

Comparing motivations of individual programmers and firms to take part in the Open Source movement. From community to business

Andrea Bonaccorsi, Cristina Rossi*

Laboratory of Economics and Management

Sant'Anna School of Advanced Studies

Pisa, Italy

1. Introduction

Stallman proposed a revolutionary idea in 1984 with the “Free Software Foundation”, subsequently confirmed in 1998 in the “Open Source Definition¹”. The key concept is that there should be unrestricted access to computer programming codes: anyone should be placed in the position to use and modify them and circulate any modification without having to pay licence fees.

According to the theorists of this model of software development, when programmers are allowed to work freely on the source code of a program, exchanging files and ideas through the Internet network, this will inevitably be improved because collaboration helps to correct errors and enables adaptation to different needs and hardware platforms². In this way quite complex programs can be written down efficiently without a central organisational authority in control.

Developers from all around the world work asynchronously on the various Open Source projects, sometimes during their spare time and weekends. The hierarchically organized and top down planned structure adopted in most productive processes is abandoned in favour of a new

* Corresponding author. Address: P.zza Martiri della Libertà 33, 56127, Pisa; Tel.: +39 050 883343; Fax: +39 050 883344. *E-mail address*: cristina.rossi@iit.cnr.it

¹ Open Source Definition, <http://www.opensource.org>.

² Introduction to Open Source, <http://www.opensource.org>.

kind of bottom up organizational structure, which is non-coercive and largely decentralized, even if shared behavioural rules are still present.

The first urgent question deals with Open Source developers motivations. Why Open Source programmers *write, read and revise all that code* for free Glass (1999, 104)? The issue is undoubtedly challenging and psychologists, sociologists, economists and computer scientists show interest in the motivations that lay at the basis of the participation in the Open Source movement. A growing body of economic literature has been addressing the problem since when the phenomenon went out from universities and research centres and became “tremendously successful” (Lerner e Tirole, 2001, 819) creating new and often profitable business models.

Nevertheless, up to now, the number of empirical investigations has been out of proportion to the theoretical debate. Few studies have collected empirical data on motivations to take part in the Free Software movement. Most of these analyses deal with the decision processes of individuals that develop software within Open Source projects (Bates et al. 2002; Ghosh et al., 2002; Hars and Ou, 2002; Hertel et al., 2003) or provide help within the on-line field support systems, which are operated by and for users themselves (Lakhani and von Hippel, 2003). Some authors have instead taken into account the motivations leading individuals (Fink, 2003), firms and other organisations (Wichemann, 2002a; Cogenio, 2001; Schimitz, 2001) to run Open Source software on their computers.

In brief, studies focusing on individuals have analysed motivations to demand and supply Open Source software while only the demand-side have been addressed for organisations. The aim of this paper is to supplement these researches taking into account the motivations of the firms that *supply* Open Source based products and services. These firms make effective the new business models that allow to link up the Free Software and the business world.

During 2002, we conducted a large-scale survey on Italian firms supplying Open Source solutions. We contacted 251 software firms and obtained 146 valid answers³. Our data deal with

³ Because there are not directories of firms producing Open Source software ,we followed a snowball procedure. We started by addressing firms participating to newsgroups and Linux associations and

firms' structural characteristics, business models and attitudes towards the Free Software and its community (Bonaccorsi and Rossi, 2003a). The questionnaire submitted to the firms included a set of answers about the motivations that have led the agents to operate on this new market.

In this paper our data on firms' motivations are compared with the ones collected by the above-cited surveys conducted on individual programmers. Statistical tests checking for differences in motivations in the two groups of agents are carried on using the data gathered by Hertel et al. (2003) on the developers of the Linux kernel⁴. Our main goal is to analyse the role played by different kinds of motivations (social, economic and technological) in determining the involvement of different set of agents in Open Source activities. Firms and individual programmers are undoubtedly heterogeneous with respect to the goals that they want to achieve by taking part in the Free Software movement.

Monetary reward is likely to be the main incentive of the firms. In fact, they enter this new field in order to profit from the Open Source. Contributions and feedbacks from the Free Software community allow them to lower down development costs while the Open Source licence schemes emancipate them from the licence policies of large software companies. As a consequence, motivations dealing with economic aspects are likely to play a leading role for firms.

Several studies have highlighted that monetary rewards are not so crucial in determining the participation in Open Source projects by individual programmers. In general, developers are not paid for their programming activity within Open Source projects. Their motivations resemble the ones that lay at the basis of the participation in the scientific community (Bezroukov, 1999; Dalle and David, 2003). Gaining a reputations among the peers, improving programs through the feedbacks of the community, learning and developing new skills shape the decision process that lead to contribute to a project (Bonaccorsi and Rossi, 2003b). Moreover programmers think that their Open Source developing activity signals their talent to software companies assuring

submitted the questionnaire on line. During the phone follow up we have asked to refer to other firms in the OSS field they were aware of. We stopped the inquiry when no new address was made available.

them professional and career advancements (Lerner and Tirole, 2002; Bates et al., 2002). At the same time many developers conform to the values of the Open Source community. They think that software should not to be a proprietary good (Ghosh et al, 2002) and declare that they are proud to be part of the Open Source community (Hars and Ou, 2002).

Finally, we address the linkage between agents' motivations and intensity of participation in the Free Software movement. We use the number of projects participated and coordinated, the percentage of lines of code (LOC) contributed on average to each project and the number of contributions (patches, modules) included in the official versions of Open Source programs as indicators of the level of Free Software engagement. Correlation analyses and regression models are carried on in order to study whether firms and individual programmers are heterogeneous with respect to the motivations leading to a deeper engagement in Open Source activities.

The paper is organised as follows.

First of all we review the literature on developers' and firms' motivations in taking part in the Open Source movement following the taxonomy proposed by Feller and Fitzgerald (2002). This taxonomy takes simultaneously into account the nature of the motivations (distinguishing among economic, sociological, technological motivations) and the one of decision agents (individuals vs. organisations).

Section II focuses on the literature addressing the incentives of individual programmers while section III is devoted to the motivations of the firms supplying Open Source solutions.

Section IV describes the sample and the variables used to test our hypotheses. A comparison between the firms in our sample and the individuals programmers surveyed by Bates et al. (2002), Ghosh et al. (2002), Hars and Ou (2002) and Hertel et al. (2003) is performed. In particular we identify questionnaire items that are comparable with the ones of the survey conducted by Hertel et al. (2003). We use their data on developers of the Linux kernel and carry on statistical tests for differences in motivations between firms and individual programmers.

Section V summarises the main conclusions of the paper.

⁴ We gratefully acknowledge the authors for providing us their data.

2. *Motivations in taking part in the Free Software movement by individual developers: an interdisciplinary taxonomy.*

At present new actors are coming into sight within the Free Software outlook. Large software companies have started to disclose the source code of their programs in order to exploit the contributions of the Free Software community while new firms are entering the software market with business models based on the supply of Open Source products and services. Because of the increasing heterogeneity of the agents taking part in the Free Software movement, understanding the structure of their incentives is more and more challenging.

In order to set up a comprehensive taxonomy of the motivations to take part in the Open Source movement, we follow the framework proposed by Feller and Fitzgerald (2002). The two authors take into account three broad motivational area (economic, socio-political and technological) and further subdivided each of these into micro-level (individual developers), and the macro-level (organization/community). Table 1 summarizes the motivations of the individual programmers.

