

Freedom to share knowledge¹

Llibertat de compartir coneixements

Paper presented to the Online Congress – Observatory for Cybersociety 2006

Wouter Tebbens

General coordinator of the SELF Project

Amsterdam, November 2006

keywords: Free Software, Libre Software, Open Source, Open Standards, copyleft, free content, vendor lock-in, democratisation, education, training, learning material

paraules clau: Software Lliure, Programari Lliure, Open Source, Estàndards Oberts, copyleft, contingut lliure, democratització, educació, training, material educatiu

1 *License:* Creative Commons Attribute-ShareAlike 2.5 See: <http://creativecommons.org/licenses/by-sa/2.5/en/>

Abstract

Societies have been able to advance in great part due to the sharing of knowledge. While in current times the internet and the use of computers have enabled new ways of communication and sharing of information, legal and technical obstacles impose serious threats. Since the beginning of the 1980s Free Software has evolved as a counter movement to these restrictive tendencies. Free Software is not only a software program that guarantees the freedom to use, study, adapt, copy and distribute. It provides as well the legal framework for sharing and creating knowledge in the digital age.

From this starting point, this article argues the importance of Free Software in science, culture, economy and society in general. In more concrete terms, this article shows how the use of Free Software and its principles are fundamental in all these domains. In science Free Software enables the verification and falsification of scientific theories. In culture the use of copyleft facilitates the sharing and collaborative creation of cultural works. In economy Free Software represents a real alternative to the current software monopolies and hence its adoption would stimulate innovation and create equal opportunities in the software market. Finally, Free Software is essential for transparency, social inclusion and democratisation of information.

If Free Software is so important, how come that still so few people are aware of it? As it will be argued, there are multiple reasons. In order to address these, it is fundamental to introduce Free Software in education. That is what the SELF project is about. By providing a worldwide platform for the collaborative production and sharing of freely available education and training materials about Free Software and Open Standards, the SELF platform aims to contribute to the adoption of Free Software. This can only be achieved by the active participation of all parties involved, that is, from educational and governmental institutes to ICT and training companies, publishers, NGOs and Free Software communities.

Resum

Les societats han pogut avançar, en gran part, gràcies al fet de compartir coneixements. Mentres actualment l'internet i l'ús d'ordinadors possibilita noves maneres de comunicació, obstacles legals i tècnics imposen serioses limitacions. Des de principis dels 1980s, el Software Lliure ha evolucionat com a resposta a aquestes tendències restrictives. El Software Lliure no és només un programa que garanteix la llibertat d'ús, estudi, adaptació, còpia i distribució. A més, ha proporcionat les bases legals per a la creació i ús compartit de coneixements en l'era digital.

Partint d'aquesta base, aquest article argumenta la importància del Software Lliure en la ciència, la cultura, l'economia i la societat en general. Més concretament, aquest article demostra com l'ús del Software Lliure i els seus principis son fonamentals en aquestes àrees. En la ciència, el Software Lliure permet la verificació i falsificació de teories científiques. En la cultura, l'ús del Copyleft facilita la col.laboració i ús compartit de creacions artístiques i culturals. En l'economia, el Software Lliure representa una alternativa real als monopolis de software actuals i, per tant, la seva adopció estimularia la innovació i crearia igualtat d'oportunitats en el mercat de software. Finalment, el Software lliure és essencial per a la transparència, inclusió social i democratització de la informació.

Si el Software Lliure és tan important, com s'explica que tan poca gent sàpiga què és? Com s'indicarà, hi ha múltiples raons. Per combatre-les, és fonamental introduir el Software Lliure en l'educació. Aquest és precisament l'objectiu del projecte SELF, una plataforma mundial per a la producció i ús compartit de

material educatiu i formatiu sobre Software Lliure i Estàndards Oberts. Aquest objectiu depén en gran mesura de la participació activa de tots els interessats, des d'instituts educatius i governamentals a empreses de les TIC i de formació, editorials, ONGs i comunitats de Software Lliure.

1. Introduction

Historically our societies have been developing through the sharing of knowledge. Sharing of knowledge between individuals and between different societies and cultures. For instance, the invention of the printing press would not have been possible without previous inventions such as presses for olives and wine that were known since Roman times and paper that was used in Arab countries and China. Once Gutenberg designed the printing press, the mechanisms behind it were shared freely and hence there was no need to reinvent it again.

