Software Freedom, Open Software and the Participant's Motivation

A Multidisciplinary Study

Author: Menelaos Tzouris

Supervisor: Professor Claudio Ciborra

Abstract

The success of Free/Open Source Software has demonstrated the viability and often the dominance of an alternative – almost paradoxical – form of software development. This innovative organisational formation and operation has, recently, been studied intensively. The current paper deals with the issue of participant's motivation to join and then to operate in a free/open source software development process. However, it does not adopt a single perspective towards reaching a definite answer. On the contrary, building on accumulative research work, it argues that no single perspective is adequate to explain free/open source software participants' motivation, since each perspective provides a rather limited understanding of the issue.

The method that this paper uses in order to reach the aforementioned conclusion is twofold. First a literature review is considered; articles relating to free/open source software participation from specialised academic sources have been examined. Following the literature review, the four most popular theories are employed and compared in order to reveal their strengths and weaknesses, as well as their conceptual intertwine. Secondly, data has been gathered from empirical research targeted to free/open source software contributors. The findings of this research are contrasted to the literature. The conclusion that this study reaches is that participants' motivation is multifaceted and hence can only be adequately explained, if an amalgamation of perspectives is being employed.

Keywords

Free Software; Open Source Software; Motivation

Acknowledgements

This research would not have been completed – maybe not even started – if it had not been for those people that originally inspired me and consequently helped me to find my way thought the maze formed by Free/Open Source Software and Social Science. I truly feel the need to thank them all, and acknowledge their help.

I start by saying "Thank you" to my dissertation's supervisor, Professor Claudio Ciborra, whose valuable advise and comments encouraged and enlightened me, at the various stages of my dissertation. I would, also, like to thank Richard Stallman for his very important comments and his help towards my better understanding of Free Software and Open Source. I feel both lucky and delighted to have the opportunity to speak to Richard, the undisputable thrill in Free Software and an idol to many FS/OSS contributors. A great "Thank You" also goes to Philip Hands, for his help and especially for his patience, who helped me broaden my perspective with his wise words. I would like to thank Kasper Edwards for his important help concerning my understanding of Epistemic Communities as a means to conceive Open Source. I also want to thank PhD students Federico Iannacci, Maha Shaikh, Prodromos Tsiavos (in alphabetical order) for their help especially in the early stages of my research. Last but not least, a great "thank you", goes to all FS/OSS contributors – named or anonymous – that volunteered to be the subject of my empirical research as well as to those that didn't respond to my request, but provided their comments to my methodology and questionnaire content. Being a "newbie" can be an intimidating business; and thanks to all the people mentioned above it went on smoothly.

As this research reaches its end, I feel the need to acknowledge the great psychological help provided by my faithful Lara, who kept me company at the long hours of writing this paper. She may not be able to understand how much her support meant to me, but I know she will always appreciate a gentle tap on the head. Last and most of all I want to say a wholehearted "Thank You!" to my family whose love and support is constantly my utmost personal motivation towards pursuing my aspirations.

Dedication

To my Family.

Table of Contents

Verbatim

When I asked the white secretary of the reserve who was the richest man, he mentioned a man none of the Indians had mentioned – that is, the man who had on the books the most stock, the most cattle and horses. When I came back to my Indian informants and asked them about Jimmy McHugh, about all his horses, they shrugged with contempt. "He keeps it", they said, and as a consequence, they hadn't even thought to regard him as wealthy.

White-Headed Chief was 'wealthy', even though he owned nothing. In what way did virtue pay? The men who were formally generous in this way were the most admired, most respected, and the most loved men in the tribe. These were the men who benefited the tribe, the men they could be proud of, who warmed their hearts.

(Maslow, quoted by Hoffman 1988, p.121)

Chapter 1: Introduction

1.1 Prologue

If this were a paper on Software Engineering[1] it would attempt to describe a rational way of software development, just like renowned papers did in the past. But it is not! Software Engineering, by and large, is a discipline that tries to put an order to the chaos of software development. It does so by suggesting solutions to known problems, however, on the hindsight. Lately, many of us have been witnessing, with great interest, the rise of a new phenomenon in software development known as Free Software (FS) or Open Source Software (OSS). Its publicity and alleged success – in terms of product quality, reliability and competitiveness – stimulates scholars for the first time to inquire this very success, rather than trying to articulate ways that would help to deter another failure.

