Matthias Studer[1]

Gift and Free Software[2]

"Creation is a pleasure, to be given, not exchanged"

Renoo, a Linuxian

Translator's introduction

The free software movement would have delighted Marx[3]. What clearer example of progress in a mode of production that dynamites the social relations (in the present case, capitalist social relations) of production?

However, in this article, Mathias Studer prefers to analyze this movement in terms of another convergent interpretation of social relations: the theory of gift exchange developed by the M.A.U.S.S. (Mouvement Anti-Utilitariste en Sciences Sociales), a network of researchers developing the insights of the founder of the French school of anthropology, Marcel Mauss. Relatively unknown in the Anglo-Saxon world (see a first attempt to present it in Olivier de Marcellus' article in *The Commoner* N. 6), it approaches the same vital heart of sociability as reflections on the commons, from a complementary angle. To be read at the same time as "Virtual Enclosures" ! (OM)

Introduction

In a world of ever more pervasive individualism, in which money takes a more and more important place in our lives, a community, whose members call themselves hackers has come together to create an information system that they offer, potentially[4], to the whole world. A phenomenon that is strange, first, for its gratuitousness and second, for the importance[5] that it has come to have among programmers. But the real originality of the hackers stems from the social process, the particular kind of life that they lead. In this article, we will try to explain what free software is, but also to illustrate certain dynamics which regulate this community. We will see that this daring bet, the bet on free cooperation, is based on a particular culture and ethic, and on a culture of gift exchange.

First, one must understand the difference between hackers and crackers. The cracker is a computer pirate[6], who breaks into Internet sites, who copies software illegally. The hacker is an enthusiast, devoured by curiosity, ready to spend hours to solve a problem - actually seeing it more as a challenge than a problem. It isn't necessary to be a programmer to be a hacker. One can very well be a hacker in

[1] matthias.studer@ezwww.ch

[2] Translated from French by Olivier De Marcellus

[3] You can find interesting discussion about Marx and Free Software here : http://www.oekonux.de/ (Author's note)

[4] Potentially, since it is first necessary to have a computer and an Internet connexion.

[5] Some projects have hundreds of participants. Debian has about 1000, Gentoo 600.

[6] The Free Software Foundation (FSF) observes correctly that copying software isn't really very like murdering on the high seas. It would therefor be more correct to speak of an illegal act – or even of « sharing information with your neighbors ».

philosophy or astronomy, for being a hacker is mainly a question of attitude. An attitude that we will also try to illustrate.

But before beginning, it is first necessary to define the properties of information goods and particularly of software.

1. Software

A software is programmed using a programming language (a series of instructions such as « If such condition, then do... »). This constitutes what is called the **source code**. This code is then compiled, that is to say transformed into a binary machine code (a series of zeros and ones), so that it can be used and understood by a computer. Practically, no one can read binary code, which guarantees to the companies that no one will plagiarize or redistribute a slightly modified product. A distinction is made between software whose source code is open, that is which are distributed with the source code, and software whose source code is closed, that is to say not available. The great majority of commercial software, such as Windows, Word, etc., are in closed source code.

What distinguishes software from other goods is that they are not scarce. One speaks of a non-scarce good when the fact that I possess it does not preclude my neighbour from possessing the *same good at the same time*. Thus, for example, a table is a scarce good,[7] because another person cannot have the same table without dispossessing me of it. This is not the case of OpenOffice[8] for example. The fact that I am using OpenOffice to write this text in no way hinders its use at exactly the same moment by thousands of people all around the planet. There is no real difference between the OpenOffice program that I am using and those used by others.

Another good example of a non-scarce good are movies. Effectively, movies cost a lot to make, even without speaking of Hollywood super-productions, but copies cost comparatively almost nothing. Just as with software, when I make a copy of one, I don't take anything away from its owner, and there is no way of distinguishing my copy from my neighbour's, apart from the physical substratum (the cassette or DVD). There is also no limit to the number of copies that can be made. The fact that I make 10,000 copies doesn't mean that there are 10,000 less for others. This isn't the case for tables. If I make 10,000 tables, there will be that much less materials available to make others.

Non-scarcity comes from the fact that reproduction costs very little, which is not the case when the vector of information is more material (as with books for example). Also, the important part is not the material substratum (the cassette or DVD), but what is on the media (the movie). This specific characteristic projects us into a world of abundance. To put it another way, the notion of scarcity, so important to capitalist society[9], is eliminated. Or if it exists, it is because some artifice (a patent or license, etc.) has been enstated in order to create it[10].

[7] It is interesting to note that the french term, *bien rival* (rival good) is in line with a Hobbesian vision. That is to say that it is necessary to struggle for resources in a universe of scarcity and lack.

[8] OpenOffice is a free software, more or less the equivalent of Microsoft Office, which is available on line.

[9] This notion is essential, since it determines prices. It must be added that the idea of rarity, generally associated to that of lack, founds Hobbes' and Locke's discourse legitimating property and a State to defend it.

