

# Operation “Open Source” - A Challenge for Governments and Citizens of the Information Society.<sup>1</sup>

by Andrzej Kocikowski

## 1. Abstract

One of the great challenges facing both citizens and governments of the Information Society is the universal use of Free and Open Source Software (FS&OSS). The main purpose of this paper is to substantiate the argument that FS and OSS constitute the only sensible alternative to absurdly commercialized and dangerously monopolized market of proprietary software. The second focus of the paper is to point out to the opportunity that presents itself to the Europeans who are taking up upon themselves the challenge of closing the technological gap between them and the Americans. The author is convinced that promoting FS&OSS should constitute a long-term and complex undertaking that, after being accepted by national governments (not only European, but first and foremost European), will become one of Brussels' leading economic programs; the project of such a program (see the Appendix) with its justification is an integral part of the paper.

**Key words:** Free Software, Open Source Software, Information Society, Quality & Security of Software.

## 2. Selected definitions

In the title and abstract of the paper we find terms that seem perfectly obvious, such as Free Software (FS), Open Source Software (OSS) and Information Society. However, in order to avoid misunderstandings connected with the interpretation of the first two terms, and to remind the reader about important distinctions between Information Society and Informational Society, which might indirectly affect the attitudes of some Europeans (but not only theirs), I'll take the liberty of making some comments on both issues.

### 2.1. Free Software *Versus* (?) Open Source Software

However we look at it, the free software movement is 20 years old by now. I have no idea if any champagne bottles will be opened for this occasion, but I assume that it is a very good moment for some new important decisions to be made in this matter that is so very important for the citizens of information society.

I will say straight away that - in this paper - I don't intend to enter the dispute between users of Free Software (FS) and users of OSS. Assuming that not all of the readers need to know the particulars of this matter, I will cite a few fundamental statements that specify the position of each party clearly enough.

In 1998, some of the people in the free software community began using the term “open source software” instead of “free software” to describe what they do. The term “open source” quickly became associated with a different approach, a different philosophy, different values, and even a different criterion for which licenses are acceptable. The Free Software movement and the Open Source movement are today separate movements with

---

<sup>1</sup> Converted (\*.pdf) from (\*.html): [source: [http://mumelab01.amu.edu.pl/ethicomp2004/ethicomp2004\\_EN.html](http://mumelab01.amu.edu.pl/ethicomp2004/ethicomp2004_EN.html)]. Original paper was published in: T. W. Bynum, N. Pouloudi, S. Rogerson, Th. Spyrou [Eds.], (2004), Challenges for the Citizen of the Information Society, Proceedings of the Seventh International Conference ETHICOMP 2004, Vol. 2, pages 532 – 540, University of Aegean.

different views and goals, although we can and do work together on some practical projects.

The fundamental difference between the two movements is in their values, their ways of looking at the world. For the Open Source movement, the issue of whether software should be open source is a practical question, not an ethical one. As one person put it, "Open source is a development methodology; free software is a social movement." For the Open Source movement, non-free software is a suboptimal solution. For the Free Software movement, non-free software is a social problem and free software is the solution.

The basic idea behind open source is very simple: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing.

We in the open source community have learned that this rapid evolutionary process produces better software than the traditional closed model, in which only a very few programmers can see the source and everybody else must blindly use an opaque block of bits.

[ ... ]

Open source software is an idea whose time has finally come. For twenty years it has been building momentum in the technical cultures that built the Internet and the World Wide Web. Now it's breaking out into the commercial world, and that's changing all the rules. Are you ready? [<http://www.opensource.org/>]

Relationship between the Free Software movement and Open Source movement [ ... ]

Radical groups in the 1960s developed a reputation for factionalism: organizations split because of disagreements on details of strategy, and then treated each other as enemies. Or at least, such is the image people have of them, whether or not it was true.