Free/Open Source software is not different than any other kind of software, not technically at least. However it challenges many of the well-established ideas in Software Engineering. Eric S. Raymond articulates this by introducing the concepts of the "cathedral" and the "bazaar" (Raymond 1999). He associates the corporate, mainstream, closed-sourced method to the "cathedral" model, and the open source development process to the "bazaar" model. An appealing example of the central differences between the two methods, is illustrated by Brooks' Law (1975). This famous law has held for more than 25 years already. However it ceases to be applicable in the FS/OSS context. Brooks' Law suggests that "Since software construction is inherently a systems effort – an exercise in complex interrelationships – communication effort is great, and it quickly dominates the decrease in individual task time brought about by partitioning. Adding more men then lengthens, not shortens, the schedule." Raymond (1999) explains that the open source model, renders this law obsolete, through customary central version control, mutual respect, and an

---

[1] "Software Engineering. 1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. 2) The study of approaches as in 1)" IEEE (1990).

army of developers and bug testers. As he puts it informally, in what he calls the "Linus's Law": "Given enough eyeballs, all bugs are shallow".

Open Source and Free Software is a phenomenon that is growing not only in impetus and membership, but also in market share. Some well-known FS/OSS programs and systems include GNU (Stallman 1999), Linux (Torvalds 1999, Torvalds and Diamond 2001), the GNU/Linux[2] system (Moody 2001, Raymond 1999, Stallman 1999), Mozilla (Hamerly*, et al.* 1999, Hecker 1999), and Apache (Behlendorf 1999, Mockus*, et al.* 2000) – for a large catalogue of FS/OSS programs, see http://www.gnu.org/directory/all/[3]. This phenomenon has already touched the lives of all Internet users, since more than 65% of the Internet's active sites run the Apache Web server (Netcraft 2002), and GNU/Linux is the fastest growing operating system in the servers market (Feller and Fitzgerald 2002). Hence, software developed and produced in free/open source software projects has become an important competitor in the software industry.

The open development model is destined to allow talented professionals around the globe to use their competence to the maximum. While not necessarily organized by multinational corporations, the open development model nonetheless affects their activities (Metiu and Kogut 2001). Companies that make money out of open-source software are increasing (Ljungberg 2000, Tiemann 1999), and that makes free/open source software a phenomenon worthy of research and attention. Traditionally, software companies like Microsoft have hoarded their intellectual investment in

---

[2] Linus Torvalds created the first version of the Linux kernel in 1991, however the higher-level functionality in a "Linux" distribution is based on the GNU system, which was launched by Richard Stallman in 1984. The higher-level functionality in today's "Linux" distributions continues to be based on the GNU system. The term GNU/Linux refers to the broad category of UNIX-like operating systems that are built on the combination of GNU and Linux.

[3] The GNU Free Software Directory is a project of the Free Software Foundation. It catalogues useful free software that runs under free operating systems — particularly the GNU operating system and its GNU/Linux variants.

source code, and sold only closed binary versions of their applications. Now they feel threatened by the "new? development methodology" (Halloween_Document 1998)[4].

Free/open source software, completely defies conventional logic and challenges this model by giving the source code away. The concept of open source/free software is not necessarily linked to free in terms of price – it rather refers to the lack of proprietary restrictions on ideas underlying its source code. It is "free" in the sense that it gives freedom towards using, modifying, and distributing the software; and it is "open" in the sense that it gives openness towards examination and alteration of the source code.

Open source can be seen as a movement, where communities of highly skilled programmers collectively develop software (Ljungberg 2000). "The ability of the OSS process to collect and harness the collective IQ of thousands of individuals across the Internet is simply amazing" (Halloween_Document 1998).