[10] For more on this, see Bruno Lemaire and Bruno Decroocq, *Microsoft pris dans la toile… Chronique d'une mort annoncée*. On line at : http://www.adullact.org/IMG/pdf/doc-157.pdf

2. Free software

The project of free software was born in 1984 (!), with the publication of the GNU[11] manifesto by Richard Stallman[12]. This manifesto was a reaction to a change in the practice in programming. Until that time, software – even commercial software- was in general free (in the sense of open source) although the term « free » wasn't used. The important point was that programmers were « free to cooperate », to quote the Free Software Foundation (FSF)[13] and that this was common practice. In the 1980ties, software become proprietary. One speaks of proprietary software when the owner of the software has the possibility of restricting its access or utilization, or when the source code is not available. Proprietary software blocks cooperation between programmers, particularly by blocking the access to the source code. In other words, the GNU project was intended as a defence of the *liberty to cooperate, the defence of an ethic*, but it was also the defence of a community based on cooperation. As we shall see, this ethic played a central role in free software.

All computers need an operating system[14]. It was thus the starting point of the GNU project, as it was a total project. The goal was total liberation from proprietary softwares and thus « to make proprietary software obsolete »[15], a strikingly ambitious project. For several years, members participated writing the operating system. It was at that time a relatively small group. In the 1990ties, all that was still missing was the kernel[16]. It was then that Linus Torvald launched LINUX, which provided it. Linux was to transform free software in several ways. First, with this kernel, the system became a complete operating system. Second Linus had the genius to create a structure in which everyone could participate. The apparition of Linux is thus also linked to the generalization of Internet. It is the beginning of real, networked collaboration. In that sense, Linux is also the child of Internet.

2.1 But what is free software, exactly ?

As we have seen, free software gravitates around the idea of liberty and cooperation. Thus the central point is not the gratuitous aspect which the english term « free » software tends to make one think of.[17]

[11] An abbreviation standing for both Gnus Not Unix and Gnu, the logo of the GNU project. The acronym is thus recursive or self-referential, a fundamental aspect of programming), and as such a sort of programmer's pun.

[12] A computer scientist then working at MIT. He later became the thinker of free software. He is now president of the Free Software Foundation (FSF).

[13] Free Software Foundation. A foundation created by Richard Stallman to take charge of the legal aspects of free software, in particular to take legal action in case of violation of licenses.

[14] The group of programs necessary to make the machine run.

[15] These points are cited in a FSF document available at : http://www.gnu.org/gnu/gnu-history.html

[16] Very important program, which in particular organizes the writing of files, the drivers, etc.

[17]In french there are two words corresponding to « free » : *gratuit* means without cost and *libre* means free. The term logiciel libre thus emphasizes the political, subversive aspect of these goods, rather than their convenient price. Has english been particularly colonized by capitalism ? This was the problem that the founders of free software addressed when they specified, « Free as in « free speech », not as in « free beer ! ». » Perhaps they should have counter-attacked, and spoken of « liberated software » or « freedom software » (as in « freedom fries » !). (translator's note)

In fact, free software is sometimes sold for a price. Free software is the affirmation of four liberties, considered fundamental [18]:

- The freedom to run the program, for any purpose (freedom 0). [19]
- 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.

This definition of free software implies that they can be sold, but that the person who buys one can freely distribute or modify it. It is interesting to note that the definition centers a lot around the community and the relation to others. It is concerned with giving the liberty, to transmit this liberty, so that others can profit from it. This is important in the measure that what is given is not only the software but *also a meaning stemming from the fact that the software is given*. This meaning is an affirmation of the liberty of the other, the recognition of his (or her) specificity (particularly with respect to liberties 0 and 1). By giving the liberty to re-distribute it also encourages the other to transmit in turn this liberty and this meaning.

Free software very often is accompanied by the concept of *copyleft*. Copyleft is a juridical provision which assures that all the modifications and above all the redistributions of the software will also be free, and guarantees the four liberties cited above. In practice, this is done by establishing a copyright (hence the term copyleft) and a license guaranteeing the legality of these principles[20]. Copyleft is thus an affirmation that no one can become proprietor of the code, that no one (including its author) can appropriate the code.

3. Hacker culture

Why do hackers offer their code *for free* ? Why do they give their time ? How did such a phenomenon become so widespread ? In other words, what are the hackers' motivations ? And finally, who are the hackers ? Questions for which we can only offer some partial answers. A famous hacker, Wozniak, summed up the hacker's ethic in the formula $H=F^3$, which stands for Happiness = Food, Friends and Fun. We shall examine these three aspects in more detail.