In the last decades the process of knowledge sharing has accelerated due to new developments such as the internet and software technologies. However, while these new technologies have increased the extension and intensity of the sharing of knowledge, legal and technical limitations impose serious threats. For instance, while the internet facilitates the distribution of music, books, software, movies, etc., new rules restrict the possibilities to share these works with others. In contrast, in the physical world, you may lend your CD, book or video tape to whoever you want².

Since the beginning of the 1980s Free Software has evolved as a counter movement to these restrictive tendencies. Free Software (also known as Libre Software or Open Source Software) is software that grants the freedom to use, study, adapt, copy and distribute copies of the software. Free refers to these freedoms and not to free of charge. This kind of software is typically written collaboratively by many authors. This means that, unlike proprietary software, Free Software can be developed freely upon the results that others have created before. In other words, the developers of Free Software 'stand on the shoulders of giants'. This is one of its main strengths: unlike proprietary software, Free Software develops exponentially. As we said for the invention of the printing press, there is no need to re-invent even the smallest application. Any development can be re-used to build more complex systems. This has resulted in practical solutions for virtually any software field.

For all these reasons, Free Software seems to provide the fundamental tools and concepts for science, culture, economy and society in general. This is the main argument of this article. To argue that Free Software is so fundamental for the development of these areas will lead me to defend the need to promote Free Software in education and to introduce the SELF Platform. Without pretending to be exhaustive in its explanations, the underlying purpose of this article is to give the reader a few thoughts and examples in order to provoke debate about education and software technologies.

2. Free Software in science

Scientific research uses more and more computers to run tests, aggregate results and present conclusions. In order to verify experiments and falsify scientific theories, the implemented processes in computer programmes need to be validated thoroughly and made available to other research groups. In other words, only by making available the software completely, experiments can be repeated and their results validated. This intrinsic need cannot be guaranteed with proprietary or closed software: the software is a black box and only limited parts of the internal formulae maybe studied and verified. Moreover, the peer review tradition – as part of our academic culture – can not be exercised fully as the peers cannot simply copy the software used in the original experiment.

² In legal terms this is called “fair use”. For more information see: http://en.wikipedia.org/wiki/Fair_use

Scientific software therefore has some additional constraints that software in other sectors not necessarily has to fulfil. First of all, the source code of the software should be available to guarantee the possibility to verify and validate its working. Secondly, the software applications should be shared freely to permit peer review. Thirdly, improvements or modifications should be allowed in order to build upon the scientific results obtained earlier. In this way science can progress without limitations and every research institute can perform ongoing research based on the latest achievements.

These conditions can only be guaranteed by working with Free Software since it guarantees the user the following four freedoms³:

- The freedom to run the program, for any purpose (freedom 0).
- The freedom to study how the program works, and adapt it to your needs (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbour (freedom 2).
- The freedom to improve the program, and release your improvements to the public, so that the whole community benefits (freedom 3). Access to the source code is a precondition for this.

In summary the free sharing of software is inherent to the scientific method. That explains why the Free Software movement has its origins in the academic world and nowadays the scientific community is one of the main participants in Free Software projects.

3. Free culture

The free sharing of cultural expressions has nowadays become an act of “piracy”. While internet and peer-to-peer technologies have enabled the most efficient distribution network in a way unimagined a generation ago, strict conditions imposed by those 'culture factories' that monopolise the music and movie industries have limited their use. To win their war on piracy, these media and software conglomerates have used two important tools: 1) legal measures to limit the options of legally copying, sharing and studying; and 2) technical measures to effectively limit these “illegal” uses.

Regarding legal measures, copyright has become the main tool to limit the free sharing of cultural expressions. While the original copyright act was designed to stimulate authors to produce original works by granting a temporary monopoly to the author for a limited amount of time (14 years initially), nowadays both the duration and scope of this temporary monopoly have been extended to 50 (or 70 in the EU and US) years after the death of the author⁴. More importantly, the extension of the scope of copyright has led to limit the use of any previous creation in such a way that any new production becomes dependent on lawyers⁵.