1.2 History of Free/Open Source Software

Before proceeding into the details of this research, it is interesting to briefly explore the history of Free/Open Software, as a means to set the issue in context and to provide a better understanding of its essence (the interested reader, can also see Appendix I – Open Source Timeline).

The concept of Free Software is as old as the history of computing. Its origins can be traced back to the 50s and 60s when the software was sold together with the hardware, and macros and utilities were freely exchanged in certain user forums. In the 80s, as software was increasingly commercialised, Richard Stallman, then a researcher at MIT founded the Free Software Foundation (FSF) that provided a conceptual

---

[4] In the last week of October 1998, a confidential Microsoft memorandum on Redmond's strategy against Linux and Open Source software was leaked to Eric Raymond. Microsoft was later forced to acknowledge its authenticity.

foundation for free software. Stallman's (1985) "GNU Manifesto" is the basis of today's Free Software Movement[5].

The term Open Source is newer than Free Software. It was coined in 1998, following the Mozilla release to describe software distributed in source under licenses guaranteeing anybody rights to freely use, modify, and redistribute, the code, and is now a buzzword. It is also a term that has gained currency as a way to describe the tradition of open standards, shared source code, and collaborative development behind software (O'Reilly 1999). The intention, behind the introduction of this new term in our dictionary, was to sell the hackers'[6] ways of doing software to industry and the mainstream by avoiding the negative connotations – to businesses – of the term 'free software'" (Raymond 1996).

1.3 "Free Software" or "Open Source Software"?

Free Software, refers to software that is "freely modifiable and redistributable". That is software which source code can be freely modified and redistributed. Free Software is also referred to as Freed Software, Liberated Software (software libre) or FRS (freely redistributable software). However there seems to be an ambiguity problem. Free Software does not necessarily mean zero-cost software. It rather means software that is free as in "free speech," not "free beer". That is, free software may or may not be distributed with a monetary cost, but the source code should be freely available in order to empower future innovation. In Stallman's words, whether it has an economic advantage or not, free software has a "social advantage, allowing users to cooperate, and an ethical advantage, respecting the user's freedom" (Stallman 1999).

Open Source is a marketing name for Free Software, coined in February 1998 as an attempt to overcome the confusion over the word "free" in the English language. It can be viewed as a marketing term for Free Software because "free" was worrying

---

[5] The early days of open source is illustrated in Raymond, E. S. (1999a) and Stallman, R. (1999); also see Appendix I.

[6] The use of "hacker" to mean "security breaker" is a confusion on the part of the mass media. We hackers refuse to recognise that meaning, and continue using the word to mean, "Someone who loves to program and enjoys being clever about it" Stallman, R. (1999).

and confusing to business. Indeed, according to its proponents, it seemed to have succeeded its purpose: "It has almost completely turned around the negative image that 'free software' had outside the hacker community." (OSI 1998)

Open Source and Free Software refer to, originally the same set of software – at least until February 1998. Now they are different but largely similar. To the novice's surprise, despite the similarities there is a considerable extend of dispute, between the proponents of Free Software Movement and Open Source Movement, that originates from diverse political philosophies[7] and extends to the issue of the most appropriate name. For the purpose of this study, however, the distinction is not important. Consequently the terms Free Software and Open Source Software will be treated as synonyms and whenever possible the composite term "Free/Open Source Software" or "FS/OSS" will be used instead.

1.4 Paper Overview

The remainder of this study unfolds as follows: In next chapter, that is chapter two, the research question as well as the main argument that this paper attempts to make, are presented and set into the context of the broader research in the field of FS/OSS. Chapter three constitutes the fundamental step before discussing the literature review; in effect it explores the prevailing research trends in the issue of FS/OSS participants' motivation. Consequently, in chapter four, the four most popular theories – as identified by the literature research in the previous chapter – are presented. Chapter five, constitutes an introduction to the empirical research that was conducted for the purposed of this paper; in particular the research epistemology and the research method are discussed. Then, chapter six illuminates the empirical findings along with the critical analysis of the literature. The intertwined nature of the literature is also discussed in this chapter. Chapter seven, claims that the intertwined nature of the literature should not come as a surprise, on the contrary it must be viewed as part of

---

[7] "The Open Source movement says that proprietary software is a suboptimal solution. They urge the developers to make the software open-source but do not insist. They focus on the development model. The Free Software movement says that proprietary software is a problem – a social and ethical problem – and replacing it with free software is the solution. […] Whether this is a more efficient system of development is a secondary question in our view" (Richard Stallman in an email correspondence with the author).

the process of scientific discovery that is currently taking place with regards to FS/OSS. In chapter eight a review of the discussion made in this the paper is made and the conclusions are being drawn. Finally, in chapter nine, the research limitations are discussed along with suggestions for fields suitable for further research.