Hackers generally touch no salary for participating in projects. Their participation is rendered possible by the fact that a part of their fundamental needs are considered covered and need not be worried about. Effectively, the great majority of hackers come from industrialized countries. They tend to be from the middle (or even upper) classes, although they don't necessarily have diplomas, many of them being self-taught. After this brief treatment of their social condition, we would like to note in what ways the hacker culture differs from the dominant one. It is not a movement which recognizes itself in any particular political trend. Hackers can be right wing as well as left. However, by their practices, by their way of conceiving their relation to work, they seem to place themselves in a non-capitalist perspective.

[18] These definitions are taken from the GNU site at : http://www.gnu.org/philosophy/free-sw.html

[19] In programming, one generally starts from zero, rather than 1.

[20] The most usual being the GPL or General Public License.

One of the motivations of hackers is pleasure, the pleasure of programming, programming as hobby. However, even if hobby is the term they usually use, it does not correspond to its common usage. Effectively, as Eric S. Raymond notes « It is very amusing to be a hacker, but its an amusement that demands a lot of effort »[21]. Linus Torvalds, the founder of Linux, speaks of it similarly, « Linux has mostly been a hobby (but a serious one, the best of all) »[22]. He goes even further, affirming pleasure as a central value, « but Pleasure with a capital P, the Pleasure that gives meaning to life. »[23].

This conception of work as pleasure is opposed, according to Pekka Himanen, to that defined by the protestant ethic, as defined by Max Weber. « God dedicates us to action [...] work is the moral and natural end of power »[24], the protestants seem to say. What seems important is not working to live, but literally living to work. It is the affirmation of work as a finality. Work becomes that which links us to society and makes us feel recognized by it. Pleasure and leisure thus come to be considered idleness, a moral degradation. Among hackers, what links is of course a common production, but above all the pleasure of programming and the recognition of that pleasure by others. The notion of work changes, it is no longer toil, but work as passion, self-realization. This implies another conception of work with respect to leisure, as Bruno Lemaire and Bruno Decroocq note, « the pertinent distinction is not, is no longer, between work and leisure, but in the interest that one has for one or another of one's activities. »[25]

But it is also the affirmation of another relation to time. Whereas in the protestant ethic there exists a time for leisure and a time for work, the hacker ethic poses a time of creativity, necessarily unpredictable. Thus it is not rare that a hacker spend several days (and maybe nights) on a problem, before taking time out for a beer with friends or a computer game. The hacker ethic is an ethic of creation (which does not mean that there aren't relations of domination and sexism in this community).

But the pleasure involved is also social, the pleasure of constructing something that is valued socially. "One feels good when one has made something that people like to use" says Linus Torvalds. We will go into this further in the next part.

4. A gift network

We have seen that the gift is a central concept in free software. But what is a gift?

Does gift rime with gratuity ? We will briefly present what gifts signify in our everyday relations, in the light of the work of Jacques Godbout, Francoise Bloch and Monique Buisson[26]. We will then try to characterize how it is experienced in free software communities. Finally, we will examine in the light of gift exchange how things happen in the case of proprietary software and the market in general.

[21] Eric S. Raymond is a hacker who maintains the fetchmail project (an e-mail software). He was one of the first to analyze the reasons for the success of Free Software and its social organization.

[22] Unless otherwise specified, the citations of Linus Torvalds are taken from an interview done by Rishab Aiyer Gosh, *Qu'est-ce qui motive les développeurs de logiciels libres?,* 1998 First Monday.

[23] Pekka Himanen, *L'ethique hacker et l'esprit de l'ère de l'information*, Exils, 2001.

[24] Baxter, Christian, Directory, cited by Max Weber in *The Protestant Ethic and the Spirit of Capitalism*, p. 255 n. 2, and p. 257, n. 1.

[25] Bruno Lemaire and Bruno Decroocq, *Microsoft pris dans la toile... Chronique d'une mort annoncée*

[26] J. Godbout is a member of the M.A.U.S.S. group (Mouvement Anti-utilitariste en Sciences Sociales). F. Bloch ant M. Buisson are researchers at the CNRS (Centre National de Recherche Scientifique).

4.1 To give, receive and exchange

What is a gift ? From a juridical standpoint, gifts differ from exchange by the fact that the giver refuses his right to demand an equivalent in exchange. Gift can thus be defined negatively with respect to market relations. But gift can certainly not be limited to that perspective.

Gift is relatively present in primary relations[27]. There are of course the presents, that we are the most inclined to consider as gifts. But there are also the divers aids given to people close to us. The help can take a variety of forms, from talk to money, or helping someone to move house. It is very present in the family, as is hospitality. Gifts can also take the form of service or of material aid.

With people who are close, gifts are made in the *name of the relationship, of the tie.* The gift is determined by the relation between the person who gives, the donor, and the one who receives, the recipient[28]. The gift is made because there is a tie of friendship or obligation with the person. Of course, the tie can be of several different sorts. Thus in a family relationship, one can give because one feels obliged to, and at the same time because one appreciates the recipient.