The relationship between the Free Software movement and the Open Source movement is just the opposite of that picture. We disagree on the basic principles, but agree more or less on the practical recommendations. So we can and do work together on many specific projects. We don't think of the Open Source movement as an enemy. The enemy is proprietary software.

Taking the above statements into account, but keeping in mind that the most important is what is common, not the differences, I would present the basic definitions (for the benefit of those who never encountered the term FS or OSS) as follows:

- the term FS&OSS denotes a philosophy expressed by numerous computer programmers all over the world, who believe that software should be distributed as open and easily modifiable code that is accessible **free of charge** to anyone. In its most extreme form, the authors of such software **should not receive payment**, because, after all, each one of us can selflessly give something to others;

- the term FS^OSS denotes a view that software should be distributed as open and easily modifiable code that is accessible **free of charge** to anyone.

The first case is the original, most noble conception of FS&OSS. I have the highest respect for the people who created this conception and those who accept it, who - thank goodness - are many. Indeed, this paper is dedicated to them in thanks for all they have done for us, the citizens of the Information Society. Of course, I am also fully aware that this idea - to which they have given a big portion of their lives - will not be widely accepted.

Another meaning of the term “open source” is a slightly less noble idea: computer programmers receiving fair pay for their work instead of giving it away for free.

To rule out any doubts I need to state that, for the purposes of my paper, the term FS&OSS is used in the latter meaning. In this way, I am expressing my approval for the view that - let's repeat - the FS and OSS movements are acting for the good cause. This is adequately (and could be even better) reflected in the Free/Open Source Research Community project. It says:

Free/open source software (F/OSS) is software for which the human-readable source code is made available to the user of the software, who can then modify the code in order to fit the software to the user's needs. The source code is the set of written instructions that define a program in its original form, and when it's made fully accessible programmers can read it, modify it, and redistribute it, thereby improving and adapting the software. In this manner the software evolves at a rate unmatched by traditional proprietary software.

For many years free/open source software has been building momentum. Beginning amidst the technical cultures that produced the Internet and World Wide Web, it is now causing quite a stir in the commercial world as large software corporations are finding themselves competing against commercially available open source software.  
[[http://opensource.mit.edu/what\\_is\\_os.html](http://opensource.mit.edu/what_is_os.html)]

## 2.2. Information Society *Versus* (?) Informational Society.

For obvious reasons, we cannot analyze fully within the bounds of this paper the question stated in the title of this paragraph. I can only promise that I will do so at the earliest occasion. For the moment, I will merely say that the distinction introduced by Manuel Castells [Castells, 1998] can be interpreted in many different ways - regardless of author's intentions - and that one of them leads to some interesting consequences.

The term “information society” is not defined clearly in many Polish publications, including the official governmental documents. The description used most commonly is “society based on knowledge and information”; seldom is there any attempt to explain what that should actually mean. Only some authors refer to, for example Fritz Machlup [Machlup, 1962] and try to connect the aforementioned unclear “based on knowledge and information” with Machlupian term “production and distribution of knowledge”.

A Polish expert, Professor Tomasz Goban-Klas explains the term “Informational Society” [Goban-Klas, 1999] as *społeczeństwo z informatyzowane*, which in one of the possible translations into English means a society in which ICT is used on a large scale. However, it is well known that using ICT is not a sufficient condition for production and distribution of knowledge. Even if it were, this knowledge does not have to possess qualities that would enable it to increase gross national product, contrary to Machlup's intent clearly visible when he writes about “information industries” (giving an example of USA), pointing out that in 1958 the so-called information sector created almost one third of GNP and employed the same part of the work force.

If we are to accept the perspective laid out by Fritz Machlup's way of thinking, we have to state that Information Society is a society in which a clearly specified **significant part of gross national product** - in the model/ideal case it is 100% - is obtained by production and distribution of knowledge; mostly scientific knowledge in the true meaning of the word.