<i>Motivation area</i>	<i>Micro level</i>	<i>References</i>
Economic	Monetary rewards	Lerner and Tirole (2002b), Feller and Fitzgerald (2002)
	Low opportunity costs	Bonaccorsi and Rossi (2003b), Lakhani and von Hippel (2003), Kolloch (1999)
	Gaining a reputation among peers	Dalle and David (2003), Bezroukov (1999)
	Gaining future career benefits	Lerner and Tirole (2001, 2002a)
Social	Fun to program (Loving to code)	Green (2000), Torvalds and Diamond (2001)
	Altruism (gift economy)	Mauss (1959), Raymond (2000), Bergquist and Ljungberg (2001)
	Sense of belonging to the community	David and Pfaff (1998), Raymond (2001b)
	Fight against proprietary software	Stallman (1984), Ullman (1998)
Technological	Learning	Lakhani and von Hippel (2003)
	Contributions and feedback from the community	Raymond (2001a), Bonaccorsi and Rossi (2003b)
	Working with a bleeding-edge technology	David and Pfaff (1998), Pavlicek (1999b)
	Scratching a personal itch	Green (2000), Franke and von Hippel (2003), Feller and Fitzgerald (2002)

Table 1: Taxonomy of individual programmers' motivations.

According to the economic theory, it is possible to trace back the motivational factors in Open Source to a classical cost-benefit framework, giving the right prominence to the constituent elements.

Some authors (Kollock, 1999; Kuster et al., 2002; Osterloh et al., 2002) argue that engaging in Open Source projects represents a *low cost situation*. The infrastructure investment for taking part in the movement is almost zero. Most of the programmers have their own computer for studying or working or even write the code at work. Hertel et al (2003) in their survey on Linux kernel developers find that 38% individuals *carry out Linux-related programming during their regular working hours*, although this is not an official duty. In the same way Internet connection is no more a problem for most of the programmers. The presence of contributions from all over the world is not much more than a romantic idea: most of the LOCs added to different projects are from European or United States developers, that clearly have no connectivity problems. According to Kollock (1999) the Internet makes it possible for software developers to access a very wide audience with very low costs. As a consequence, posting code on the Internet can pay off even if the expectations for helpful comments and contributions from the community of the Open Source developers are relatively low. Given that, the main investment involves time and intellectual resources. With regards to this point, several studies underline that investment in time is quite limited too. In an analysis of more than 1,700 individuals taking part to an important newsgroup that gives on line help about the Apache Web server, Lakhani and von Hippel (2003) find that over 80% of subjects answering to the help requests spend an average time of five minutes for giving the solution to the question. The reason of this is quite simple: programmers know the answer as they know the operation of the newsgroups, the main e-mail programs, the programs for file compression that, reducing file dimension, allow them to circulate easily on the Net. Most of the developers had the possibility to work with Unix from which many Open Source programs come from. All this gives to them the possibility of bearing their competencies easily to the problem at hand.

Non-negligible benefits stand against low costs. First of all some well-known myths about Free Software need to be reappraised. Not all the programmers work for free. Surveying 81 individuals involved in Open Source projects, Hars and Ou (2002) find that 16% respondents

are directly paid for their Open Source programming⁵. Moreover some licence schemes drafted in compliance of the dictates of the Open Source Definition⁶ allow *to build a commercial operation around the Open Source code* (Lerner and Tirole, 2002b). Developers can add new features and functionalities to the code developed within an Open Source project and include them into a proprietary program.

From the point of view of the economic theory motivations dealing with intrinsic utility, gaining a reputation among one's peers, signalling quality of human capital and learning are outstanding within the Open Source movement. Such elements have always been the moving cause of the scientific community (David, 1991).

Open Source programming is a form of intellectual gratification whose intrinsic utility resembles the one of the scientific discovery (Dalle and David, 2003). Emerging as it does from the university and research environment, the movement adopts the motivations of scientific research, transferring them into the production of technologies that have a potential commercial value. The new way of developing code is basically nothing other than the natural extension of Western scientific culture (Stone, 1999). The process of scientific discovery involves the sharing of results, just as the dictates of the Open Source movement involve sharing source code. Sharing results enables researchers both to improve their results through feedback from other members of the scientific community and to gain recognition and hence prestige for their work. The same thing happens when source code is shared: other members of the group provide feedback that helps to perfect it, while the fact that the results are clearly visible to everyone confers a degree of prestige which expands in proportion to the size of the community. A further element is that working on Open Source projects provides the prestige and visibility that often gives programmers the chance to be noticed by commercial software companies (Lerner and Tirole, 2001, 2002a) . All of the founding fathers of the movement have earned enough of a

⁵ A remarkable finding about this issue is reported by Hertel et al. (2002). In their survey on the developers of the Linux kernel they found that about 43% of the programmers receives some form of income while 20% receives a salary "for their Linux programming on a regular basis".

reputation to allow them to pay the rent and maintain their kids (Di Bona, Ockman, Stone, 1999, 14). By way of contrast, many of the signalling benefits arise even if the subsequent work of the programmer is unrelated to the open source project.

The founding fathers of the Free Software movement, that were the very first to address motivations of Open Source developers, focused mainly on a set of *social* incentives. Raymond (2001a) traces back programmers' incentives to the values of the so-called hacker culture. Open Source developers are considered the heirs of the *real programmers* of the immediate post-war period, *who came from the engineering and physics fields...[and] programmed in FORTRAN and another half-dozen languages that have now been forgotten* (Raymond, 2001a, 2001b). Hackers, who are basically computer scientists, come prevalently from the academic world or from the research centres of large software corporations – MIT's Artificial Intelligence Lab in Boston, the Laboratory of Artificial Intelligence at the University of Stanford, the Computer Science Department at Berkeley, the Xerox Park, to name just a few. These software developers regard programming as an art form and feel an artistic satisfaction in writing code. Such a satisfaction is quite similar to the one that music gives to a composer and painting gives to an artist. Green (2000) points out that programmers love the code and defines them *source code mavens*. They think of the Open Source production paradigm as a way of recovering the pleasure of the creativity that is disappearing within the commercial software framework. In particular Free Software developers are not subject to the pressures that in software houses are induced by corporate announcements of more up-to-date and efficient releases.

Sociology and social psychology number the hedonistic (or playful) motivation among the determinants of taking part in whatever social movement, while altruism is considered as lying at the basis of many human behaviours (Monroe, 1996). The so called *gift economy* (Mauss, 1959) states that giving away goods and services for free allows to make and maintain social links and implies the duty to reciprocate. This happens also when the exchange is not in favour

⁶ The Berkley Software Distribution licence (BSD), for instance, allow for the mixability with proprietary licensed products.

of well-known individuals but of a community of unknown subjects. In fact, people hope to have help and support in future given that the previous contributions have created a tacit reciprocity agreement. In writings on the Open Source software development model, it is often argued that it is successful as a result of the gift economy that embraces activities in online communities. Raymond explicitly declares that the society of Open Source hackers is in fact a gift culture (2001b). According to the author gift giving practises rely on two factors: abundance of resources (as opposed to scarcity) and willing to signal social status. In the society of open-source hackers *there is no serious shortage of the 'survival necessities'* - disk space, network bandwidth, computing power. As a consequence the software is freely shared and this creates a situation in which the only available measure of competitive success (social status) is reputation *among one's peers*. Bergquist and Ljungberg (2001) propose a similar argument. The authors, examining *gift-giving practices in digital domains*, relate the code sharing within the Free Software community to the practise of sharing discoveries within the scientific community. *Giftng away the code* allows getting new ideas and prototypes out into circulation. In brief, the altruistic motivation resembles the economic incentive of the reputation gain.

Such effects are strengthened by the identification of the programmers with the community (collective identification, Simon et al., 1998). Collective identification is an important aspect of the Open Source movement. During an interview with the *New York Times*, Linus Torvalds declared that *making Linux freely available ...was a natural decision within the community that I felt I wanted to be part of*.

Developers belonging to the Free Software community conform to the values of knowledge sharing and freedom that lay at the basis of the Open Source production mode. They often think that their contribution is fundamental for success of the projects in which they are involved (*instrumentability*, Hertel et al., 2002) and attribute a great importance to project goals.

Collective identification is enhanced by the presence of a mutual enemy. Hackers are highly suspicious of the *customerization of the computer* (Ullman, 1998) that, in their opinion, has been favoured by large software companies in general and Microsoft in particular.