But if the gift is made in the name of the relation, it is also made *as a function* of it. That is to say that the gift must *make sense* in the relation. One doesn't offer the same thing to everybody one, or at any rate, not in the same way. It all depends on how one perceives the other and on the type of relation that one has with that person. Thus when someone needs help, often several people will intervene, but they won't all necessarily give the same thing. There is not an equality between the donors. Rather, each will act according to the relation that he has with the recipient and/or according to his possibilities. Once again, this relation may be of an obligatory nature, either because of group pressure or because of the nature of the relationship between the persons involved.

When one receives, one generally feels "in debt" towards the donor. The recipient has the impression that he "owes" something to the person who gives. This is an everyday experience. How many times have we invited someone who has invited us? Or given a gift in order to say thank you? It is important to note that the feeling of indebtedness towards a donor is not as such a negative feeling. According to Piaget, we even seek to be in debt: "[...] one never claims all that one is owed, and one never pays all one's debt: the circulation of social value reposes, on the contrary, on a vast, enduring credit, constantly diminished by usury and forgetfullness, but also constantly reconstituted."[29] Obviously, it can be experienced as an obligation and as a restriction of one's liberty, but it is also an affirmation of one's recognition of what the Other has done for us. A powerful recognition of what links us to the Other, and which in turn will push us to wish to give. Recognition must be understood in two senses: recognition of what the Other has given, but also gratitude for it.

We have seen that the gift varies according to the situation, relation, etc. That is because the gift is a bearer of *meaning, of symbolism*[30]. If we speak of symbolism, it is because the gift bears meaning which goes beyond the tangible gift. It is a sign which indicates another, possibly inexpressible

[27] By primary relations, we mean the network of family and close friends. It is a network with undefined borders. The family part has been much more studied, because it can be more clearly defined.

[28] More exactly, the donor is the person who is in the position of giving, while the recipient is the one in the position of receiving.

[29] Cited by J. Godbout, *Le don, la dette et l'identité homo donator vs homo oeconomicus*, La découverte, 2000

[30] Meaning and symbolism are necessarily social. It is not an individual dimension, in which case neither meaning nor symbolism would be comprehensible to others. They are like language, which must be shared in order to be comprehensible. Individual languages don't exist.

dimension. Thus the gift is not only an object, it transports a meaning which is affirmed in the relation. According to Marcel Mauss, it is this that will push the recipient to reciprocate: "that which obliges in the gift received, exchanged, is that it is not inert. Even abandoned by its owner, it conserves something of him"[31]. In other words, it is the affirmation of the donor's subjectivity within the relationship. Thus when the recipient returns, or rather gives in turn, he shows that he accepts the meaning given by the donor. "To give is to return and to return is to give"[32] say F. Bloch and M. Buisson. It is not so much a case of returning but of in turn giving, in order to affirm one's own subjectivity in the relationship. Similarly, by giving, the recipient participates in the creation and development of the relation. It is an unending cycle, a dynamic inscribed in the relation between the actors. The first gift can be understood as a call, an invitation to the Other to enter into this dynamic, to begin the dance of gift-exchange. The recipient is free to refuse to enter into the dynamic, to refuse the dance.

It also happens that the recipient cannot refuse the gift. This can be because of the particular nature of the relationship or of "social pressure", in which case the recipient will tend to feel "obliged" to enter into the dynamic. In that case the gift does in fact constrain the recipient.

It also occurs that the gift is not made with a recognition of the Other, of his otherness. It is the kind of gift that oppresses, which affirms only one side, what one calls prestige gifts. In those cases, the dominant motive is to affirm the donor's prestige. There is no recognition

of the other, because he doesn't have the possibility of returning the gift. The recipient will not be able to affirm his own subjectivity and will thus feel flattened by the affirmation of the donor. Receiving gifts can thus be dangerous.[33]

If a gift does not *make sense* in the relation, for one or the other of the actors, it is only given vague attention. Consider for example, the vase that one received for Christmas and which ended up in the cellar. Couples often illustrate this more explicitly. If the man thinks that doing the dishes is a gift, it is because he puts this meaning on it. But for the woman, doing the dishes may be totally normal, something that is done as "a matter of course". The fact that doing the dishes is not considered as a gift may become a source of conflict, because the man will feel that what he has brought isn't valued. He won't feel *recognition*, gratitude, from the other. The woman may well also feel that the man's attitude is unacceptable. Thus the recognition of what constitutes a gift can be a source of conflict.

4.2 Gifts and free software

Thus in the case of free software, giving can be understood both as a call for recognition (Hey there! I exist!) and as a recognition of the community, of what it does, of its ethics and symbolic. When one gives in free software, one doesn't necessarily give to "everyone, everywhere"[34], to quote the FSF, but probably first of all to the project in which one participates, secondly to a larger community, idea and *ethic*. To a sort of regrouping of all the free software projects. Of course, this larger whole doesn't really exist concretely, it has no center. One could say in a way that one gives to a culture, one thus

[31] Mauss quoted by F. Bloch and M. Buisson, *Du don à la dette : la construction du lien social familial*, Revue du M.A.U.S.S., N° 11, 1991.