Chapter 2: Research Question

Nowadays, both technologies and organisations are undergoing dramatic changes in form and in function. New and unprecedented forms and functions are becoming evident (Orlikowski 2000). Through the rapid introduction of the Internet, the potential for non face-to-face, lateral written communication that take place across time and space has been greatly facilitated (Hinds and Kiesler 1995) bringing together widely dispersed actors in temporary and shifting networks (Kallinikos 2001). So far, the implications of this novel means of communication are evident in software development and academic research environments, such as the genome project or nuclear physics. The Free Software/Open Source phenomenon is a prominent illustration that is taking place in the Software Industry.

The success and popularity of the Free /Open Source Software phenomenon yields the question of whether this type of organising can be applied to other activities aside from software development. This hypothesis follows the intuitive idea that knowledge intensive production can be potentially "freed", "liberalised", or "opened". According to Michael Porter's strategic analysis, the importance of information-processing components of both products and "value activities" is gaining importance, as opposed to their physical components (Porter 2001, Porter and Millar 1985). This idea renders the aforementioned hypothesis more pervasive. Hence, the expectation, that traditional Industries might "open" some of their knowledge intensive processes in the future, increases in probability. It is now exceptionally promising to expand contemporary research on the Free/Software Open Source phenomenon, into more traditional organisations (i.e. IT support, law soliciting, consulting etc.). The question thus, is to what extent this is feasible.

This broad research question constituted the trigger for the current paper. In order to be answered, it requires analysis of contributors' motivation, and analysis of industrial and organisational cultures as well as widespread observations and empirical research – see chapter "Further Research" for details – hence, it is too extensive to be addressed in an MSc dissertation. The current work will, therefore, attempt to address

only the first part of this complex question. That is, contributors' motivations that underlie their participation in Free/Open Source Software.

The literature demonstrates two kind of Free/Open Source Software contributors (Dempsey*, et al.* 2002, Moody 2001, Torvalds and Diamond 2001); first, those that contribute actual code to the project, and second, those that participate in discussions, report bugs and in general perform other activities aside from programming. Moreover one can distinguish between micro level of contributors, that is, individuals and the macro level of contributors, that is, often profit organisation (Feller and Fitzgerald 2002). As a means to keep this study narrowed, the current research will focus on individual contributors of source code.

Moreover, this research will draw upon accumulative work on the new organisational form that is dictated by FS/OSS. These organisational form is characterised by a loose community connections where there are no strict and explicit economic incentives at work; rather there are symbolic ones (Ciborra and Andreu 2001). Indeed this very ambiguity that underlies contributors' incentives, offers a wealth of different analyses that can be used towards their exploration.

This study claims that none of the analyses in the literature explains the motivational issue in its full. Each employs a limited perspective that offers rather a partial understanding of the phenomenon and can provide only limited enlightenment on the motivational issues and group dynamics of FS/OSS. Thus, a combination of perspectives is imperative in order to understand the phenomenon.

Following this analysis the current work will argue upon a composite way of understanding motivations. Namely, community structure, contributors' motivations (often called non-conventional economics), licences as a means of code and ultimately governance, and conventional economics. These four issues have been addressed in the literature quite extensively already, however not in relation to each other. Hence, the objective of the current study is to demonstrate the intertwined nature of the aforementioned theories and provide the basis of a new conceptual understanding of FS/OSS contribution.