The fight against the monopoly power of the software giants shapes the history of the Open Source movement. The AT&T decision of turning the UNIX operating system into a commercial program triggered the reaction of Richard Stallman, a computer science researcher at the MIT Artificial Intelligent Laboratory in Boston. In 1984 Stallman resigned from the MIT and established the *Free Software Foundation* that is often regarded as the very starting point of the Open Source movement. Stallman's aim was to fight against the conversion of software into a proprietary good. He resolved to develop a complete Unix-compatible software system and give it away free to everyone who can use it. In order to keep aloof from UNIX, he called this project *GNU*, a recursive acronym that means *Gnu is Not Unix*. In the GNU Manifesto (1984), written down at the beginning of the GNU Project to ask for participation and support, Stallman declares that in order to *continue to use computers without dishonour*, he has *decided to put together a sufficient body of free software so that he will be able to get along without any software that is not free*. In his opinion software companies are prejudicial to users' interests by forbidding them to share and modify the source code of computer programs. In this way they have broken *solidarity* among the users and are going to *conquer the market* according to a *divide et impera rule*.

Moreover, allowing distribution of computer programs only in binary form software companies forbid the programmers to study the code written by the others. The new development paradigm constitutes an immense learning opportunity: having all the code they want, programmers can study it very deeply and use it for implementing new solutions. This aspect is crucial in the software industry characterized by incremental and sequential innovations.

Learning play a leading role also in explaining the so-called "non sexy" work involving, for instance, the development of graphical interfaces, the compilation of technical manuals, the on line support in newsgroups and so on. According to Lakhani and Von Hippel (2003) learning from reading the answers to the questions posed by the users is one of the most important motivations for participants to the newsgroups devoted to the resolution of the problems that arise from Open Source programs. Not counting that many developers are computer science

students that take part in Open Source projects for finding materials for their undergraduate or graduate theses. In this way they develop skills in programming that are going to be useful when they enter the labour market.

Learning opportunities are enhanced by the feedbacks that programmers within Free Software projects obtain not only from other developers but also from users. According to Raymond (2001a) two laws rule the Open Source production paradigms. The former deals with the activity of bug reporting and fixing within Open Source projects. It states that *given enough eyeballs, all bugs are shallow*. Not only developers but also users test Free Software programs whose source code is freely available on the Internet network. This testing procedure is potentially carried on by a large number of subjects. As a consequence it is likely to be more efficient than the one performed by a limited number of beta testers within software companies. Such practise may therefore lead to quick implementation of new features and solutions thank also to the interplay with the other Open Source development law: *release often and release early*. Early releases of the Open Source programs enhance the benefit of decentralized beta tests. Bug fixing and new features are quickly included in the official version of the program. The software is then ready for another decentralized beta test.

According the founding fathers of the movements (David and Pfaff, 1998) this procedure is likely to lead to software programs representing a *bleeding-edge technology*. Several Open Source packages are well known for their high reliability. Pavlicek (1999a) reports that the Web server equipped with the Linux RedHat 4.2 operating system and Apache, that he installed in 1997 in the laboratories of the Compaq Computer Corporation, resisted 267 times before crashing and even then it only did so following a massive storm that led to a total power cut in the building where it was housed. Liebovitch (1999, 42)numbers reliability among the characteristics that make Linux “an easy sell”, pointing out that in comparison to Windows NT, it has a resistance to crashing that can be measured in *months and years rather than days or weeks*. The operating system designed by Torvalds is also highly portable. *Nowadays Linux*

runs on anything, from the PalmPilot to the Alpha workstation, and is the operating system most subject to porting amongst those available for PC (Torvalds, 1999, 112).

Feedbacks and contributions from the community turn out to be very useful when programmers need to implement a software solution, which is not already available on the market.

Filling an unfilled market (Green, 2000; Feller and Fitzgerald, 2002) is another important incentive to write Open Source code. Many Open Source projects take shape because people promoting them have looked in vain for a program performing a particular task. They arise, that is, to satisfy a demand for which there is no corresponding supply. One typical example is the Perl language developed by a systems administrator who saw a need for something combining the speed and immediacy of the Unix shell languages with the possibility to develop more complex programmes, a typical feature of high-level languages like C. Unable to find that something, he created and then distributed it to the entire Open Source community. Today Perl is a commonly accepted language used above all for Internet-related applications.

3. Profiting from Open Source software. Why do firms engage in Open Source activities?

Despite the growing importance of the new Open Source based business models, the literature has devoted scarce attention to the motivations behind the engagement of firms in the Open Source movement (Wichmann, 2002).

According to Kuster et al. (2002), actors in the Open Source community can be divided into two tiers. In the first one there are the [individual] developers of the source code. The second tier is made up of firms that commercially use the source code, which was developed by programmers in the first tier. These two groups of agents are heterogeneous with respect to the goals that they want to achieve by taking part in the Open Source movement. As a consequence the sets of their incentives overlap only partially.

Exploring such heterogeneity of motivations, we have to address the distinction between intrinsic and extrinsic motivations. On one hand, a motivation is extrinsic if needs are satisfied indirectly, especially through monetary compensation (Osterloh and Frey, 1998). On the other

hand, there is an intrinsic motivation when an activity is undertaken for one's immediate need satisfaction. In particular intrinsic motivations can stem from the pleasure of carrying out an activity (Kuster et al., 2002).

It comes out from the previous discussion that developers share both intrinsic and extrinsic motivations. On the other hand firms' incentives are mainly extrinsic. Empirical data witness the leading role played by intrinsic motivations in determining the participation of individual programmers in Free Software projects. Developers often number aspects such as *fun to program, sense of belonging to the Open Source community, hope in generalized reciprocity and agreement about the values of the Free Software movement* among the main incentives to carry out Open Source activities (Bates et al., 2002; Ghosh et al., 2002; Hars and Our 2002; Lakhani and von Hippel, 2003).

Table 2 summarizes firms' incentives to carry out Open Source activities. Also in this case, we distinguish among economic, social and technological motivations. Nevertheless the aim of profiting from involvement in Open Source projects shape firms' motivations, which therefore can be wholly classified as extrinsic.

<i>Motivation area</i>	<i>Macro level</i>	<i>References</i>
Economic	Being independent of the price and licence policies of the large software companies	Lerner and Tirole (2002b)
	Addressing the new model of software as a consumer-driven service (making money on complementary services)	Feller and Fitzgerald (2002), Wichemann (2002a), Lerner (2002)
	Obtaining indirect revenues by selling related products	Lerner and Tirole (2002a), Wichemann (2002a), Feller and Fitzgerald (2002)
	Affording innovation (by exploiting the R&D activity of the Open Source community)	Hawkins R. E. (2002), von Krogh et al. (2002)
	Hiring good IT specialists	Fink (2003), Wichemann (2002b), Lerner (2002)
Social	Conforming to the values of the Open Source community (in order not to betray the trust of the Open Source developers)	Kuster et al. (2002), Lerner and Tirole (2002b)
	Sharing code and knowledge with the community (reciprocating in order to sustain cooperation)	Kuster et al. (2002), Lerner and Tirole (2002b), Franck and Jungwirth (2002)
	Thinking that software should not to be a proprietary good (in order to reduce market power of the large software companies)	Feller and Fitzgerald (2002)
Technological	Exploiting feedbacks and contributions from developers of the Open Source community for cutting development costs and improve the software	Hawkins R. E. (2002), Lerner and Tirole (2002b), Hecker (2000)
	Exploiting feedbacks and contributions from the user community for testing and improving software	Aoki et al. (2001), von Hippel (2002), Fink (2003)
	Cutting hardware costs	Feller and Fitzgerald (2002)
	Promoting standardisation	Fink (2003), Wichemann (2002a), Lerner (2002)
	Addressing security issues	Fink (2003)

Table 2: Taxonomy of firms' motivations.