3 <http://www.fsf.org/licensing/essays/free-sw.html>

4 The Berne Convention is a copyright agreement which was first adopted in 1886 and served as the basis for the current TRIPS agreement:

http://en.wikipedia.org/wiki/Berne_Convention_for_the_Protection_of_Literary_and_Artistic_Works

5 The use of patents on (software) ideas imposes other serious restrictions on the sharing of knowledge and limits the possibilities for software developers in general to create new programs. Interesting starting points for reading are provided by the Foundation for a Free Information Infrastructure: <http://www.ffii.org/>, the Electronic Frontier Foundation: <http://www.eff.org/IP/> and the Free Software Foundation Europe: <http://www.fsfeurope.org/projects/swpat/swpat.en.html>

Lawrence Lessig, for instance, explains in his book *Free Culture* about a filmmaker who was making a documentary and wanted to include a shot of a TV set (Lessig, 2004: 95). The TV set was visible just in a tiny corner of the screen and happened to play *The Simpsons*. As this was thought to be a value-adding element, the filmmaker decided to ask for permission to the director of *The Simpsons*. In short, it turned out that the film was owned by media company Fox and, even if it was a shot of just four and a half seconds in the corner of the screen, they required ten thousand dollars as a license fee. This example shows two things: first, that the creators/authors are rarely the owners of the copyright and, second, that the reuse of existing creative material is made impossible or limited to just a few, those who are supported (and controlled) by big media companies (Liang, 2004: 9).

Regarding technical measures, particular control mechanisms have been introduced to limit the use and reproduction of particular products. Movies, e-books and music as well as software start being distributed with control mechanisms that allow people to use the work of art they just bought, only under the exact conditions of the vendor. That means that, even if you are the owner of an electronic device, it is ultimately the vendor who determines what you may and may not do with it. For instance, recent file formats for music distributed to Apple's iPod music players only allow copying the music to just a few computers that use Apple software. In a similar way, electronic books and magazines can only be read by a particular user while copying and often printing are not allowed⁶. Such mechanisms are called DRM⁷, which corresponds to Digital Rights Management according to the recording industries or to Digital Restrictions Management according to digital rights user groups. If DRM is not stopped, it will affect every user of digital technology, as it is increasingly built into all sorts of electronic devices, including computers, media players, mobile phones and video games.

The answer to these legal restrictions comes partly from the so called copyleft culture. Copyleft is a play of the word copyright. Instead of restricting the users rights, copyleft guarantees the rights to use, copy and modify the works. It comes close to the concept of "public domain", which refers to works that can be used freely without copyright. However, copyleft differs from this concept in two ways: 1) the author remains the copyright holder and hence should be recognised as such; and 2) modified works should preserve the same freedoms. The concept of copyleft arose when Richard Stallman, founder of the Free Software Foundation, started to work on Free Software. Being an opponent of existing copyright legislation, he used the existing copyright legislation to flip its meaning from "all rights reserved" to "all rights reversed".

Copyleft has been especially prosperous in the field of software. However, in recent years the concept has been applied to other works of art and in consequence several license forms have derived from it. Creative Commons is one of the best known. It provides the author with a set of different options to distribute her/his work of art. Instead of "all rights reserved", the author can choose for "some rights reserved". For instance, authors can choose whether or not they want to be attributed, whether their work may be exploited commercially or not and whether it can be modified or not.

One might ask how creators are able to make a living if the copyright of their work is released under these free terms. First of all, as we said before, it is a myth that the creators are the owners of the copyright. In

6 Richard Stallman wrote a futuristic story about the right to read to show the horrible future if we do not act against these restrictive practices (Stallman, 2002: 73).

7 DRM refers to those technologies to control the access to and usage of data and hardware. More information about DRM can be found here: <http://drm.info/> ; <http://www.defectivebydesign.org> ; <http://eff.org/IP/fairuse/>

many cases, it is big media conglomerates that own the copyrights. The example of the Simpsons was, in this regard, very clear. Secondly, most creators make a living in other ways. For instance, musicians often make most of their money from live performances rather than from royalties on sales of their records (Liang, 2004: 10). In this regard, as many programmers and designers do, they sell 'services'. This would lead us to conclude that the parties that might see their interest damaged by the free sharing of cultural works are not the authors themselves but the media conglomerates that control the cultural market.