Chapter 3: Literature Research

An attempt to address the issue of FS/OSS participants' motivation using concepts from the traditional software engineering literature, for example the assumption that programmers involved in FS/OSS software development perform 'egoless programming' (Weinberg 1971), or that they contribute as a result of altruistic motivations are very simplistic and seem not to suffice (Lancashire 2001). Free/open software development is a rising phenomenon that ought to be studied by numerous disciplines and from various perspectives (Feller and Fitzgerald 2000). The current study embraces this stance. Thus, it is essential to explore the extent of attention that the various approaches have been given in the literature.

In an effort to pinpoint the most promising theories that contemporary studies adopt, the author researched the literature. The resources upon which this research drew were articles available on the Electronic Journals of the British Library of Political & Economic Science[8], as well as on the Free/Open Source Research Community[9], and First Monday[10]. In particular, the research was focused on articles related to free/open source software development, as well as, to volunteer or distributed work and participants' motivation. Its findings are presented below.

3.1 Findings

It was found that the work conducted by scholars who investigate what motivates people to put effort in FS/OSS development, as well as in, volunteer or distributed work perceives the issue from multiple perspectives. Though, it must be explicitly stated, that more often than not, scholars do not take a single stance in order to address the issue. On the contrary they are using more than one perspective to understand the phenomenon. However, the purpose of this chapter is not to provide an in depth literature review, rather it aims to provide awareness upon the trends of the

---

[8]Electronic Journals considered are available to LSE staff and students (http://www.lse.ac.uk/library/).

[9]FS/OSS researchers can post their papers on open source and free software, so that the community can become steadily larger and more comprehensive (http://opensource.mit.edu).

[10]First Monday is one of the first peer-reviewed journals on the Internet, solely devoted to the Internet (http://www.firstmonday.dk).

contemporary research in the field of FS/OSS. For presentation clarification, the individual papers of the studied literature are classified according to their primary stance, in Appendix III.

In essence, it was found that out of the thirty-six (36) articles, that were considered overall, 33.3%, that is 12 articles, are using theory of communities, 27.7%, that is 10 articles, are using motivational psychology, 13.8%, that is 5 articles, are using governance structure, 13.8%, that is 5 articles, are using economics, and 11%, that is 4 articles, are using gift economy or some other theory.

Chapter 4: Literature review

In what follows, the most widely accepted theories – based on the aforementioned research findings – are presented. Namely, community norms and structures, theory of motivations, licences as a means of code and ultimately governance, and economics**.**

4.1 Communities

Ciborra and Andreu (2001) argue that the main characteristics of the situation illustrated by the development of the Linux kernel example are: First, that the final product is a piece of very explicit (codified) knowledge. Second, the very same structure of the product can be used as a guide for the development process. Third, meta-knowledge can be spelt out in a similar language that is used to describe the product and transferred by using the same medium, i.e. the Internet. Fourth, the sense of community, the values underlying it, the rewards and the behaviour codes among the members of the group of software developers. They continue by stressing that the three first characteristics are intertwined with the fourth. That is, the cultural context needs to be present when the three first conditions hold and probably it is not adequate by itself to generate the same results. In other words, they assert the importance of the community structure in the process of building software in the FS/OSS context.

According to Marshall (1998), the concept of community concerns a particularly constituted set of social relationships based on something which the participants have in common – usually a common sense of identity; furthermore, the term community traditionally has been linked with people who share a common interest, as well as with, a strong geographical referent. Computer Mediated Communication technologies offered an alternative to constraints of geography and time associated with face-to-face meetings or telephone conversations (Zuboff 1988). Thus, the development of information technologies and telecommunications created new and cohesive social structures in cyberspace (Castells 1996) that resembles communities. Place is becoming less relevant and is gradually being replaced by networks (Blakely 1989), forming what Rheingold (1994) calls "virtual communities".

