Tere Vadén
tere.vaden@uta.fi

Professor
Hypermedialab
University of Tampere, Finland

Intellectual Property, Open Source and Free Software

The concept of "intellectual property" has gained a lot of attention recently. Several conflicting trends make it a focal point of controversy. The global trend towards an "information society" gives an increasing role to information, knowledge and other immaterial assets in production. Therefore the economy also needs ways of controlling, identifying and using immaterial assets. This happens largely through the concept of intellectual property.

In economical terms, the notion of intellectual property and the connected immaterial property rights are a way of regulating free markets, setting up limited monopolies in the name of economic incentive for innovation and creativity [1].

A conflict readily arises, because our cultural traditions include deep-seated convictions and practices that emphasize the fact that intellectual "things" are immaterial also in the sense that they can not or should not be made into property or into commodities. Just to mention one example, the ideal of modern science has been that knowledge or at least scientific knowledge is by definition something that can not be owned, something that by definition is open to all. It has been argued that the special status and role of scientific knowledge – its so called objectivity and potential for liberation – are dependent on this non-proprietary nature of scientific knowledge, as contrasted in the classical enlightenment discussions to religious and secular authority.

This conflict between the "realities" of economy and "idealism" in scientific and other everyday practices, including a belief in unrestricted free markets, can be seen in the case of software, too. It has been often noted that the hacker culture of the 50's, 60's and 70's understood software largely through the model of scientific knowledge, as algorithms, theories and so on [2]. There were also other features in the culture that supported a practice in which software code was shared and in that sense free from ownership. In the 80's with the advent of the PC and new mass market economic possibilities, a new way of viewing software arose: software became to be seen as a creative work that is protected through copyright. This made it possible to own

1

software and to make money through selling it in large quantities, or, more precisely, selling the right to use software. A regime of software licensing arose.

The trend toward the proprietarisation of software has been counteracted by the free software and open source software (F/OSS) communities that develop software the use of which is not limited by "owners". The idea of free software, formulated in the GNU project by Richard M. Stallman [3], is to give the user the freedom to study, use, modify and redistribute the software without having to ask for permission from anybody else. A prerequisite for these freedoms is the access to the source code of the program ("open source"). The free software movement has since the 80's been creating free software for various purposes. The most famous example of free software may be the free operating system GNU/Linux, a set of free software with the Linux kernel as its core.

The development and use of F/OSS is of course dependent on the licensing practices of the communities and the distinctions between free software and open source are partially reflected in the preferred licenses. For free software enthusiasts, the license of choice is the GNU General Public license (GPL) that in essence demands that one is free to use the software and to modify and redistribute it, provided that the modified and redistributed versions are also licensed under the GPL. The GPL is the major legal and social innovation that has made F/OSS development possible given the current copyright laws and their interpretation.

The strict and principled adherence to software freedom exemplified by the Free Software Foundation and by Stallman himself may have been one contributing factor in the birth of the open source movement. The unyielding commitment to freedom and the strictness of the GPL were seen too limiting, and the open source movement essentially wanted to formulate itself as a business friendly alternative, with a business model and licensing practices of its own. Open source as a movement puts emphasis on the openness of software code as a new method of software development that results in better and more reliable code, possibly with a lower total cost of ownership for the user [4]. Crudely put, the Free Software movement is more a social and political movement, while the Open Source movement is more practically oriented towards the improvement of software, and through that, the empowerment of the users. The Free Software movement is interested in creating a particular kind of community. The Open Source movement is interested in creating software in a particular way. This difference in the underlying commitments and philosophical underpinnings also results in different attitudes towards the concept of intellectual property.

The reductio ad absurdum of intellectual property

In the same way as one may ask "do ideas exist and in what way?", one may ask "does code exist and in what way?". Leaving the long-winded philosophical discussions and the classical answers aside, we need to notice that the existence of

2

immaterial things is quite different from the existence of material things. Are immaterial things independent from the observer? How independent? Would money exist if there were no humans? What about the state of France or the theory of relativity? What about code? It is important to keep in mind that both the existence of code (information, the digital) and its ownership are socially and historically contingent phenomena, and that different social and philosophical convictions lead to different attitudes toward code as "intellectual property".