There are many examples on the use of copyleft or free content materials. These examples go from photo, music or video sharing websites to radio stations, scientific repositories or other knowledge bases. Here we will point out some concrete cases:

- Wikipedia⁸: a multi-language, web-based, free content encyclopaedia edited collaboratively by volunteers. Wikipedia builds upon the experiences from the Free Software movement in three ways: 1) the software of the platform itself is published as Free Software, 2) the articles are published as free content, under the Free Documentation License⁹ and 3) the encyclopaedia is a collaborative effort from a worldwide community.
- Eyespot¹⁰: a web platform where people can share and mix their photos, music and videos freely. It follows the ideas of Creative Commons where the cultural creation process as “rip, mix and burn” can be performed freely. It is a matter of taking the intermediary out: for instance, musicians don't have to negotiate with global record companies but they can build on the works of their peers freely. Entirely new creative communities have sprung up. See for more examples the Creative Commons website¹¹.
- Public Library of Science / PLOS¹²: a scientific journal on the internet where scientist can publish their results after peer review. Once published, the articles can be accessed freely, reprinted, translated and modified while assuring one legal condition: to accredit the authors and sources properly. A similar initiative is the ScienceCommons¹³ project that promotes the freedom to archive and reuse scholarly works on the Internet.
- Elephants Dream¹⁴: an interesting project where the Free Software spirit comes together both in software as in content is the short movie Elephants Dream, produced by Ton Roosendaal from the the Blender Foundation. The movie is produced with Free Software tools such as Blender and its content (footage) is completely available under a Creative Commons license. The result is that several groups have reused its footage to create derivative versions of the movie, an excellent example of the new possibilities of sharing knowledge and creating Free Culture.

4. Free Software economy

Until the 1980s software was not considered a product that could be made profit on. However, since then onwards companies tried to control software and earn more money by turning it into proprietary software. The proprietary software model, as any business, seeks to establish monopolies in order to maximize the

8 <http://wikipedia.org>

9 <http://www.fsf.org/licensing/licenses/fdl.html>

10 <http://eyespot.com>

11 <http://creativecommons.org>

12 <http://plos.org>

13 <http://ScienceCommons.org>

14 http://en.wikipedia.org/wiki/Elephants_Dream, <http://www.elephantsdream.org/>

profit. This tendency has a particular development in software. Since software is used to exchange information with other people, it has a “viral” or network effect: the more people use the same software, the better the exchange of information. This network effect combined with the proprietary aspects (one company owns the software and restricts its distribution) reinforces - more than in other markets - the consolidation of software monopolies.

In current software markets we can indeed observe this mechanism and conclude that near-monopolies control big chunks of them. A clear example is the desktop market where one single player controls more than 80% of the worldwide market. Governments and companies realise that they are dependant on one or just a few software vendors, what has been called vendor lock-in. This vendor lock-in manifests itself in three main aspects. Firstly, the continuous development and modifications are limited. Secondly, users are forced to use other applications from the same software vendor as this is the only intuitive option to integrate the software (due to the proprietary aspects of the software). And thirdly, the vendor lock-in makes sure it is hard or costly to migrate to alternative software solutions.

The drawback of this vendor lock-in is mainly two-fold. On the micro level, it makes the entry of new software companies in the market hard or in many occasions impossible. Since one player controls the market, the level of competition is low and thus diversification of products and services is limited. Therefore innovation of new products and services can only take place in a very controlled way. Besides the technical limitations to introduce new products on a closed platform, innovators face the risk of the monopolist software vendor to substitute their innovation as soon as it shows to be successful. Finally, the proprietary software lock-in forces customers to pay enormous amounts of money for the software licenses or services: since opting for an alternative is – or seems to be – hard, the dominant vendor can demand enormous prices.

On the macro level, this means that, instead of investing in local knowledge and skill development, money is mainly transferred to foreign software vendors. For instance, European countries pay tens of billions of Euros every year to companies in the United States only for software licenses. In the context of third world countries, this implies an excessive economic drain. To illustrate this, just look at the retail price¹⁵ of Microsoft Office in comparison to local wages. Note that the retail prices are similar around the globe. In Nicaragua, where 45% of the population earns less than a dollar a day, the purchase of a standard Microsoft Office license would cost them more than a full year's salary. In the Netherlands, the average person would spend only 5 days working to obtain the same license. As most non-Western countries cannot afford to be subject to this tremendous annual drain, they remain locked-out of the technology – or remain at the sideline - and can hardly participate in knowledge development around information and communication technologies.