[32] F. Bloch and M. Buisson, *op. cit.*

[33] Gifts of organs are a similar case. The recipient has no way of returning (since the donor is generally dead) and in a situation in which the symbolic weight of the gift (the gift of life) is important for him. On this example, see in particular Godbout, *op. cit*.

[34] FSF, http://www.gnu.org/philosophy/free-sw.html

recognizes its existence, and that it in turn recognizes one's own existence by the feedback that its members provide. Basically, as Bruno Lemaire and Bruno Decroocq so well put it, in this culture " I give, therefor I am"[35].

But "giving to everyone, everywhere"[36] has also another meaning. It is also another *call*, a call to all to enter the dance, the dynamic, of free software. One is not obliged, but invited to enter this dynamic, if we understand its meaning.

Liberty is essential in the gift because it is what distinguishes what is given from what is owed. The owed is what is a matter of course, what is normal. The liberty of giving adds more value to the gift, so to speak, by permitting the affirmation of a particular meaning by the donor.

Liberty is present in free software in various ways. First, the call made does not oblige. The recipient is free to enter into the dance or not. He is free to respond to the gift. But he is also free to make his own gift, to transform and transmit what he has received. Sartre expresses well this idea already present in gifts: "One of the essential structures of the gift is the recognition of others' liberty: the gift is an opportunity (...) to transform what is given into another creation, in brief, into another gift."[37]. It is the recognition of the Other's specificity. After, if the recipient does indeed enter the dynamic, he is free to bring what he wants, to bring something of himself, be it by creating a new project, by participating in some way in the movement. That is one of the enormous strengths of the movement, to accept as gifts and enormous variety of contributions. Effectively, reporting a bug[38] writing help for a subject, participating in the activities of the website, proposing code, proposing a new functionality, or being a developer, coordinator, etc. : all these are recognized as gifts. Of course, these contributions are not considered at the same level or in the same way according to the community, but for a small community feedback is considered a gift, a recognition of one's identity within a community. Linus Torvald refers to this when he says "The whole project is constructed on the idea that everyone returns what he is able to return - and on the idea that small efforts make the great systems". Finally, the concept of liberty is absolutely central and related to gift when the FSF speaks of "the liberty to cooperate", the liberty "to help one's neighbor".[39]

One must note that the strength of gifts in free software is also to be sought in its longevity. Effectively, by the non scarcity of information goods (the fact that they can be indefinitely copied), one gives once, forever, to everyone everywhere. Which is obviously not the case with traditional gifts. This longevity allows one to be recognized in the whole community on the basis of a single gift.

4.3 The difference between market and gift exchange

Let us take the opposing view to that of economists, who generally try to explain everything in terms of the market, and let's try to look at the market in the light of gift exchange.

What is essential in gift exchange is the relation that it creates, whereas the market is interested only in the exchange of objects, and allowing immediate exit from the relation thereafter.

[35] Bruno Lemaire and Bruno Decroocq, *op. cit.*

[36] FSF, *ibid.*

[37] J.P. Sartre quoted by Godbout, *op. cit.*

[38] A bug is a problem in a program. A function that doesn't do what its supposed to, or simply blocks the program.

[39] FSF, http://www.gnu.org/philosophy/free-sw.html

With the market, the actors are supposed to act according to the theory of rational choice, that is to say utilitarianism. They are supposed to choose in a rational way the actions that maximize their preferences. This theory is based on two concepts: *preferences* (satisfaction) and *optimization* (maximization) of satisfaction[40]. The preferences aren't defined, they are individual. The optimization tells us that the individual will seek the action that permits the best results. Thus the individual is supposed to compare, in a rational way, each action and to choose the best : the one that is the most *useful*. Utilitarianism thus poses a clear distinction between goals (preferences) and means. It does not consider itself as a philosophy of finalities, because these are considered as given, but as a philosophy of means.

This distinction is a problem in many cases, in particular because the relationship between goals and means is not linear. Effectively, goals are frequently changed during action (during the means) and because of the means. But above all, the utilitarian conception engenders an exteriority with respect to action. Action is considered as a *tool* [41] by which one attains the goal that the person has fixed. That is precisely why people call it utilitarianism. Thus, I no longer take pleasure - to go back to Wozniak's definition - I use a programming tool in order to take pleasure. Or to take another example, if I study, it is to later earn a better salary. Studying isn't important as such, what is important is the goal of the action, that is to say the desired salary. In that case, it is not important to be active or passive as a student. What is important is to get the diploma. One is thus in a dynamic where what is important is not what I do, but a goal that is beyond my action.