An important economic incentive to take part in the Open Source movement deals with the emancipation from the price and licence conditions imposed by large software companies. The business models of firms working with Open Source software are mainly based on the supply of software related services (Wichemann, 2002a). The Open Source framework, in fact, does not allow to profit from licence fees. If these firms supplied services such as installation, integration, maintenance or support based on proprietary programs, they should pay licence fees to the companies owning proprietary rights on such programs. Working with Open Source software these firms avoid licence fees. This lowers down their production costs. The break-even point is then reached at lower sales levels contrasting the tendency toward concentration of the software market.

The new business models of Open Source firms address the evolution of software from a commodity to a service driven market. Software is getting more and more complex. As a

consequence both private and corporate users have increased their demand of support services. This has paved the way for firms that want *to make money on complementary services* rather than selling software (Lerner, 2002). Nevertheless the availability of services for Open Source products turns out to be a controversial issue. Some authors (Gill, 2002) number the lack of after-sales assistance among the obstacles to the diffusion of the Open Source software⁷.

At the same time firms decide to engage in Free Software activities in order to obtain indirect revenues by selling related products. This is the strategy followed by large software firms that have chosen to release the source code of their programs to the Open Source community. The aim to promote the sales of their e-commerce servers is numbered among the motivations beyond the decision of IBM to invest on Linux.

The linkage between Open Source and innovation processes is widely debated in literature (von Krogh et al., 2003). On one side it has been claimed that *Open Source is about imitation, not innovation* (Pavlicek, 1999a), pointing out that most of Open Source programs are imitations of proprietary solutions. On the other side it has been underlined that the free circulation of ideas within the bazaar production mode makes up an environment that is particularly suitable for innovation (Raymond 2001a). This is witnessed by the several Open Source successful projects that have been initiated by someone that had a technical need that was not being addressed by available proprietary (or Open Source) technologies (Feller and Fitzgerald, 2002). 2% of the firms in our sample declare that all their turnover originates from solutions that have no parallel in the proprietary world while more than 8% do the same with reference to Open Source world. Firms that release the source code of their programs to the Open Source community can exploit the R&D activity carried on by individual developers.

Moreover firms benefit from the spillovers of the R&D activity pursued by other firms that eventually download source code from the Internet, adapt it to their needs and, in their turn, places the changes at disposal of the community. Within this framework also small firms are

⁷ Above 60% of the firms in our sample think that the lack of after-sale assistance is an important obstacle to Open Source diffusion.

likely to afford innovation. Our data support this statement: above 40% of the firms in our sample strongly agree that Open Source software allows small enterprises to innovate. Moreover innovation processes are likely to be pushed by the talented programmers working within Open Source projects (Lerner, 2002) that firms can spot for hiring purposes. In his survey on organisations that have adopted Open Source software, Wichemann (2002b) finds out that 21% of the interviewed firms agree about the easier availability of IT specialists in the Free Software world.

Social motivations need to be reinterpreted in the light of the firms' goal of profiting from Open Source software. According to Osterloh et al. (2002) firms who use [Open Source code] face considerable competitive disadvantages if they do not respect the intrinsically motivated rules of cooperation and trust in Open Source community. Firms have to conform to the Free Software values in order not to betray the trust of the community of the developers. Behaviours such as including pieces of Open Source code into proprietary programs or keeping close parts of the code of the programs released to the community contrast with the norms ruling Free Software projects. As a consequence they are likely to bring down cooperation, reducing the incentives of individual programmers to contribute to Open Source projects in which firms are engaged (Kuster et al., 2002).

This is witnessed by the unlucky decision of Netscape to release the source code of its web browser. Netscape released its web browser Netscape Navigator to the Open Source community in 1998 to oppose the increasing market power of Microsoft Internet Explorer. Nevertheless the source code of the program was released only partially, not opening modules in which Open Source developers were interested. Moreover Netscape selected a licence scheme (the Netscape Public Licence, NPL) allowing it to hijack the modifications made up by the developers. Because of this policy, Netscape was able to attract only few contributions from the Free Software community. The 1.0 version of Mozilla was released only in 2000 while Netscape decided to shift to a GPL-like licence scheme (the Mozilla Public Licence, MPL) in order to try to win back the trust of the Open Source developers.

The cooperation between firms and individual programmers resembles the form of a *psychological contract* (Osterloh et al., 2002). A psychological contract is defined as *a set of beliefs about what each party is entitled to receive, and obliged to give in exchange for another party contribution* (Levison et al., 1962). Psychological contract framework has been applied in particular to the analysis of the obligations within employee-organization relationships (Morrison and Robinson, 1997).

Several scholars (Allen and Lucero, 1994; Mclean Parks and Schmedemann, 1994; Kratz et al., 1994) agree that a psychological contract has two main features that distinguish it from related constructs. On one hand *beliefs about the obligations underlying [the] relationship are not necessarily shared by the agents and the organization* (Rousseau, 1989), unlike what happens within the so called *implied contracts*⁸. On the other hand psychological contracts *are based on perceived promises* (Rousseau and McLean Parks, 1993). Not only obligations that are established *via formal contract are encompassed* but also *perceived obligations that result from more implicit* means. Both these aspects emerge within the relationship between firms and individual developers.

First of all heterogeneity of expectations shapes such relationship. Firms try to adhere to the social norms that rule the Open Source community. As we have underlined above, in fact, the observance of these norms lays at the basis of the cooperative behaviour of the individuals programmers. Moreover firms often provide financial support and *staff their talented programmers* (Lerner, 2002) to Open Source projects in which they are interested, as IMB is doing with the Linux operating system. In turn, firms expect that developers are bound by the promise of an endless stream of feedbacks and contributions steadily lowering down their R&D costs.

Reciprocation is very important for the achievement of this goal. Firms have to cooperate in order to sustain cooperation. They have to *gift* their code returning the one contributed by individual developers in order to enforce the generalised obligation to gift that characterizes the

Open Source mode of production. In this way a virtuous circle of mutual contributions is established.

Moreover, several minor expectations lead firms to pursue the cooperation with the Free Software community. At present the Open Source phenomenon arouses lively interest not only among scholars but also in the public opinion. Taking part in Open Source projects is therefore a strategy to improve the corporate image. Working with an emergent technology that is one of the possible solutions to the *software crisis*, in fact, is likely to strike customers and venture capitalists favourably (Feller and Fitzgerald, 2002).

As we have highlighted before, many individual programmers (Gosh et al., 2002), in turn, engage in Open Source activity in order to gain a reputation among their peers, signal their talent to software houses and learn by studying the code written by the other developers. They can achieve these goals only if the code to which they contribute is under everybody's eyes. That is, if it is kept open (Lerner and Tirole, 2002a, 2002b). They feel therefore bound to contribute to Open Source projects in which firms are involved only if they expect that their code is not hijacked and inserted in proprietary programs. Nevertheless firms can commit themselves to keep the code open only by using an Open Source licence scheme that explicitly forbids the inclusion of Open Source code into proprietary programs (Lerner and Tirole, 2002b). As a consequence programmers prefer to work on projects with a GPL-like licence and firms' expectations of contributions to projects with different licence schemes may be unfulfilled.

GPL-like licence schemes are a formal mean for binding firms not to hijack the code. The contribution of the developers is instead an *implicit promise* based on the non written rules of the Free Software community. Individual developers follow established patterns of behaviour against firms that violate these rules. In particular *breaking of rules is sanctioned by flaming intended to cause shame in the rule-breaker* (Osterloh et al., 2002). Moreover if *firms are insensitive to shame*, Open Source developers are likely to *stop cooperating or simply migrate*

⁸ Implied contracts consist of commonly understood and shared expectations (Morrison and Robin, 1997).

to other projects. As a consequence the perception of an obligation of the firms to not violate these rules arises.

Firms' technological motivations resemble individual developers' ones. First of all, feedbacks and contributions from the Open Source community allow firms not only to cut software development and R&D costs but also to improve their software through the bug fixing activity. The learning aspect that plays a leading role at the micro level is very important also for organisation: ideas gathered from the Open Source projects may be used to develop commercial solutions (Lerner, 2002). At the same time, user community form a very large set of beta testers for firms programs. Nevertheless it has been underlined (Franke and Von Hippel, 2003; Von Hippel, 2002) that such an activity is carried on only by advanced users.