As noted, scientific information or knowledge is an example of "code" as something that can not be owned, can not be property. In the other extreme we have proprietary, copyrighted or even patented software, the use of which is limited by the owners of the software. It is good to keep in mind that there is nothing in the information itself that can tell whether it is owned or not. A digital codification of a scientific theory or formula and a patented software algorithm are indistinguishable unless they are interpreted. This means also that in order to assign owners to code one has to have means of identifying bits of code Ð and the users of the code.

Taken to its extreme, the notion that information or knowledge is owned becomes absurd. A child being born either has to be taught that information is owned or otherwise remains ignorant of the fact. In both cases information freely shared is the basis on which the ownership of information can be based. The reductio ad absurdum in its crudest form is: if all information is proprietary, then the information that information is proprietary is proprietary, too, and I can choose to stay ignorant of that information. As with material property, intellectual property relies on the goodwill of non-proprietary social functions and arrangements. Therefore its beneficiality is not a given. A society has to strike a balance between proprietary and non-proprietary knowledge and information.

All of the above considerations seem to point out that the way in which we think that "code" exists and the way in which intellectual property and software are in the world has wide social consequences. As a crude example: when the system of intellectual property rights kicks in is not socially neutral. Consider the case of Disney that has been widely used as an example in the discussions [5]. In creating its products, Disney has extensively used the creations of previous generations, especially children's stories by the brothers Grimm. Now, after the copyright term has been repeatedly extended in the US, and after this extension by the congress has been judged to be constitutional, it seems that Disney can keep its content closed for an indefinite period. This creates an unjustified fragmentation in the creative chain of generations. When the copyright system kicks in, how it is interpreted, and how long the term is, are all factors that have an influence on the economical and intellectual balance of a society. In this way, the development of digital rights management and other IC-technologies are not socially or politically neutral.

Intellectual property and different types of knowledge creation

In order to highlight the different views taken on intellectual property, it might help to consider different types of knowledge work, or knowledge creation. Following sociologist Scott Lash, we may separate between two different kinds of social

3

structure: organisations on the one hand, and disorganisations on the other [6]. According to Lash, organisations are hierarchical wholes bound by norms. Examples include companies, research institutes, churches and so on. In these kind of social structures the participants pretty much know their role and place. They communicate via chains of command that are legitimated through normative rules.

Control and production go hand in hand with clearly defined goals and assigned tasks. This is the type of organisation with formal boundaries and Taylorist division of labour that we are familiar with. However, Lash proposes that the information society is increasingly based not on organisations but on disorganisations. Disorganisations are networks or "rhizomes" bound by shared values. In that sense they are non-institutional, non-formal, but as Lash emphasises, also quite non-chaotic. A disorganisation can be much more effective and creative than an "organisation" even though its rationale is not bound to a hierarchical set of norms. The contrast is portrayed in the contrast between a church and a sect. A church is a hierarchical whole bound by normative rules. A sect typically is a non-institutional group of people bound together by the values they share. Shared values give the disorganisation its power to act and its organisational principles.

Lash claims that the vital processes of the information society are increasingly dependent on informal networks of disorganisation that work inside and across organisations. The disorganisations, value-bound networks, use organisations in order to reach their goals, but they do not themselves get organised into institutional and in that sense identifiable wholes with stable structures. Disorganisations disband and form new channels of co-operation on the fly. Therefore the power wielded in disorganisations is not the legitimate power of organisations, but rather (cultural, not typically physical) violence based on the ability to interpret and open the underlying values; charismatic authority often plays a crucial role.

It is easy to see that this characterisation of the difference between organisations and disorganisations quite nicely captures the difference between software development in a commercial company and software development in the F/OSS models. While software development in an organisation is based on hierarchical division of labour and Taylorist forms of work, disorganisations "scratch their own itch" and cohere and decohere insofar as the underlying values that loosely define the community are shared. The Linux community, for instance, is a disorganisation that has no institutional organisation and in which the authority of the informal disorganisation is based on how well the disorganisations is able to embody and advance the values that are central to the community (such as the progress of the kernel, openness and effectiveness of development, freedom of the software, openness of the source, technological advantage, fun, etc.).