This theory also supposes that we are supposed to compare actions in order to choose the most useful. This comparison can only be made through an equivalent, because it is necessary to have third object as a common reference or measure in order to compare two fundamentally different things. That is why the second norm of this theory is *equivalence*. It is necessary to establish the equivalence between actions in order to find which one is best.[42]

This conception can explain what happens on the market. This is due, among other things, to the fact that equivalence is present in the form of money. But this conception is also present in a much wider framework. Pekka Himanen notes very well that the handbooks of self-development, some of which have become real best-sellers, are based on in it in an almost religious way. Thus for example, negative emotions must be transformed into positive ones, ones that are *useful* with respect to the goals that govern us.

Utilitarianism is thus not only a theory, but is really the reflection of a practice. Our actions also depend on how we conceive of them, whether our own or those of others. In other words, action conditions theory and theory conditions our actions.

But even if utilitarianism is the official ideology of the market, the market often demands that one go beyond it. The labor market is a good example, since the employee is often expected to not only act in a utilitarian way, but to *give of himself* in his work. A person who doesn't give himself, engage himself, with what all that implies - and particularly the fact of giving meaning to his work - has a good chance of being fired, or at least being very badly considered.

We have seen that, with the market, the actors are exterior to the action. This exteriority can also be found in the relation to the object exchanged. That is to say that the object of exchange is also

[40] One also speaks of a function of utility, instead of satisfaction.

[41] That is the very definition of a tool : something exterior to the person that uses it.

[42] Thus action take the form of merchandise. It is generally measured by money, the universal equivalent, although this measure is often implicit.

considered exterior to the people concerned. We also saw that "something remains of him (the donor)", as Mauss says, in objects exchanged as gifts. That, among other things, is what obliges. The separation of subject and object does not exist with gifts. Note that this separation confers a certain "liberty" to actors exchanging on the market, since the object that one possesses after the exchange does not imply a relationship, a dynamic in which the actors remain involved. This brings Insel to say: " The pair constituted by individualism and neoclassical economy attempt to found an ethic of behavior of a man having no debts towards anyone. This founds the claim of this theory to be a discourse of liberty"[43]. This is manifestly not the same conception of liberty as the hackers'.

Gifts pose themselves in a non-utilitarian point of view. The gift must be taken for itself, within a relation, which is to say that the goals and the means tend to become confused. This does not mean that the gift is disinterested. It often occurs that the gift has a finality that goes beyond the gift itself. However, this finality must make sense within the relationship. We are no longer dealing with a tool, an exterior object, but with a relation to which one belongs. In free software, there is an affirmation of an ethics of a meaning which goes beyond the relation. There is thus the affirmation of a finality. But the finality is an integral part of the relation, that finality *makes sense* within the relationship et within the community. Of course it happens that the gift is made in a utilitarian perspective, but that is extremely badly looked upon and actors tend to withdraw from such exchanges. The utilitarian gift is the Machiavellian gift, which tends to be rejected by the recipient. It is a betrayal of meaning, because the real meaning is hidden and goes beyond the expressed meaning.

In addition, the gift tends to hold equivalence at a distance, or rather, equivalence applied to gifts means its minimization. Effectively, to pose an equivalence is to refuse the meaning of the gift and to only accept its material aspect, while emptying it of its symbolism. For it is impossible to establish an equivalence of meaning[44]. But doing so would also refuse the dynamic of debt. If one poses an equivalence, one says that one no longer feels in debt, that one is not one's turn to give, but simply that one is going to return what was given. The meaning conveyed is that one doesn't want to feel linked. This is common practice to protect oneself from relationships that one wants to avoid[45].

The different conceptions of the relation engender different behaviors within the relationship established. In the market, if a product no longer pleases one goes to the competition to see if the product is better, corresponds more to what one expects. There is no problem changing, because the merchant relation is only seen as a tool and the object of the exchange is also exterior to the relation. What is important here is not to develop a relationship with the producer, unless one is obliged to. It is what is called a possibility of *exit*, a possibility of escaping the relation to go look somewhere else. Of course, in some cases one cannot flee, for instance if one has to do with a monopoly. In that case, the only alternatives are to act within the relation, so that the partner will change his product (*exercising*

[43] Insel quoted by Godbout, *op. cit.*

[44] One can quote Seneca on this: " "Return what you owe" Well {this proverb} is exceptionally shameful with regards to gifts. What? Would one return life, if one owes it? Honor? Security? Health? Returning is precisely impossible for all great gifts. "At least, in exchange for that" people say "let us render an equivalent service." Well that is what I was saying: all the merit of such an eminent action will be lost, if we make a merchandise out of a gift [*si beneficium mercem facimus*] ». Quoted by Godbout, *op. cit.*

[45] Godbout quotes an interview that he made that expresses this kind of behavior very well: "At Helen's sister's { her sister-in-law} place, its generally I who made the meals, because I don't want to impose myself. It cost me 300 dollars for that week, because I don't want any comments. I arrived with my child and my wife's child, I know she is the aunt, but I brought toothpaste, detergent to wash the laundry, I bought the food; for me, it was very important to not owe her anything, because I know it would come back to me. I'm prudent." Prudence in this case is to preserve the possibility of exit. Godbout, *op. cit*

voice), or to do nothing and follow on (*loyalism*)[46]. In the case of information technology, the situation of monopoly is important because of Microsoft. In this case, the general reaction is loyalism.