An important step to get out of the vendor lock-in situation is the use of Open Standards. An open standard is a freely available specification of a file format or protocol that can be implemented by other software developers without restriction (European Interoperability Framework, 2004: 9; Bruce Pehrens, Open Standards - Principles and Practice). Open standards are maintained by a not-for-profit organisation which follows an open decision-making process. Examples of successful Open Standards are the internet protocols TCP/IP, HTML for web pages, OpenDocument for office documents, the SIP protocol for Voice over IP, etc. Using Open Standards guarantees the users choice: the possibility to change to competing software that implements the same standard. Since the information is kept in an open format instead of a secret one, the

15 See for a comparison of office suites including prices: http://en.wikipedia.org/wiki/Comparison_of_office_suites

information can always be accessed or converted in some future.

Most Free Software supports one or more Open Standards. Although these are different concepts, Free Software tends to use and help define Open Standards. The reason is clear: Free Software consists by definition of publicly available specifications, while the availability of its source code promotes an open and democratic debate around these specifications. When a standard is considered an open standard, no restrictions apply for its implementation so it can always be implemented in Free Software. Free Software and Open Standards make a winning team, as can be learned from the success story of the internet. Internet has become so successful because it is based on Open Standards and all of these standards have been implemented in Free Software. It must be said that not all software that runs the internet is Free Software, but the fact that for all protocols and standards do exist implementations in Free Software assures freedom of choice and avoids the negative lock-in effects we have discussed earlier.

Therefore, I argue that Free Software is the ultimate paradigm to avoid vendor lock-in. By definition, Free Software limits the emergence of monopolies since everyone has the freedom to copy and create new software projects based on the existing Free Software. On the micro level, this creates equal opportunities for all market players, both existing and new entrants, and makes sure that different players compete on quality and specialisation. On the macro level, Free Software enables governments to invest in local knowledge development instead of buying software licenses overseas. In this way, local companies can be contracted for development, implementation and support. Interesting examples are seen in Brazil, China, India and many other countries, where governments proactively pursue a Free Software strategy.

A question that may have risen when reading this article is: “how do people make money with Free Software?” The answer is with services. Free Software development is typically demand-driven: developers are paid for the programming of new parts or customisation of existing software. In other words, companies and individual specialists earning a living with Free Software usually offer services for development of new or adaptation of existing software, for support, translation and training. Many companies make money out of these services, from extremely large companies (e.g. the largest IT company, IBM, makes billions of dollars using GNU/Linux and other Free Software) to individual specialists. This fits in a broader trend that has been observed by market analysts: that the software market is developing more and more towards a “service-based economy”.

Another question that remains to be answered is: “how long will our economies remain dependent on software vendors that have locked us into their proprietary platforms?” The answer lies not only in the hands of the market but as well in the hands of governments. This is at least how it should be. First of all, by setting the example. Governments, as launching customers, can give a huge impulse to the generalised adoption of certain formats and software. Secondly, as common in other kinds of markets, by introducing rules and regulations to avoid market dominance and stimulate innovation. An example of that is the EU anti-trust case in which Microsoft was convicted guilty of several anti-competitive practices¹⁶.

5. Free Society

Free Software is essential for transparency, social inclusion and democratisation of information. Being these three aspects fundamental for the functioning of any free society, Free Software becomes hence

¹⁶ http://en.wikipedia.org/wiki/European_Union_Microsoft_antitrust_case

indispensable. In the following paragraphs, I will argue how and why.

First of all, it is a fact that growing number of government functions are represented in digital processes. For these particular processes, we need to have transparency. That means, in relation to the software, that we need to know how it works. In other words, we need to know what happens when we click a button: what does the system do? how is the desired output obtained? what happens with the information that we insert into the system? Apart from the criteria of cost-effectiveness that any organisation has, government organisations have some additional requirements. Peruvian Congressman Edgar Villanueva identified the following criteria for governmental software services and processes¹⁷: 1) Free access to public information by the citizen; 2) Permanence of public data (information durability); and 3) Security of the state and citizens.