Through this characterisation it is easy to see that knowledge creation in organisations and disorganisations functions under very different sets of conditions, and, consequently, benefits or disbenefits differently from the notion of intellectual property and its application. Briefly put, organisational knowledge creation can benefit from the notion of intellectual property. When information, knowledge and other immaterial assets can be identified and when owners can be assigned to them, the knowledge creation in an organisation can be rationalised, incentives can be

4

formulated, one can measure the accumulation of property and so on. If we know which department in an organisation is responsible for this or that process in knowledge creation or has ownership over this or that asset, then the division of labour and prediction of outcomes becomes much easier. In this sense, the notion of intellectual property is also an organisational tool that helps to guide and structure the processes of co-operation.

On the other hand, for a disorganisation the notion of intellectual property is potentially disruptive if not downright destructive. Disorganisations are based on shares values, therefore artificial barriers to sharing are a hindrance and the phenomenon of free-riding almost always creates friction and possibly fragmentation in a disorganisation. Being a part of a disorganisational network involves a willingness to operate across the boundaries of recognised and legitimated organisations. The co-operation in a disorganisation happens in good faith, and is disrupted by ways of acting that do not harmonise with the values that underlie the disorganisation.

There might be disorganisations in which the disruptions produced by intellectual property rights might not be great, such as disorganisations that to a large extent are based on charismatic authority. However, knowledge creation in the free software and open source communities is greatly harmed by the notion of intellectual property, because one basic value on which those communities are based on is precisely the freedom or non-proprietary nature of software code and related intellectual skills. Participation in free and open source development is voluntary, communities often centre around a charismatic leader, reputation, respect and "karma" are sought. This kind of community would be impossible if the object of development, code, would be owned by somebody. If one wonders what (cultural) violence means in the case of these kinds of disorganisation, one may ask oneself what prevents the Linux kernel from forking. Nothing, in fact, other than the disrespect that a forking community would receive.

The fact that intellectual property as a concept is beneficial to organisational knowledge creation and potentially destructive to certain types of disorganisational knowledge creation is the source for a great part of the current debate and problems around intellectual property rights, such as copyright and patents. The interests of independent communities and the interests of organisational IP owners are clearly different. The problem is that as a whole, the information society is dependent on both kinds of knowledge creation, and should be able to strike a balance that benefits both types of knowledge intensive work. Furthermore, several sociologists, including Manuel Castells and Scott Lash, have suggested that the importance of disorganisational activity is continually increasing. One more ingredient making the discussion between the different interests harder is the fact that organisational knowledge creation done in companies and institutions is much more visible than the disorganisational creation that is often channeled through organisations. This creates a systematic bias in favour of organisational interests, because those interests are readily quantifiable.

Fun or freedom: hacker views on intellectual property

5

Because of the different philosophical underpinnings, different "wings" of the F/OSS community hold different views towards intellectual property. As has been noted, the free software movement is more a social movement than a software development model. The principles of free software apply to other similar intellectual creations, as well. The president of the Free Software Foundation, Richard M. Stallman has repeatedly emphasised that we should prefer no software to proprietary software. In this way, the Free Software movement sets freedom higher than utility, and the community higher than software development. Therefore the ideals behind free software also mean that something like intellectual property is always destructive. If the ideal is the freedom of information, then the ownership of information can only be an obstacle or a dangerous misunderstanding of the nature of information and an ethically balanced society.

In contrast to this, the Open Source movement is careful to point out that it takes a pragmatic view on things. Whatever promises better development of software and technological advantage is good, and the claim is that open source development promises precisely those things. Therefore the issue of freedom is not crucial, and software may be owned in so far as that does not fundamentally threaten the further development of open source software. In this view, intellectual property rights might be good, if they are practical and advance the development of good quality software. With regard to intellectual property, the Open Source community moves with caution, but in a noncommitted way. The position is best described as agnostic.

Linus Torvalds has expressed his convictions on the importance of openness and freedom in several interviews and in the famous "Linus Law" found for instance in the foreword to Pekka Himanen's book the Hacker Ethic [7]. Torvalds emphasises that having fun is the ultimate motivation for him and the hacker community. If we conceive of the hacker community as a community of fun as opposed to a Stallmanian community of freedom, then we have another attitude towards intellectual property rights. Intellectual property rights might be fun. There's no reason to believe that intellectual property could not provide the basis for a community that enjoys programming and that co-operates – up to a point. For instance, Torvalds has consistently expressed his belief in the pragmatism of fun: people will do what they enjoy, and in so far as freedom and fun travel together, that is nice, but if intellectual property is even more fun, so be it.