By its non-utilitarian nature, the gift encourages voice reaction. Effectively, the relation is important in itself, it cannot be broken as simply as in the market. Furthermore, the separation of subject and object is not present in the same way as in the market.

Even if free software seems to share a lot of the gift dynamic, one important difference subsists. Whereas with the gift the capacity for exit is quite small, it is affirmed in free software. First of all, one should note that their is no obligation to cooperate, one is free[47]. Liberty with respect to participation also signifies that a project can be abandoned for want of participants. It is a risky bet, but which seems to have rewards. One should add that if one participates in a project, one does not necessarily engage oneself beyond a specific participation. It is frequent to read a note on the web page of a project, specifying that the support (in particular help, bug correction, etc.) is furnished, but not guaranteed, even though in many cases such a note signifies that the help is almost certain. This absence of guarantee is already present on the legal level. Effectively, many free software licenses specify that the software is furnished “as is” and that the developers take no responsibility for it. This means that the person using the software can demand no support. This absence of responsibility or guarantee shows that the developers assure themselves a possibility of exit. The participants thus signify that even if they enter into a gift exchange dynamic, they can also quit it. The possibility of exit is very important, as it decreases the risks of engaging in a relation which would be too much for one, but from which one could not withdraw. It also reduces the risk of “prestige gifts” which we referred to above. Which doesn’t mean that prestige gifts don’t exist in the free software community. But the possibility of exit reduces the importance that projects based on such conceptions of the gift can take on[48]. One should add that even if this possibility for exit exists and is present, many hackers engage themselves in the projects.

However, despite this possibility of exit, free software is based fundamentally on the reaction of voice. Exercising voice is encouraged in two ways. On the one hand, by offering possibilities to act within the relation (by distributing the source code, by calling for propositions, etc.), voice is valued and encouraged. On the other hand, the relationship created between donor and recipient seems to favor voice. Thus, the relation that emerges is quite different from that between producer and consumer. In the case of free software, the notion of consumer loses its meaning. As Linus Torvald says, “the users act as another form of producers: they do not produce code, but information relating to the product and a precious evaluation of how it can be improved”. The FSF also rejects the term consumer: “describing the users of software as "consumers" presumes a narrow role for them. It treats them like cattle that passively graze on what others make available to them.”[49] Thus this distinction tends to reduce itself, bringing consumers and producers closer together.

Free software by nature favor non-merchant relations. That does not mean that the market keeps away from it, after all, they are products with no cost and therefor interesting for a business. Some businesses or even holdings, employing full-time programmers[50], participate in, or direct free software projects. The

[46] This corresponds to the types of reactions established by Albert Hirshmann.

[47] This difference can perhaps also be discerned in the way the new activist movement is organized more generally. Groups seem to be less permanent, individuals engage themselves for limited times or projects, and may well pull up stakes and move to another continent (or another activity) when its over. Older activists often have difficulty with this. A new sort of individualistic solidarity? (translator’s comment)

[48] There are projects that accept very few suggestions, or participants who try to impose their views for reasons of prestige. Once discovered, such behaviour is generally quite disapproved of.

[49] Quoted from FSF : http://www.gnu.org/philosophy/words-to-avoid.html

[50] For example, Sun, IBM, Novell.

businesses take up a utilitarian standpoint withing the community. They participate because it's a more efficient tool. It is the point of view defended by the partisans of *open source*[51]*,* a utilitarian point of view within the community. They participate because it is a more efficient means or tool. However, most participants situate themselves somewhere between the positions of the FSF and those of *open source*. However the influence that such a point of view will have on the free software, as gift exchange communities by nature tend to keep their distances with respect to utilitarianism.

5. A peculiar of life form

This work ethic and the movement created by gift exchange engenders a new life form, a new form of social organization for several reasons:

> Even if a hierarchical authority exists, one has no obligation with respect to it. The absence of salaried relation and the affirmed liberty to cooperate or not allow this relation. Work is not prescribed, it is attributed on a voluntary basis. Thus each person does what he can and likes to do.
>
> Direct cooperation exists between the different participants of a project (or specific part of a project).

The structure is very horizontal, even if one generally observes that there are administrators, sorts of leaders, who decide which modifications to accept, which to refuse and the general organization of the project. One must note that this can be the subject of long discussions within the community. Thus there can exist patches[52] or propositions for changes distributed on the side, with official or un-official status. This reinforces the horizontality of the structure by permitting the expression of diverging directions. However, un-official patches can also be looked down on by a part of the community and generate internal conflicts which can even lead to a division of the community.