Free/open source software contributors comprise communities that have many characteristics in common with *Communities of Practice*. This has been claimed in the

literature either explicitly (Franke and Shah 2001, Moon and Sproull 2000, Neus 2001, Scacchi 2002, Tuomi 2001) or implicitly (Faraj and Wasko 2001, Madanmohan and Navelkar 2002, Orlikowski 2002, Zhang and Storck 2001). Communities of practice are groups of people informally bound together by shared expertise and interest. A community of practice is an emergent social collective where individuals working on similar problems self-organise to help each other and share perspectives about their work practice (Brown & Duguid, 1991; Wenger, 1998). The term community of practice is defined as the participation in an activity system about which participants share a common identity and motivation (Lave and Wenger 1991). These communities are called by some researchers Electronic Communities of Practice (Wasko and Faraj 2000), Virtual Communities of Practice (Neus 2001) or Networks of Practice (Faraj and Wasko 2001) and they are claimed to provide a valuable insight into the issue of motivation for FS/OSS contributors. However as it has been argued by Tzouris (2002), the original conceptualisation of communities of practice prevent the theory from being applied in the distributed/networked context and thus it will not be discussed further herein.

Another promising perspective to explain FSS/OS motivation, from a community's point of view, is the theory of *epistemic communities*. In the effort to explain the particulars of free/open source software, authors identify important resemblances to scientific communities (Bergquist and Ljungberg 2001, Bezroukov 1999, Ciborra and Andreu 2001, Dalle and Jullien 2001, Raymond 2000). A comprehensive analysis of free/open software as epistemic communities is made by Edwards (2001).

This approach sheds some light on how it is possible to develop free/open software, as well as on motivational issues. Inspired by (Haas 1992), Edwards stresses that free/open software communities are epistemic communities, where actors are motivated by influence and reputation. In Raymond's words "the reputation game may be critical in providing a social context within which the joy of hacking can in fact become the individual's primary motive". The primary motivation for people to enter the development process is a personal need for particular software functionality. Furthermore, in the framework of Haas (1992), "epistemic communities are largely motivated by the existence of a power vacuum that has come about in the face of uncertainty" (in Edwards 2001).

The four characteristics of free/open software epistemic communities are:

- Shared normative and principled beliefs: refers to the shared understanding of the value-based rationale for contributing to the software.

- Shared causal beliefs: refers to the shared causal understanding or the reward structures. Therefore, shared causal beliefs have a coordinating effect on the development process.

- Shared notions of validity: refers to contributors' consensus that the adopted solution is a valid solution for the problem at hand.

- Common policy enterprise: refers to a common goal that can be achieved through contributing code to the software.

In simple words, there is a mutual understanding, a common frame of reference of what to develop and how to do it.

4.2 Motivational Psychology

Understanding the physiological motivation of participants in free/open source software is a crucial step towards understanding why people participate in it. The academia has a lot of work to demonstrate, towards the answer of this question, from the perspective motivational psychology.

One of the first scholars who talked about what motivates people was psychologist Abraham Maslow. He identified a hierarchy of human needs, which posits the topmost category of *self-actualisation* (Maslow 1970). Only when our basic needs of food, shelter and security, for example are met do we become concerned with self-actualisation. Maslow's theory has been drawn upon to help explain the motivation behind the committed contribution of OSS developers by Raymond (2000). As Raymond puts it, "the joy of hacking is a self-actualisation or transcendence need which will not be consistently expressed until lower-level needs (including those for physical security and for 'belongingness' or peer esteem) have been at least minimally satisfied" (Raymond 2000).

Another very important distinction of motivations is between intrinsic and extrinsic. *Intrinsic motivations* include the desire of feeling competence and self-determination;

whereas *extrinsic motivations* include factors such as direct or indirect monetary compensation, and other's recognition as well (Deci 1975). Building upon this distinction, as well as other important psychological literature on motivations – in particular Herzberg (1982) and Klandermans (1997) – Hars and Ou (2001) analyse the psychology of free/open software contributors and they distinguish the motivation factors between "internal factors" and "external rewards". Following a survey, which was targeted to free/open software programmers, they conclude that, external rewards have greater weight than internal factors.