This range of positions seems to suggest that even though hackerdom as a social group can in some sense be identified and studied, the spirit of a "hacker ethic" is too loose to allow for a definite characterisation. This is no surprise, of course, as in a similar vein something like "Christian ethics" or "medical ethics" is equally impossible to pin down. There is one further complication, however. It is not clear that the hacker ethic as portrayed, for instance, by Himanen or Torvalds is an ethic in the first place. The same goes for the tenor of the Open Source movement. If the commitments and principles characterised are formal placeholders like "work enthusiastically", "have fun", "do not pay attention to social conventions", "do the best product you possibly can", etc., the question of what the work is about or what it will result in is completely left out. By describing a method of working – a hacker ethic, the open source model – one does not in fact give any sort of ethical guidelines.

6

The model and the method might coincide with principled goals, but they do not say anything about the goals themselves; they are not contentful.

In order to get to the ethical content one has to ask some "why" questions, merely answering "hows" is not enough. It seems that the portrayal of the open source movement in its agnostic and Torvaldsian forms does not really ask the question: what is information technology for? Why are we doing this? The answer "for fun" doesn't cut much ethical ground. Hitler and Göring also enjoyed themselves and their work immensely. As a development model, F/OSS is, at best, ethically neutral. When it comes to social and political questions like intellectual property, development models and work ethics are not enough. One has to actually say something about the society that one wants to create and the values that one believes to be ethically justified. The hacker ethic as a way of working and as a way toward some specific types of community are two quite different things.

Answers to the "why" question diverge, of course. For instance, economic and cultural "whys" may give different weights to different factors. Globalisation as a trend and the creation of a particular type of information society push towards a strict intellectual property regime. This, however, does not mean that intellectual property as a concept or as a practice systematically favours democracy or development. Intellectual property rights might, for instance, protect the livelihood and cultures of indigenous populations and local cultural endeavours, but in practice they next to never do. This is because established organisations have an upper hand when it comes to interpreting the concept and the laws that codify it.

Intellectual property rights were seen as an obstacle to economic growth in the US of the 19th century, and it is reasonable to believe that they are an equally bad obstacle to the developing countries of today. This is why the use of F/OSS software is ethically motivated through concepts like freedom, independence, swantantra; economic growth narrowly defined might in fact speak in favour of proprietary software and proprietary ideas. Ethics and fun may, indeed, coincide; this is one of the beautiful things about hacking. However, the point of social arrangements like democracy is to take extra efforts in order to guarantee equality and justice. The question is whether we view F/OSS development as something done for fun or as an extra effort to be taken on ethical, social or technological reasons. Here also the perspectives of the developers and the users might diverge. For developers, fun might be motivation enough, while for non-savvy users the adoption of F/OSS might mean taking an extra effort in need of some sort of ethical or social justifications.

Notes and References

[1] See, for instance, Michael Perelman, "The Political Economy of Intellectual Property", *Monthly Review*, January 2003; http://www.monthlyreview.org/0103perelman.htm

 [2] Steven Levy, *Hackers*, Penguin, London 2001; Pekka Himanen, *The Hacker Ethic*, Random House, New York 2000.

[3] See Richard M. Stallman, "The Gnu Manifesto",
http://www.gnu.org/gnu/manifesto.html, and "The Initial Announcement of the GNU
Project", http://www.gnu.org/gnu/initial-announcement.html, from year 1983.

 [4] For open source, see e.g., http://www.opensource.org/ and Chris DiBona et al.
(eds.), *Open Sources*, O'Reilly, Sebastobol 1999.

[5] See, for instance, Lawrence Lessig, "Free culture" presentation at the OSCON
2002 conference; http://randomfoo.net/oscon/2002/lessig/, and http://www.oreilly-
net.com/pub/a/policy/2002/08/15/lessig.html.

 [6] Scott Lash, *Critique of Information.* SAGE, London 2002.

[7] Linus Torvalds, "Linus' law", foreword in Pekka Himanen, op. cit.