Moreover firms can carry out Open Source activities in order to *pre-empt the development of a standard around a technology owned by a powerful rival* (Lerner, 2002). Several authors (Feller and Fitzgerald, 2002) have also underlined that Open Source software lowers down hardware costs (National Advisory Council on Innovation Open Source Software Working Group, 2002) and address very well security issues.

4. An empirical analysis of Open Source firms' motivations. A comparison with individual programmers

In order to gather data on the incentives that have led the firms in our sample to adopt business models based on Open Source, we asked them for a subjective appreciation of the reasons to supply Free Software solutions. We used a closed – response question made up by eleven items. Firms had to assign a score ranging from one (*I totally disagree*) to five (*I totally agree*) to each entry (five –point Likert scale). Items have been selected on the basis of the above cited literature on motivations to take part in Open Source activities. Following the taxonomy described in section III, we distinguish among economic, social and technological incentives and motivations.

Table 3 reports the list of the items and the descriptive statistics of the collected variables. Table 4 contains the percentages of the agents that have chosen a given entry distinguishing also between low (1-2) and high (4-5) scores.

<i>Motivation</i>	<i>Acronym</i>	<i>Area</i>	<i>Mean</i>	<i>St. Dev.</i>	<i>Median</i>	<i>Mode</i>
Because Open Source software allows small enterprises to afford innovation	M4	E	3.97	1.15	4	5
Because contributions and feedbacks from the Free Software community are very useful to fix bugs and improve our software	M8	T	3.89	1.21	4	5
Because of the reliability and quality of the Open Source software	M10	T	3.87	1.16	4	5
Because we want to be independent of the price and licence policies of the large software companies	M1	E	3.78	1.19	4	5
Because we conform to the values of the Free Software movement	M6	S	3.76	1.29	4	5
Because we want to place our source code and skills at disposal of the Free Software community and we hope that others do the same thing	M5	S	3.41	1.27	4	4
Because in the field of Free Software we can find easily good IT specialists	M3	E	3.37	1.25	3	3
Because we want to study the code written by other programmers and use it for developing new programs and solutions	M9	T	3.29	1.29	3	3
Because opening our source code allows to gain a reputation among our costumers and competitors	M2	E	3.10	1.20	3	3
For having products that are not available on the proprietary software market	M11	T	2.99	1.35	3	3
Because we think that software should not to be a proprietary good	M7	S	2.99	1.42	3	2

Table 3: Descriptive statistics of firms' motivations.

Note: E: economic motivations, S: social motivations, T: technological motivations

Scores are biased towards the right side of the scale. The percentages of high and low scores assigned to the items range from 38.1% to 71.9% and from 12.2% to 40.7% respectively. This witnesses that firms in our sample agree upon the set of the motivations that we have taken into account.

Moreover the distributions of the variables are quite concentrated around their mean values. These mean values range from 3.97 to 2.99 (variance: 0.14). Variables measured at a Likert scale often display this structure. The tendency of the respondent to chose the intermediate score 3 is numbered among the biasing effects caused by Likert scales (Friedman et al., 1993).

<i>Motivation</i>	<i>Acronym</i>	<i>Area</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>Low score (1,2)</i>	<i>High score (4,5)</i>
Because Open Source software allow small enterprises to innovative	M4	E	5.0	7.2	15.8	29.5	42.4	12.2	71.9
Because contributions and feedbacks from the Free Software community are very useful to fix bugs and improve our software	M8	T	6.4	7.8	17.0	27.7	41.1	14.2	68.8
For the reliability and quality of the Open Source software	M10	T	5.0	7.8	21.3	27.7	38.3	12.8	66.0
Because we want to be independent of the price and licence policies of the large software companies	M1	E	5.0	11.4	19.3	29.3	35.0	16.4	64.3
Because we conform to the values of the Free Software movement	M6	S	8.6	9.3	17.9	26.4	37.9	17.9	64.3
Because we want to place our source code and skills at disposal of the Free Software community and we hope that others do the same thing	M5	S	9.2	15.6	24.8	25.5	24.8	24.8	50.4
Because we want to study the code written by other programmers and use it for developing new programs and solutions	M9	T	12.2	14.4	26.6	25.9	20.9	26.6	46.8
Because in the field of the Free Software we can find easily good IT specialists	M3	E	7.3	19.7	26.3	21.9	24.8	27.0	46.7
Because we think that software should not to be a proprietary good	M7	S	20.0	20.7	19.3	20.0	20.0	40.7	40.0
Because opening our source code allows to gain a reputation among our costumers and competitors	M2	E	10.6	22.0	27.7	26.2	13.5	32.6	39.7
For having products that are not available on the proprietary software market	M11	T	19.4	16.5	25.9	21.6	16.5	36.0	38.1

Table 4: Percentage of the scores assigned to firms' motivations

Median and mode are particularly informative measures of the central tendency of Likert scale variables (Harvey, 1999). Nevertheless, in our sample, they take the same values for most of our variables. As a consequence we arrange motivations by their mean value. There are several interesting results from this preliminary analysis.

First economic and technological motivations are near the top. All of them display mean values that are above the mean value of the whole set of variables (3.49). Their median and modal values are respectively 4 and 5. Social motivations are ranked fifth and sixth.

Second, and even more interestingly, incentive ranking first deals with the role played by Open Source software in promoting innovation by small enterprises. Above 42% of the firms *totally agree* upon proposition M4 that displays also the highest percentage of high scores (71.9%) and the lowest percentage of low scores (12.2%)⁹.

⁹M4 displays also the lowest frequency of the value 3. Agents seem to have a clearer view of such incentive with respect to other ones.

This is a remarkable finding that gives a contribution to the debate upon innovation processes that take place within the framework Open Source. Our data, in fact, do not corroborate at all the hypothesis that *Open Source is about imitation, not innovation*.

The idea that within Open Source production mode even small enterprises can develop innovative software is likely to originate from the chance to exploit the contributions of the Free Software community. R&D investments of small firms are notoriously scanty. As a consequence they peculiarly benefit from the R&D activity carried out by the community of developers take advantage of the spillovers of the R&D activity of other firms¹⁰. Exploiting Open Source software small firms can carry out such processes without property rights infringements. Nevertheless it is worth to notice that the *fill an unfilled market* issue (M11), which is often regarded as an indication of the innovativeness of the Open Source production mode (Feller and Fitzgerald, 2002), ranks at the bottom of the list of the motivations. This is probably due to the business models of the firms in our sample. Most of them, in fact, supply Open Source based services (Bonaccorsi and Rossi, 2003a) providing support and integration for well established Open Source products. As a consequence, in carrying out their activity, they do not experience the need to develop new products from the very start.

Third, data on motivations ranking next corroborate some theoretical hypotheses on firms' incentives to take part in Open Source movement. Above 69% firms in our sample attach much importance to the cooperation with individual developers and 66% choose Open Source software because of its quality and reliability. The emancipation from the price and licence policies of the large software companies represent a crucial motivation for 35% respondents.

Among technological motivations, firms point directly to the fundamental properties of the Open Source.

As we already noted, social motivations dealing with the conformity to the values of the Free Software movement and the observance of the reciprocation rule rank 5th and 6th respectively.

The other social motivation, fighting for the software freedom (M7), ranks at the bottom of the list. These findings get into line with the literature regarding firms' social motivations as extrinsic. The respondents in our sample do not agree upon the fight against intellectual property rights carried out by the Free Software advocates. By adhering to the non written norms that rule the Free Software community, firms aim to satisfy cost saving and quality goals. In fact the observance of the Open Source rules allows them to sustain cooperation with individual developer community.

