring to the quality of software under these models. First, he mentions that the people from the NASA have declared that “software is not software without source code.” NASA needs not high reliability, but perfect reliability, and this is what they get when they have control over every single part of the software in which their whole work is based on. The second example is Grant Guenther, a member of Empress Software's database development team who needed a secure method for moving large files from his office to his home and back. At that time, there were not Zip drive drivers for Linux, so instead of purchasing a proprietary solution, he wrote one and worked with other Zip drive users across the Internet to test and refine the driver. These two examples also stand for the reliability on the quality and commitment of the people involved in the implementation of free/open source products.

5 Robert Young. *Giving It Away.* 2003 on-line http://www.lulu.com/content/29611

Now then, the way they charge for the software may be by selling the copies of the software, or by giving for free but charging for installation and support, which even though may seem a weak source of income, there is a considerable need in major global corporations for support and maintenance services in the rapidly evolving software.

Current intellectual property protection for computer programs.

Section 2 of the Canadian Copyright Act[6] defines computer program as "a set of instructions or statements, expressed, fixed, embodied or stored in any manner, that is to be used directly or indirectly in a computer in order to bring about a specific result". This section also considers computer programs as included in the category of literary works. As such, the owner of the copyright of the program has, according to section 3, the sole right to produce or reproduce the program or any substantial part thereof in any material form, and if the program is unpublished, to publish it or any substantial part thereof, including the right to rent it out.

Regarding infringement, in section 27, it is provided that a person will be considered as infringing copyright if he or she, without the consent of the owner of the copyright, produces, reproduces, publishes or rents out the program or any substantial part thereof, or does any other right exclusively reserved to the owner of the copyright of the program. It is also provided that the act of selling or renting out, distributing to such an extent as to affect prejudicially the owner of the copyright, exposing or offering for sale or rental, possessing for the aforementioned purposes or importing to Canada for the same purposes a copy of the program, will be considered secondary infringement.

On the other hand, section 30.6 provides that it will not be considered an infringement if a person who owns a copy of the program makes:

a) a single reproduction of the copy by adapting, modifying or converting the program or translating it into another computer language, if, and only if that copy is proved to be "*essential for compatibility of the computer program with a particular computer, solely for the person's own use, and destroyed immediately after the person ceases to be the owner of the copy*", (emphasis added) or

6 *Copyright Act* R.S.C. 1985, c. C-42

b) makes a single reproduction for backup purposes, and then "*destroyed*" when that person ceases to be the owner of the copy. (emphasis added).

Conflict

The Copyright Act does not mention anything about the source and the object code, it does not mention anything about reverse engineering, and still, in the US exists the DMCA that tries to regulate these issues in a way that it may deter innovation in the computer field.

Moreover, given that any reproduction or distribution of the computer program or a substantial part thereof is considered an infringement, unless the reproduction can be framed within compatibility issues, own personal use and backup situations and immediately destroyed after the person ceases to be the owner of the copy, it seems to me that the purpose of intellectual property of promoting creative intellectual activity stated when the WIPO became part of the UN is contradictory this provision.

As explained by the people from the free software and the open source movements, in order to efficiently implement in the computer field, there is a need for cooperation, there is a need for access to the source code, and there is a need for freedom to create. There is a need for the intellectual property protection system, if it will still comply to its aforementioned purpose, to acknowledge the way the computer field works and what their needs are, and to maximize every creation on which the whole picture of innovation is being built as opposed to *destroying* modified and sometimes, improved copies.

Proposal

"To establish, within the intellectual property regulation, that software is to be protected only if the copies thereof are distributed with the source code open and with the freedom to adapt, modify, use and implement on it".

This does not include the right to commercially distribute, in whole or in part, copies of the computer program, which still should be an exclusive right of the author.

Thus, a programmer, can get and dedicate his or her life to innovate in the software area

without needing another way of living. However, this would not have to be the only business model. As mentioned before, recouping investments and making profits may come from charging for installation, service and training, from creating such a strong product that more people, companies, and the government itself are interested in investing, using and distributing.

By doing this, innovation and culture are promoted at the same time. Culture is something that is continuously built on past culture, and today's culture is the basis for tomorrow's culture. Within this model, culture and knowledge are accessible for everybody.

Innovation on the other hand, gets more efficient. It allows interoperability among the different elements that comprise a robust computer system. This efficiency is more obvious when the creation process in software is understood. As Bob Young points out, when a new version of Red Hat Linux is shipped, 800 separate packages are shipped. Without the collaboration of many people, without the freedoms proposed, it would take much more time to finish the project, it would not be as robust and stable because fewer people would be working on it, and the process would be more complicated and less practical because of the several permissions that would have to be asked for any modification or improvement.

With respect to competition, according to the American Antitrust Division and Mark A Lemley et. al.[7], since in innovation markets the close substitutes are research and development efforts, technologies, and goods that significantly constrain the exercise of market power, the fact that a hypothetical monopolist may have the ability to retard the pace of research and development can be considered an anticompetitive practice. Thus, it seems to me that since the source code is open, the IPR holder cannot unduly try to prevent hypothetical competitors from creating applications, operating systems, or other elements of the computer system, and therefore the IPR holder would not be able to extend the protection granted by intellectual property laws to products other than those

7 Mark A. Lemley, et. al. *Software and Internet Law*. (United States: Aspen Law ad Business, 2000) at p.553.

over he or she is entitled to.

That the source code is accessible to everyone, also reduces the chances that an IPR holder has market power over the relevant market because there would several potential competitors that may create close substitutes.

Where is it aiming at?

On October 4, 2004, the General Assembly of the World Intellectual Property Organization agreed to adopt a proposal called "Proposal for the Establishment of a Development Agenda for WIPO[8]" presented by Brazil and Argentina that expresses their concern about:

- The inequality of access to education, knowledge and technology that undermines development and social cohesion,
- The enormous costs on consumers and the retard in innovation due to anticompetitive practices in the knowledge economy,
- The barriers to follow-on innovation that authors and inventors are facing,
- The concentrated ownership and control of knowledge, technology, biological resources and culture harm development, t
- The threaten to the exceptions for disabled persons, libraries, educators, authors and consumers because of technological measures designed to enforce intellectual property rights in digital environments and how this undermines privacy and freedom, among other things, which highlights social, cultural, economic and competitive arguments that support a change, or more like an adaptation of the intellectual property system to this era, embracing a balance, appropriateness and stimulation of both competitive and collaborative models of creative activity.

8 http://www.cptech.org/ip/wipo/genevadeclaration.html