Following this line of reasoning, it should be said that any other society, regardless of the scale of ICT utilization in economy, banking, media, administration and education, can be at best more or less *zinformatyizowane*, in other words - to put the matter in a neutral light - an Informational Society, not an Information Society. In an Informational Society ICT facilitates efficient functioning of many traditional parts of economy, allowing for sufficiently efficient connection with information societies; which is in interest of both the former and the latter.

Let us analyze a simple example. Let assume that automobile corporation "X" built a factory of its model "A" in Poland, or another similar country. That means that in the factory many industrial robots need to be set up and activated because such is the method of producing automobiles in corporation "X". The fact that the most modern and refined ICT is used in that factory (the robots can be remotely programmed from corporate HQ, a thousand kilometers away) does not imply that the society in the country in which the factory was built is an Information Society. That would be possible only if the knowledge necessary for design, manufacturing and programming of robots set up in the factory was created in this country, and then sold - increasing the gross national product.

### 3. Why FS&OSS?

A key question here is: Why would citizens of the Information Society, their governments, and the representatives of both - the European Parliament - support the promotion of increasingly universal use of open source software? As in any other case, here too the reasons are many and of various importance. I will try to discuss the most important of them below.

#### 3.1. Cost of software.

There are several factors that, in the current situation, determine the cost of software to society. One is the direct consequence of a phenomenon known for almost three hundred years by the name of market monopoly. One of the standard monopolistic practices, described in detail already in the 18th century, is the practice of using a dominant position to extract an extraordinary profit from the market. There is no reason to think that the software market is different in this case. Another standard monopolistic practice is the conviction that adherence to basic economic practice of cost rationalization applies to everybody else, except for the monopoly. Thirdly, I would also call monopolistic the standard practice - resulting exclusively from pure egoism and from the pursuit of extraordinary profit - of arbitrarily forcing users of the product (in this case a software) to replace old versions with a new one, even though older versions have not been used "up" yet.

It turns out almost grotesque: imagine a situation when in result of expensive advertising efforts somebody was coaxed into buying a cup of yogurt and after that the cup has been torn from his/her hand without regard to the fact that he/she has only eaten two spoonfuls; and he/she is forced to buy a new cup of yogurt (at a suitable price, of course) because the new one is even better than the last one; and so on, and on, and on.

The second factor determining the cost of software to society is, paradoxically, the direct consequence of dispersion of the process of creation of the product. Presently, in many private

companies working on the same, or very similar software, a multitude of different people is trying to solve the same, or very similar problem. It can be argued that duplicated work leads to waste of time and, as a consequence, to an increase of production costs (which translates as highest cost to society. I am convinced that all those people would solve their problems much faster if they worked in smart cooperation - and therefore with lower cost.

Those two factors should suffice as an initial proof for the thesis that the current situation on today's software market - on the one hand huge, private monopoly and on the other hand the dispersion of efforts of many small (smaller than the monopoly in question) companies - is conducive to increasing costs of software.

The alternative way provided by FS&OSS makes it therefore possible to **decrease drastically the cost of software to society**; mainly because of the disappearance of the unfairness and nonsenses of monopolistic practices. The problem of duplication of effort invested in solving the same problems, so characteristic of market with small, dispersed companies also disappears. If we add the possible elimination of costly advertising and expensive and time-consuming patent issues, it turns out that the real cost of software to society is surprisingly low.

### 3.2. Quality and security

Let us address the next important issue - quality and security of software. It is well known that both the former and the latter depend strongly and directly on talent and qualifications of the programmers; they depend also on, sometimes very complex, quality control procedures. Let us focus on the latter for a moment. I believe, that it is much easier to discover problems resulting from a simple mistake, or a wrong use of particular tools, or negligence, or (as Gene Spafford calls them) "programmed threats" when independent experts take part in the development process; and when when the process itself follows the "open doors" policy. My conviction is justified by the existing similarity between computer programs and scientific theorems. Like a "regular" scientist, a computer programmer too assumes basically that if certain conditions will exist then certain effects will take place. Indeed, every line of code is a step in the "proof" of a particular theorem. Basic knowledge in the methodology of science states that the best way to judge the value of a theorem is through the process of intersubjective verification. That means that many individuals - a great many, if necessary - comprehensively examine the "proof" for correctness. I believe that this kind of (essentially social) control of the correctness of software code cannot be replaced even by the best organized control within a **profit-oriented corporation** that creates programs "behind closed doors." A Case in Point: The Infamous Story of the Enron Corporation. Enron, with its alleged "model" control system, elaborate employees' training programs and assessment, was a "poster boy" in business ethics textbooks - and in secret illegal money making through "creative bookkeeping"! Is there any better proof that lack of an "open door" policy and social control enables a significant discrepancy between declarations and reality?