To guarantee these requirements, three main aspects are indispensable. First of all, the encoding of data should not be tied to a single provider. The use of Open Standards and open formats gives a guarantee of this free access, if necessary through the creation of compatible Free Software. Secondly, the usability and maintenance of the software should not depend on the goodwill of the suppliers, or on the monopolistic conditions imposed by them. For this reason, the state needs systems whose development can be guaranteed due to the control over the source code. Thirdly, to guarantee national security, it is indispensable that the state relies on systems that do not allow control from a distance nor transmit undesired information to third parties. The required knowledge of the source code would lead to the elimination of the growing number of programs with 'spy code'.

Secondly, Free Software promotes social inclusion and hence works against any further development of the so called digital divide. In contrast to proprietary software, Free Software by definition does not exclude people. It does not exclude people from using the software because they cannot afford to pay the license costs, from sharing the software with others or from studying how it works and hence from advancing their knowledge on software engineering. As observed by many analysts, a huge knowledge gap exists nowadays in our societies between those who have access to the internet and software technologies and those who have not. This so called digital divide is not only observed between the South and the West but also within the 'advanced societies'. Although it is not the whole solution, general access to software through the adoption of Free Software would thus break down part of the mechanisms that reinforce increasing differences between those who have knowledge and access to the tools of the digital age and those who have not.

Thirdly, Free Software contributes to the democratisation of information. As observed by Castells, the expansion of the internet has enabled a democratisation of information on a global scale. "If information technology is the present day equivalent of electricity in the industrial era, in our age the Internet could be likened to both the electrical grid and the electric engine because of its ability to distribute the power of information throughout the entire realm of human activity." (Castells, 2001: 1) Castells refers to the internet as the technological basis for the organizational form of the Information Age. While the internet is the network that connects us all, it is Free Software that provides the tools to move freely in this worldwide network.

On the server side, the internet is formed in large extent by Free Software infrastructure: to name a few, 60%

17 Reply to a letter from Microsoft, translated into English: <http://www.gnu.org.pe/resmseng.html> Overview: <http://www.gnu.org.pe/preyres.html>

of webpages are served from Apache web servers, emails are sent through sendmail and domain names are managed by DNS-servers called BIND. All these are Free Software applications. On the user side, Free Software makes available all kinds of tools (such as web browsers, email programs, website content management systems, chat servers, mailing lists, etc.) to participate in the digital age. This is fundamental since, for a real democratisation of information, we need not only the existence of the network but also free tools to access, circulate and participate in this network.

6. SELF education platform

In this article I have argued the importance of Free Software in the field of science, its implications on culture in the digital age, the need to get out of the vendor lock-in by using Free Software and Open Standards and its role for the development of free societies by facilitating transparency, social inclusion and democratisation of information.

If Free Software is such an important concept and provides so many practical tools, how come that still so little people are using or even aware of it? What is holding back the massive adoption in all these fields and what can be done to change that? As a group of international experts has observed in the Barcelona Declaration for the Advance of Free Software¹⁸, some of the most important factors that hold back the adoption of Free Software are: 1) the lack of awareness about Free Software; 2) the perceived lack of professional support. Support may exist, as offered by large and small service companies, although many people have the idea that it is hard to get; 3) the lack of qualified teachers that are able to train students in the concepts and tools of Free Software and 4) the lack of quality educational materials about Free Software.

To address these issues, education is one of the crucial areas of focus. I mean education in a broad sense, that is, education of students and teachers in schools and universities, training of employees in all kinds of organisations and training of system administrators. Only by teaching the future and current users and teachers, it is possible to raise awareness and contribute to the building of a critical mass. Moreover, there is another reason why Free Software in education is a logical choice. In education – at least when publicly financed – we expect the students to be taught concepts and vendor neutral applications. If students learn concepts, they are usually better prepared for changes and new developments. Free Software does just that: it is by definition freely available software, not owned by one company and permits – or normally stimulates – the study of its concepts and functioning.