Finally, surveys on individual developers show that learning from the others and gaining a reputation among peers play a leading role in motivating individual developers to contribute within Open Source projects. The firms in our sample do not agree on this matter. Learning and reputation incentives rank respectively 8th and 9th next to the *easier availability of good IT specialists in the Free Software field*. In particular, the signalling motivation strongly emphasized by Lerner and Tirole (2002a) does not play any important role.

In order to get an aggregate measure of each set of incentives, we compute the mean values of the scores assigned by each firm to economic, social and technological motivations respectively. Table 5 reports descriptive statistics of these aggregate variables. Results corroborate once again our hypothesis on the minor role played by social motivations in determining firms' engagement in the Open Source movement.

<i>Variable</i>	<i>Acronym</i>	<i>N</i>	<i>Minimum</i>	<i>Maximum</i>	<i>Mean</i>	<i>Std. Dev.</i>
<i>Economic Motivations</i>	EM	143	1.5	5	3.56	0.80
<i>Technological motivations</i>	SM	142	1	5	3.51	0.79
<i>Social motivations</i>	TM	142	1	5	3.39	1.07

Table 5: Descriptive statistics of the aggregate variables of each motivation set.

Taking into account the mean values, the variables summarizing economic motivations (EM) ranks first followed by the ones summarizing technological (TM) and social (SM) incentives.

¹⁰ The role of R&D activity carried on by other firms in promoting innovation represent a controversial issue. About xxx of the firms assign a high score to the item *The solutions developed by the other firms [working with Open Source Software] often are not directly utilizable for our costumers*.

Paired-samples T test¹¹ shows that the differences in the mean values of EM and SM are statistically different from zero (p value = 0.031).

The analysis above sheds lights on the heterogeneity of firms and developers with respect to the incentives that lay at the basis of their participation in Open Source activities. Following a methodology that is well established in Social Psychology studies (Wachter and Straf, 1990; Schmidt, 1992), we deeply investigate this issue using a meta – analytic approach.

We take into account four surveys on Open Source individual programmers (Bates et al. 2002; 2002; Ghosh et al., 2002; Hertel et al., 2003; Hars and Ou, 2002) and variable capturing incentives that are comparable with our motivation variables.

The respondents of the survey conducted by Bates et al. had to fill out their three top motivations to contribute in Open Source projects while Ghosh et al. enabled the developers to tick a maximum of four answers. Then the authors calculated the percentage scored by each incentive. In order to make our results comparable with the ones of these two studies, we use the percentage of firms that have chosen the score 5 (*I totally agree*) as a proxy of the percentage of firms that have chosen that item. Moreover Ghosh et al. distinguish between *reasons to join OS/FS community* and *reasons to stay in OS/FS community*. As we asked the operators for the motivations leading them to exploit Open Source software as an input, we take into account the latter reasons. Hars and Ou (2002) measure each motivation at a 7-point Likert scale ranging from *I strongly disagree* to *I strongly agree*. They report the percentage of agents choosing the score 7. We compare this figure with the percentage of the firms in our sample that choose the score 5.

Table 6 summarize the results of this meta analysis.

¹¹ Kolmogorov-Smirnov normality test states that the three variables are normally distributed (EM p value= 0.001, SM p value = 0.000, TM p value = 0.001).

Comparable items	Motivations	Motivation class	% of respondents top 3 motivations (Bates et al.)	% of respondents top 4 motivations (Ghosh et al.)	% of respondents score 7 (Hars and Ou)	% of respondents score 5 (our survey)
E	Limit the power of large software companies	E		28.9		
	Because we want to be independent of the price and licence policies of the large software companies	E			35	35
E	My contributions will enhance my reputation in the Free/Open Source software community	E	11			
	Get a reputation in OS/FS community	E		12		
	Because opening our source code allow to gain a reputation among our costumers and competitors				13.5	13.5
S	Participate in a new form of cooperation	S		37.2		
	Community identity	S			27.8	
	Because we want to place our source code and skills at disposal of the Free Software community and we hope that others do the same thing					24.8
S	Code should be Open	S	33.1			
	Think that software should not to be a proprietary good	S		37.9		
	Because we think that software should not to be a proprietary good					20
T	Get help in realizing a good idea for a software product	T		27		
	Because contributions and feedbacks from the Free Software community are very useful to fix bugs and improve our software					41.3
T	Improves skills	T	41.3			
	Learn and develop new skills	T		70.5		
	Human capital	T			88.3	
	Because we want to study the code written by other programmers and use it for developing new programs and solutions					20.9
T	Open Source reputation	T	11			
	For the reliability and quality of the Open Source software					38.3
T	My contribution creates specific functionality in the code needed for my work	T	33.8			
	For having products that are not available on the proprietary software market					16.5

Table 6: Meta-analysis of the motivations of firms and individual programmers.

Surprisingly economic motivations are not very different between individual programmers and firms. Both attach much importance to the emancipation from large software companies. Nevertheless the formulation chosen within the survey conducted by Bates et al. (*limit the power of large software companies*), might has been understood as referring to the fight against large software companies and intellectual property rights. As a consequence this item should be labelled as social and excluded from the comparison with our economic incentive.

Developer surveys show that gaining a reputation among peers does not rank among the main incentives of the Free Software programmers. This is at odds with the theoretical literature that numbers reputation among the most important motivations to contribute within Open Source

projects. Firms are not unlike the individual programmers. Only 13% of them *strongly agree* that exploiting Open Source software allows to gain a reputation among customers and competitors. Such reputation, in fact, is gained by opening pieces of code implementing smart software solutions. As a consequence a reputation gain is made possible by sharing a common communication protocol (the programming language) and of a set of computer science skills enabling them to recognize valuable solutions. Nevertheless most of the firm's customers are likely not to be advanced users that can read and value the source code of a program. Disclosing the code does not therefore enhance firm's reputation among them. At the same time, as we stated above, most of the firms in our sample supply services. Releasing the source code is not the most suitable mean to signal competences even to competitors.

Individual developers in all the four surveys attach more importance to social motivations than firms in our sample do. In particular, the item dealing with software property rights displays the higher percentage differences in the two groups. This corroborates the above analysed finding on the different nature (*intrinsic vs. extrinsic*) of social motivations in the two groups of agents.

Above 41% of our firms attach much importance to the feedbacks from the developers' community. Such feedbacks are instead numbered among the four top motivations only by 27% of the developers taking part in the Ghosh et al. survey.

A different outlook emerges taking into account the learning incentive. Individual developers value very much the chance to learn and improve skills through the participation within Open Source projects. Above 80% of the respondents to the Hars and Ou survey strongly agree that Free Software activities are suitable to develop human capital. Most of the firms in our sample do not regard learning as a crucial motivating factors. This is probably related to the relating low level of graduate education in the employees in our sample.

Reliability and quality of Free Software products represents an important incentive for more than 38% of the firms supplying Open Source solutions while only 11% of the developers (Bates et al. survey) list this incentive among the three most important ones. Quality and reliability of the solutions are crucial in the software market. This holds in a peculiar way for

Open Source products and services that have to overcome the network externality effects that push forward proprietary solutions.

According to several authors, this is a very important incentive also for individual developers that are often reported to denigrate the performances of proprietary programs. The data of our survey seem to deny this hypothesis. Nevertheless, most of the developers might have taken quality and reliability of the Free Software solutions for granted not including this incentive among their top three ones.

The *fill an unfilled market* motivation shows a different pattern. Such motivation ranks among the three top ones for 33.8% individual developers while only 16.5% of the firms in our sample strongly agree with this item. As we have pointed out above, this is probably due once again to the service oriented business models of the respondents to our survey. Focusing on services, such firms are likely not to need products performing peculiar tasks as individual developers do. Having access to the data gathered by Hertel et al. on 143 developers of the Linux kernel, we perform a direct comparison of their and our findings. This survey poses no measurement problem. The authors, in fact, measure the motivations to take part in Free Software movement through a five-point Likert scale. After having picked out the comparable items, we compute descriptive statistics and percentages of the scores for each incentive. Mann Whitney tests have been run in order to check for statistically significant differences in the mean values (Table 7 and Table 8).