Let us ask the following question: is there any social, rational reason for which the monopolist on the software market **MUST** introduce a new operating system every 2 years and - for example - new versions of office suite?

My answer is: No! There is no reason. The reason for the above can be only a boundless greed for the money in our wallets; a money that will strengthen the monopoly's position so it can continue to enslave new generations of consumers. Ask me why I'm asking this question. Well, I am asking because I find it interesting whether we could live with an operating system that is changed "only" every 4 - 5 years. If yes, than let us imagine a situation when there are five

times as many programmers working in a rational cooperation on the efficiency and security of an operating system than there are in a corporation with the monopoly power. They do not have to hurry, since they are exempt from the Development Schedule of the Monopolsoft Corporation (a made-up name), which enforces new release of an operating system every two years; therefore, they work calmly and thoughtfully for, let us say - four years. After four years of such cooperative work - with forethought, and "behind open doors" - wouldn't we have a product the cost of which to society is incomparably lower, and the quality, security and efficiency - much better? How many times will we have to rediscover the truth that we can achieve more in the same time by working together; that the product can be more elegant, better adapted to user's needs, more secure and efficient?

### 3.3. The only acceptable monopoly – democracy

I do not care for monopolies, because they contain various totalities, and these are rarely good for people. I live in a country where, which not all might remember, not too long ago we had everyday contact with political totality, and where today still various totalities in economy (monopoly in telecommunication, railroad transportation, fuel and energy sectors) make it often impossible to undertake sensible, rational projects. For that reason, I may be paying more attention, than others, to the dangers coming from various monopolies. As I said, too many old and not so old experiences developed a habit in me, which is not easy to get rid of. And there are still many fears.

Those who read my paper from the last ETHICOMP conference [Kocikowski, 2002] might still remember its main thesis, which contained a picture of consequences of a monopoly's actions on the software market, especially the potential dangers of the .NET project; I stand by my concerns, expressed in that paper. For this reason, let me say again, that any monopoly on this peculiar market of software products creates an incredible danger to freedom and democracy in the broadest, most noble sense of these - so important to humanity - words. The danger is even greater here than in other parts of economy (oil, gas) or - as mentioned before - in politics. Software is a public, common, and strategic good in an information society; especially the kind of software that controls the functioning of the global net. It can not, it should not be under control of any monopoly, except for maybe one: the monopoly of democracy.

The monopoly of democracy. Isn't there a joke hidden in that? I don't know. We cannot hide the fact that democracy is also a kind of totality. However, the totality that constitutes doing EVERYTHING that influences OUR lives "behind the open doors," with OUR participation, seems acceptable to me. And a democracy of the FS&OSS movement can be brought to that. If we believe that our matters don't need to rest in the hands of specialists separated from the rest of society by Monopolsoft Corporation - or any other.

That is why We, the people, citizens of Information Society and of Informational Society have to demand from ourselves, our governments and from the agency representing Europeans - The European Parliament - that a program promoting UNIVERSAL and rational production of FS&OSS will find its place in the forefront of economic programs of those countries. We need Washington, Moscow, Tokyo and Brussels alike to accepted it. Such are the demands of the times: the times that are, and those that are coming.