There exist several forms of leaders, historical leaders or project founders, with often undisputed authority, as well as leaders accepted for their competencies, whether social or computational. Thus the status of authority is open to all. It is a kind of benevolent dictatorship, based on the history of the project and/or competence, a sort of meritocracy. In some projects, there are elections to choose the directing committee.

There is an infrastructure which facilitates cooperation among developers. There are the usual forms of Internet communication, such as mailing lists, forums, chat, etc. But above all, there are tools permitting direct cooperation on the source code itself. These tools in particular allow one to see the latest modifications made and their authors. It is also possible to see the differences between two versions of the same file[53].

[51] It is interesting to note that the term “free software” is founded on an ethics of liberty. In contrast, the term “open source” is based on the means and its evaluation. The concept of open source was brought by Eric S. Raymond in his famous article, *The bazaar and the cathedral*, in which he analysed the reasons for the success of free software (particularly with respect to performance). Raymond was also among the first to bring businesses into the free software community through the Mozilla project, very linked to Netscape, which can produce a commercial version. It is also interesting to note that Eric S. Raymond brings the concept of function of utility of the participants into his analysis, thus showing his utilitarian conception and minimising the ethics defended by the FSF. This attitude is not total however, in the measure that he also defends the hacker attitude.

[52] Modifications of the source code that can or cannot be applied. A patch can offer additional functions, correct a problem, etc.

[53] The software most commonly used for this is CVS https://www.cvshome.org/

It should be remarked that even if part of the projects are cooperative, there are many that are maintained by a single individual.

6. Conclusion

One can ask oneself if the new life form characteristic of free software communities is not a new form of social organization destined to take an important place in western society. Effectively, this society is more and more centered on the virtual economy, an economy which requires a creativity that the closed model seems unable to provide as well as the open, the model of cooperation used in the free software movement. As Pekka Himanen notes, the hackers are inserted in a contradiction of capitalism. Capitalism involves producing always more goods derived from an ever more sophisticated technology. The model of non-appropriation of immaterial goods seems to better correspond to emergence of the kind of creativity required.

However, capitalism also depends on the constant appropriation of new technologies. One can suppose that the recent adoption of patents on software by the European Commission is part of an attempt to appropriate knowledge, faced with the open model that has taken up positions within this contradiction. For what is striking about the patents on software is not the appropriation of the application of an idea, but the appropriation of the idea itself[54] Similarly, there are attempts to bring Internet to heel, for instance the adoption of the LEN[55] in France. But we leave to another article in this issue the task of describing these phenomena of "enclosures".

These attempts must not mask the fact that free software is a real danger for the capitalist software enterprises. In fact, the recent "studies" published by Microsoft on the differences between free software and the Microsoft versions show that the monopoly of Redmond is afraid. The decisions of the county of Munich and of the Peruvian government to adopt free software, rather than their proprietary equivalents shows that free software is becoming a serious competitor. Microsoft is particularly afraid because free software isn't playing the same game as they are. There is no way of buying out free software, or trying to gain control of the movement via financial means. The only possible war is ideological or legal[56].

It is interesting to see how the model of free software can be extended to other similar domains having the same properties of immaterial goods. There are already free music, free films and free texts[57]. The lesson to be learned is above all political: *let's be political hackers*, we would like to suggest. The practice of free software brings elements that allow one to conceive of a real practice of cooperation concerning political theories. A practice which is still little developed among anti-globalisers. Effectively, even if texts are exchanged, it is difficult to imagine redistributing a text after having changed a four or five phrases[58]. We still conceive of political thought as individual opinion, and not as the fruit of cooperation, the fruit of a history also made by others. We have difficulty accepting that others

[54] It is now possible to put a copyright on a program which shows a menu when one clicks the right hand side of a mouse. It is possible to copyright the *idea* of a menu summoned by a click. This means that every program using this idea would have to pay rights to the holder of this copyright.

[55] Loi de confiance dans l'Économie Numérique (Law of confidence in the digital economy), whose goal is to increase control over the content of Internet.

[56] Microsoft is using both modes of action: publishing "studies" on free software and also acting on the level of patents.

[57] For example WikiPedia, a free encyclopaedia whose content can be modified by everyone, thus participating in its elaboration : http://www.wikipedia.org

[58] Actually, I made a few minor improvements, but he will never notice! (translator's note)

appropriate what we have written, generally considering that to be plagiarism. And yet free software is in no way plagiarism, for the history of the text or software is recognized, the different authors cited, etc. To say that its history is recognized means that one keeps a record of its development which is accessible[59]. That would be another interesting practice for politics, as to see the development of the thought that led to such and such a political position being taken could be just as interesting as the final text itself.

You can obtain the latest version and propose your modifications, discuss this text, etc., at : http://hacker.nabix.net.

[59] One keeps a trace of the development of a software because at a certain moment it may be necessary to go back and follow an alternative path.

[60]You can obtain full license text , here : http://creativecommons.org/licenses/by-nc-sa/2.0/