For these reasons the SELF Project¹⁹ has been initiated to provide an educational platform about Free Software and Open Standards. In short, it is a worldwide platform for the collaborative production and sharing of freely available educational and training materials about Free Software and Open Standards. It addresses the issues mentioned in this article in various ways:

- it is a technological platform that is based on Free Software
- the formats and standards used in the platform are commonly used Open Standards in order to set an example and maximise interoperability with other platforms and tools
- the platform aims to provide educational and training materials about Free Software and Open Standards in order to facilitate the interested users from all communities to get out of the vendor lock-in and use free technologies

18 <http://www.uoc.edu/activitats/docbcn/eng/docbcn.html>

19 <http://selfproject.eu>

- the technological platform permits both reading and publishing in new, innovative ways
- its materials are free content or, in other words, use the philosophy of copyleft as derived from the Free Software movement
- the platform is supported and used by a worldwide community of interested users and strategic alliances.

The SELF Platform has been initiated by an international consortium of seven partners in Europe, Asia and Latin America: Internet Society Netherlands, Universitat Oberta de Catalunya (Spain), Free Software Foundation Europe (most countries in Europe), Gothenburg University (Sweden), Internet Society Bulgaria, Fundación Via Libre (Argentina) and Homi Bhabha Centre for Science Education (India). The European Commission supports the SELF Project for the initial set up of the platform and the first educational and training materials.

The SELF platform aims to be a community of all interested parties related with educational and training materials on Free Software and Open Standards. This means to include not only the main target groups of the educational materials, such as educational institutes, training organisations and local and national governments, but also publishers, IST companies and Free Software communities. All of these actors will benefit from their participation in the SELF platform while at the same time adding value to the SELF platform by collaboratively developing new or improved materials:

- **Universities or schools** have the need to develop specific course materials. By collaborating with other interested experts through the SELF platform the production of such materials becomes more feasible.
- **Training companies** can benefit by using training material from the platform and at the same time delivering part of their own training materials to the SELF platform. These materials can then be improved and translated by the whole community, which again provides value for the original company as well as others.
- **Publishers** may take materials from the platform to publish these in their books. At the same time, they might add part of their own materials to the SELF platform.
- **Industrial manufacturers** with hardware products as a strategic asset can also use Free Software – to be used on their hardware – as a means to offer a more competitive product. For these companies, training and educational materials on Free Software are hence essential. In many cases they develop these materials internally. If they provide these materials to the SELF platform, they can gain recognition while getting their materials improved and/or translated and having the possibility to get new ones.
- **Free Software communities** can benefit from the platform by providing general information on their Free Software applications as well as making training material available that will facilitate the adoption of their software.

In general terms, we can conclude that those who participate in the platform can improve collaboratively their own contents while gaining prestige or authority within the community. At the same time their participation will help to improve the SELF contents and enhance the SELF community. The final result is that the community model of the SELF platform will assure a “win win situation” for those who provide efforts to the platform and those who use it. It is also important to note that the multidisciplinary and

multisectorial character of all the actors involved will facilitate new initiatives and ideas that otherwise would not have been raised that easily.

In short, the commitment of all these groups as users and strategic partners is essential to make the SELF Platform the leading platform for educational and training materials about Free Software and Open Standards. This will guarantee the ongoing character of the Platform as well as a connection with real needs. As a mode of conclusion, this article should hence be understood not only as a discussion of the role of Free Software in the sharing of knowledge but also as an invitation to participate in the SELF Platform.

8. References

EUROPEAN COMMISSION. European Interoperability Framework. IDABC Website, 2004. URL: <http://europa.eu.int/idabc/en/document/3761>

LESSIG, Lawrence. *Free Culture: how big media uses technology and the law to lock down culture and control creativity*. New York: The Penguin Press, 2004. URL: <http://www.free-culture.cc/>

LIANG, Lawrence. Guide to open content licenses. V1.2. Rotterdam: Piet Zwart Institute, 2004. URL: http://pzwart.wdka.hro.nl/mdr/research/liang/open_content_guide

PEHRENS, Bruce. *Open Standards: principles and practice*. URL: <http://perens.com/OpenStandards/Definition.html>

STALLMAN, Richard. *Free Software, Free Societies*. 2002. URL: <http://www.gnu.org/doc/book13.html>