### 3.4. Europe catching up on America.

Software production is enormously important for any modern economy. That is why Americans can watch the actions of the corporation that monopolizes the software market with

understanding; because to them - at least on the level of the national economy - those actions are beneficial. However, it so happens that besides the North-American continent and its market there are several others, including Europe and the European market. From the European point of view, matters look a little different. It seems that we are paying huge sums of money for product, which is not good enough, while being exposed to some absurdly unilateral licensing and patent actions of the monopolist; unfortunately, they are in many cases backed by irrational laws. Can one be satisfied with that?

Americans can say: You don't like it? Make your own software!

I can honestly say that I haven't heard advice this good for a long time. Yes, I agree. Like in no other matter, Europeans should take this advice of their younger brothers, and prepare a great European project (open to everybody) of encouraging the production and use of FS&OSS. They should also plan and finance other corresponding great educational, political and legal programs, without which such an ambitious task can not be realized.

European politicians are looking for ways to decrease the distance, especially the economic distance, between Europe and America. The hypothetical advice of the younger brother from the other side of the ocean presented above is for this problem like a winning lottery ticket.

The creation of open source software that is fairly paid-for can benefit the economy of many countries, not only in Europe. And national governments or international organizations like the European Parliament could finance well-thought-through assistance programs to support the work of computer programmers, a great many of whom for financial reasons cannot afford to join the somewhat exclusive open source club - in the first meaning of the term "open source." Instead of working in fast-food restaurants, scores of talented young computer programmers could make a living, for example, by testing professionally created open-source software. Ingenious virus-makers should receive prestigious awards for their help in popularization of secure software.

To summarize: there seems to be no reason for which Europeans couldn't finance rational production of good software and to be successful economically in this area.

### 3.5. Software is a strategic resource.

So, are there any **fundamental reasons** why the task described in the main thesis of this paper should be pursued? Yes, there are. Governments **should finance the creation of good software** the way they finance the construction of highways used by all, schools attended by our children, museums and libraries in which our cultural heritage is preserved, hospitals, and so on. In addition, as mentioned before, free, safe, more efficient, more elegant, more user-friendly software is as much a strategic product as some natural resources.

To clarify any possible doubt, I shall add that from the above it does not follow in any way that governments should finance the production of good software by placing orders with commercial companies, especially with monopolistic corporations. The aim of the governmental financing should be the creation of world-spanning cooperation of producers and users of FS&OSS.

I repeat: control over the global information infrastructure should not rest with any monopoly. Open source software is the only reasonable existing alternative to any software source that - with the help of trade secrets or intellectual property ownership regulations - tries to dominate

the global market (and also our PCs!), thereby threatening democracy. Not to mention the emptying of our pockets.

#### **4. Appendix**

I propose the creation of “Open Source Forward”, an international long-term (preferably 10-years or more) assistance program that should contain, among others, the following provisions:

- subsidizing national network services specializing in delivery of complete, up-to-date information about open-source operating systems and good open-source applications;
- subsidizing individual computer programmers and companies creating open-source systems in national languages;
- creating laws mandating retailers to disclose how much of the price of a computer system is the cost of the added operating system (Windows, Linux, etc.);
- saturating governmental and self-government agencies with open source software and applications;
- preferentially taxing businesses - that is, taxing them less - that install and use open-source operating systems;
- creating laws prohibiting the use of operating systems other than open-source systems in programs created for the government or for self-government agencies.

#### **5. References**

- Castells, Manuel (1998), *End of Millenium*, Blackwell.
- Goban-Klas, Tomasz (1999), in Polish, *Media i komunikowanie masowe*, Warsaw.
- Kocikowski, Andrzej (2002), *Hypernomadization of Society Within the Context of Monopolization of the Software Market*, *Proceedings of ETHICOMP 2002*, pp.391-402.
- Machlup. Fritz (1962), *The Production and Distribution of Knowledge in the United States*.
- Perens, Bruce (n. a.), *The Open Source Definition*, [<http://perens.com/Articles/OSD.html>].