The comparison highlights very well the heterogeneity in the motivations of firms and individual developers.

It is worth to notice that high score percentages and mean values are higher in the group of the individual developers. The firms in our sample, instead, display higher low score percentages. The set of the comparable items, in fact, does not include the main economic and technical incentives that, according to our survey, lay at the basis of firms' engagement in Open Source activities.

<i>Comparable items</i>	<i>Motivation class</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>Low score</i>	<i>High score</i>
Gaining reputation as an experienced programmer inside the Linux community	E	4.3	12.1	31.4	29.3	22.9	16.4	52.1
Personal exchange with other software developers	S	0.0	5.0	12.9	37.9	44.3	5.0	82.1
Code should be free	S	0.7	3.6	7.1	26.4	62.1	4.3	88.6
Improving programming skills	T	1.4	2.2	2.2	25.2	69.1	3.6	94.2
Because opening our source code allows to gain a reputation among our costumers and competitors	E	10.6	22.0	27.7	26.2	13.5	32.6	39.7
Because we want to place our source code and skills at disposal of the Free Software community and we hope that others do the same thing	S	6.4	7.8	17.0	27.7	41.1	14.2	68.8
Because we think that software should not to be a proprietary good	S	20.0	20.7	19.3	20.0	20.0	40.7	40.0
Because we want to study the code written by other programmers and use it for developing new programs and solutions	T	12.2	14.4	26.6	25.9	20.9	26.6	46.8

Table 7: Comparison of the motivations of firms and individual programmers: percentages of the scores (Hertel et al. data).

As we expected, developers assign higher scores to the items dealing with reputation gain, code reciprocation within the Free Software community, software freedom and learning. These two latter motivations show the higher differences in the percentages of high and low scores. Above 88% of the developers assign an high score to the items *code should be free* while only 40% of the firms do the same thing. Percentage of low scores are respectively 4.3% and 40.7%. The differences are not so large for the social motivation dealing with reciprocation that, as we have underlined above, firms regard as a crucial mean to sustain cooperation with the developers' community. Learning incentive show the largest differences both in high and low score (high score: 94.2% vs. 46.8%, low score: 26.6% vs. 3.6%). This is in accordance with the results of the meta-analysis.

<i>ID</i>	<i>Comparable items</i>	<i>Motivation class</i>	<i>Mean (our survey)</i>	<i>Mean (Hertel et al.)</i>
A**	Gaining reputation as an experienced programmer inside the Linux community	Economic		3.54
	Because opening our source code allow to gain a reputation among our costumers and competitors		3.1	
B*	Personal exchange with other software developers	Social		4.21
	Because we want to place our source code and skills at disposal of the Free Software community and we hope that others do the same thing		3.41	
C**	Code should be free	Social		4.46
	Because we think that software should not to be a proprietary good		2.99	
D**	Improving one's own programming skills	Technical		4.58
	Because we want to study the code written by other programmers and use it for developing new programs and solutions		3.29	

Table 8: Comparison of motivations of firms and individual programmers (Hertel et al. data)

*p value < 0.05, ** p value < 0.01.

Finally all the mean differences are statically different from zero in the two groups at a significance level of 5% or 1%. The item dealing with software property rights display the highest mean difference in the two groups.

5. Conclusions

For many theoretical reasons, the problem of motivations of participants to Open Source movement has attracted the interest of economists and sociologists (Bonaccorsi and Rossi, 2003). Most empirical analyses have addressed the motivational profile of individual programmers, while little empirical evidence is available on firms that base their business model on Open Source.

We find significant differences between the set of motivations of individuals and firms. In particular, firms emphasise economic and technological reasons for entering and contributing to the Open Source and do not subscribe to many social motivations that are typical of individual programmers. While one might expect these differences, it is interesting to observe that the more pragmatic motivational profiles of firms are accepted in the Free Software community, provided firms comply with the rules of the community. This means that the organisation of Open Source production is robust to a variety of motivations. Business motivations apparently do not destroy but reinforce the community.

References

- Allen R. A., Lucero M. A. (1994) *Employee benefit: a growing source of psychological contract violation*. *Human Resource Management*, 33, 425-446.
- Aoki A., Hayashi K., Kishida A., Nakakoji K., Nishinaka Y., Reeves B., Takashima A., Yamamoto Y. (2001) *A case study of the evolution of Jun: an object oriented Open Source 3D multimedia library*. Proceeding of the 23rd ICSE Conference Toronto, May 12-19.
- Bates J., Di Bona C., Lakhani K., Wolf B. (2002) *The Boston Consulting Group hacker survey*. <http://www.osdn.com/bcg/BCGHACKERSURVEY-0.73.pdf>, accessed on October 21st 2003.
- Bonaccorsi A., Rossi C. (2003) *Why Open Source software can succeed*. *Research Policy*, 32(7), 1243-1258.
- Bezroukov N. (1999) *Open Source software development as a special type of academic research (critique of the vulgar raymondism)*. http://www.firstmonday.dk/issues/issue4_10/bezroukov/, accessed on April 9th 2003.
- Bergquist M., Ljungberg J. (2001) *The power of gifts: organising social relationships in Open Source communities*. *European Journal of Information Systems*, 11 (4), 305-320.
- Cogenio (2001) *Indagine circa la propensione delle PMI industriali a migrare verso sistemi operativi alternativi: il caso Linux*. Report presented to "SMAU 2001", Milan, October 18 – 22.
- Dalle J. M., David P. A. (2003) *The allocation of software development resources in 'open source' production mode*. MIT Working Paper, <http://opensource.mit.edu/papers/dalledavid.pdf>, accessed on October 21st 2003.
- David K., Pfaff B. (1998) *Society and Open Source: why Open Source software is better for society than closed source software*. <http://www.msu.edu/user/pfaffben/writings/anp/oss-is-better.html>, accessed on 21st April 2003.
- David P. (1991) *Reputation and agency in the historical emergence of the institutions of Open Science*. Stanford University, CA, March, Mimeo.
- Diamond D., Torvalds L. (2002) *Just for fun: the story of an accidental revolutionary*. Harper Collins Publishers, New York, NY, USA.
- Feller J., Fitzgerald B. (2002) *Understanding Open Source software development*. Addison Wesley, Boston, MA, USA:
- Fink M. (2003) *The business and economics of Linux and Open Source*. Prentice Hall, Upper Saddle River, NJ
- Franck E., Jungwirth C. (2002) *Reconciling investors and donators -The governance structure of Open Source*. MIT Working paper, <http://www.ifbf.unizh.ch/orga/downloads/EGOS2002/FranckJungwirth.pdf>, accessed on 21st October 2003.
- Franke N., von Hippel E. (2003) *Satisfying heterogeneous user needs via innovation toolkits: The case of apache security software*. *Research Policy*, 32(7), 1199-1215.
- Friedman H. H., Herskovitz P. J., Pollack S. (1993) *The biasing effects of scale-checking styles on response to a Likert scale*. Proceedings of survey research methods section, American Statistical Association, <http://www.amstat.org/sections/srms/Proceedings/y1993.html>, accessed on 9th May 2003.
- Ghosh R. A., Glott R., Krieger B., Robles G. (2002) *Survey of developers*. Free/Libre and Open Source Software: Survey and Study, FLOSS, Final Report, International Institute of Infonomics,

- Berlecom Research GmbH, http://floss.infonomics.nl/report/FLOSS_Final4.pdf, accessed on October 21st 2003.
- Gill L. (2002) *Does Open Source software really work?* <http://www.newsfactor.com/perl/story/16923.html>, accessed on April 25th 2003.
- Green L. G. (2000). *Economics of Open Source software.* <http://badtux.org/home/eric/editorial/economics.php>, accessed on April 19th 2003.
- Hars A., Ou S. (2002) *Working for free? Motivations for participating in Open Source projects.* International Journal of Electronic Commerce, 6, 25-39.
- Harvey J. (1999) *Evaluation cookbook. Web version.* <http://www.icbl.hw.ac.uk/ltidi>, accessed on 9th May 2003.
- Hawkins R. E. (2002) *The economics of the Open Source Software for a competitive firm.* MIT Working Paper, <http://econpapers.hhs.se/paper/scescecf1/234.htm>, accessed on October 21st 2003.
- Hecker F. (2000) *Setting up shop: the business of Open-Source software.* <http://www.hecker.org/writings/setting-up-shop.html>, accessed on March 26th 2003.
- Hertel G., Niedner S., Hermann S. (2003). *Motivation of software developers in the Open Source projects: an Internet-based survey of contributors to the Linux kernel.* Research Policy, 32(7), 1159-1177.
- Kollock P. (1999) *The economics of on line cooperation: gift and public goods in cyberspace.* In Smith M., Kollock P.(ed.) *Communities in Ciberspace*,220-246. Routledge, London, UK.
- Kratz M. S., Robinson S. L., Rousseau D. M. (1994) *Changing obligations and the psychological contract: a longitudinal study.* Academy of Management Journal, 37, 137-152.
- Kuster B., Osterloh M., Rota S. (2002) *Trust and commerce in Open Source – a contradiction?* MIT Working Paper, http://www.wiwiss.fu-berlin.de/w3/w3sydow/EURAM/pdf_2002/EURAM%20Paper%20Rota%20et%20al.pdf, accessed on 21st October 2003.
- von Krogh G., Spaeth S., Lakhani K. (2003) *Community, joining, and specialization in Open Source software innovation: a case study.* Research Policy, 32(7), 1217-1241.
- Lakhani K., von Hippel E. (2003). *How Open Source software works: “free” user-to-user assistance”.* Research Policy, 32(6), 923-943.
- Lerner J. (2002) *Incentives and Open Source software.* <http://cip.umd.edu/lerner.htm>, accessed on April 25th 2003.
- Lerner J., Tirole J. (2001) *The Open Source movement: key research questions.* European Economic Review, 45, 819-826.
- Lerner J., Tirole J. (2002a) *Some simple economics of the Open Source.* The Journal of Industrial Economics, 2(L), 197-234.
- Lerner J., Tirole J. (2002b) *The Scope of Open Source licensing.* MIT Working Paper, <http://opensource.mit.edu/papers/lernertirole2.pdf>, accessed on August 2nd, 2003.
- Levison H., Price C., Munden K., Mandl H., Solley C. (1962) *Men, management and mental health.* Harvard University Press, Cambridge MA, USA.
- Liebowitch E. (1999) *The Business case for Linux.* IEEE Software 16(1), 40-44.
- Mauss M. (1959) *The gift. The form and the reason for exchange in archaic societies.* Routledge, London, UK.

- McLean Parks J., Schmedemann D. A. (1994) *When promises become contracts: implied contract and handbook provisions on job security*. Human Resource Management, 33, 403-423.
- Mockus A., Fielding R., Herbsleb J. (2000) *A case study of Open Source software development: the Apache server*. In Proceedings of the 22nd International Conference on Software Engineering (ICSE'00), Limerick, May IEEE Press, June 4-11, 263-272.
- Monroe K. R. (1996) *The heart of altruism*. Princeton University Press Princeton, New Jersey, NJ, USA.
- Morrison E. W., Robinson S. L. (1997) *When employees feel betray: a model of how psychological contract violation develops*. The Academy of Management Review, 22(1), 226-256.
- National Advisory Council on Innovation Open Software Working Group (2002) *Open Software and open standards in South Africa. A critical issue for addressing the Digital Divide*. <http://www.naci.org.za/docs/opensource.pdf>, accessed on 6th May 2003.
- Osterloh M., Frey B. (2002) *Does pay for performance really motivate employees?* In Andy Neely (ed.), Business Performance Measurement, 107-122. CUP, Cambridge, UK.
- Osterloh M., Rota S., Von Wartburg M. (2002) *Open source – new rules in software development*. <http://www.unizh.ch/ifbf/orga/downloads/opensourceaom.pdf> , accessed on April 25th 2003.
- Pavlicek R. (1999a) *Already a contender*. <http://archive.infoworld.com/articles/op/xml/01/10/29/011029opsourcexml>, accessed on April 25th 2003.
- Pavlicek R. (1999b) *Keys to effective Linux advocacy within your organisation*. <http://users.erols.com/plavlicek/oreilly/als-fullpaper-1999.txt>, accessed on April 25th 2003.
- Raymond E. S. (2000) *Software Release Practice HOWTO*. <http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/pdf/Software-Release-Practice-HOWTO.pdf>, accessed on June 7th 2003.
- Raymond E. S. (2001a) *The cathedral & the bazaar. Musings on Linux and Open Source by an accidental revolutionary*. O'Reilly & Associates, Sebastopolous , CA, USA.
- Raymond, E. S.(2001b) *Homesteading the Noosphere*. In Raymond E.S., The Cathedral and the Bazaar: musing Linux and Open Source by an accidental revolutionary. O' Really, Sebastapol, CA, 19-64.
- Rousseau D. M. (1989) *Psychological and implied contracts in organisations*. Employee Responsibility and Rights Journal, 2, 121-139.
- Rousseau D., McLean Parks J. (1993) *The contracts of individuals and organizations*. Research in Organizational Behaviour, 15, 1-47.
- Schimitz P. E. (2001) *Study into the use of Open Source software in the public sector*. An IDA Study (Interchange of Data between Administrations), <http://www.cio.gv.at/alternativen/ida-oss/IDA2-OSS-usage-report.pdf>, accessed on July 7th 2003.
- Schmidt F. L. (1992) *What do data really mean? Research findings, meta-analysis, and cumulative knowledge in psychology*. American Psychologist, 47, 242-248.
- Simon, B., Loewy, M., Stürmer, S., Weber, U., Freytag, P., Habig, C., Kampmeier, C., Spahlinger, P. (1998) *Collective identification and social movement participation*. Journal of Personality and Social Psychology, 74, 646-658.

- Stallman R. (1984) *The GNU Manifesto*. <http://www.gnu.org/gnu/manifesto.html>, accessed on March 26th 2003.
- Stone M. (1999) *Science of the New Renaissance*. In Di Bona C., Ockman S., Stone M., (eds.) Opensources. Voci dalla Rivoluzione Open Source. Apogeo, Milan, I.
- Torvalds L. (1999) *Il vantaggio di Linux*. In Di Bona C., Ockman S., Stone M., (eds.) Opensources. Voci dalla Rivoluzione Open Source. Apogeo, Milan, I.
- Ullman E. (1998) *The dumbing down of programming*. http://archive.salon.com/21st/feature/1998/05/cov_12feature.html, accessed on April 19th 2003.
- von Hippel (2002) *Open Source projects as horizontal innovation networks - by and for users*. SSRN Working paper, http://papers.ssrn.com/sol3/delivery.cfm/SSRN_ID328900_code020906590.pdf?abstractid=328900, accessed on September 24th 2003.
- Wachter K. W., Straf M. L. (1990) *The future of meta-analysis*. Russell Sage Foundation, New York, NY, USA.
- Wichmann T. (2002a) *Firms' Open Source activities: motivations and policy implications*. Free/Libre and Open Source Software: Survey and Study, FLOSS Final Report, International Institute of Infonomics, Berlecom Research GmbH, <http://floss.infonomics.nl/report/index.htm>, accessed on September 24th 2003.
- Wichmann T. (2002b) *Use of Open Source Software in Firms and Public Institutions. Evidence from Germany, Sweden and UK*. Free/Libre and Open Source Software: Survey and Study, FLOSS Final Report, International Institute of Infonomics, Berlecom Research GmbH, <http://floss.infonomics.nl/report/index.htm>, accessed on September 24th 2003.
- SPSS Inc. (2001) *SPSS base 11.0 guida dell'utente*. SPSS Inc. Editor, IR.