Other related research studies stress that many people contribute to free/open source software without expecting to ever receive any individual reward (Franck and Jungwirth 2002). In this case, since it is not monetary rewards – as in commercial software production – and not peer recognition or a notable talent indication, it seems appropriate to talk about *idealistic motivations*. However even if we accept as an empirical phenomenon that people donate due to idealistic motivations, the question that still matters is, which kind of organisations they choose for their contributions (Franck and Jungwirth 2002). The theory of non-profit organisations by (Hansmann 1980) makes a strong point to support this view by proposing a major theme of economic theories of nonprofits and the dominant rationale.

Following the previous analysis, Franck and Jungwirth (2002) distinguish between two classes of actors in free/open source software development. That is, *investors and donators*. The former group is characterised by the fact that although no wages are paid to contributors, other pay-offs may turn their effort into a profitable investment. The latter group is characterised by the fact that they contribute to free/open source software without expecting to ever receive any individual rewards.

4.3 Governance

A similar concept to opens source software, which is often confused as a synonym, is public domain software. It is often assumed that public domain software is free software, but free/open source software proponents disagree with this view. Public domain software is software that has been released without any copyright (or more accurately, with an explicit rejection of the author's copyright). This gives to the users complete freedom upon that code. As a result the users may then convert the program

into proprietary software by re-copyrighting the software, and redistribute it in this form. This action will strip away the freedom originally attached to the program.

Free/Open Source software proponents, such as the original advocate of free software, Richard Stallman, argue on this basis that public domain software does less service to the community than Free/Open Source software. Free/Open Source software is distributed with a license that, in simple terms, prevents it from being made proprietary in the future. As a result, it is arguably "more free" than public domain software. "The central idea of copyleft is that we give everyone permission to run the program, copy the program, modify the program and distribute modified versions - but not permission to add restrictions of their own. Thus the crucial freedoms that define 'free software' are guaranteed to everyone who has a copy; they become inalienable rights." (Stallman 1999).

In accordance to this analysis, some scholars argue that the essence of free/open source software is their governance mechanism which derives from a specific license agreement for software. This mechanism forms not only contributor interaction and product alteration and development, but also the incentives that underpin programmers' participation. The poor success of firms, which disclose the source code of their commercial software in an attempt to tap improvements by direct involvement, is unsurprising in this sense. They fail to impose to their software the governance that will lure participation (Dalle and Jullien 2001). The most prominent example is the attempt of Netscape in 1998 to open the source code of Netscape Communicator Standard Edition – otherwise known by its original code name, Mozilla – making a free software out of it. Netscape offered licence conditions that contributors were reluctant to accept in the first place (Hecker 1999, Lerner and Tirole 2001).

The ways that licenses together with code provide a governance framework is explored by Syme and Camp (2001). They elaborate their argument, even further, to explore how different licenses combined with code provide a range of visions for governance of the information society. They do this, by building upon the pioneering work of Stallman and its popularisation by Lessig (1999). Syme and Camp, discuss the governance implications of the elements of open code and they present the view

that governance consists of code plus a licensing framework. Their work offers distinct views of the network society drawn from examinations of the various forms of governance currently applied to code; namely, open code licensing[11], public domain code, proprietary licenses, and the Uniform Computer Information Transaction Act (UCITA). They state that open code proponents are seeking, in essence, a highly libertarian form of governance, in particular a government of code without the power to enforce or obligate the user to take action. They close by arguing that different visions of society are implicit in different licensing mechanisms for code. As they put it: "if code is law the rules governing the creation of code become rules governing the creation of law" (Syme and Camp 2001).

4.4 Economics

This section presents the use of economic theory, as a means to shed some light on the economic incentives of Free/Open Software contribution. Indeed, this is an issue that seems to be an important stimulus of numerous research work, and it is often articulated implicitly or explicitly in most of the Free/Open Software related reports (see for example: Feller and Fitzgerald 2002, Hars and Ou 2001, Lerner and Tirole 2001, Raymond 1999, Torvalds and Diamond 2001). The economics of FS/OSS can be studied from two different perspectives, that is, the micro level of individuals and the macro level of the broad organisation/community (Feller and Fitzgerald 2002). However due to the scope of the current research, only the micro-level perspective will be discussed.

One of the most complete presentations of open source software economics, is made by Lerner and Tirole (2001). They attempt to identify the extent to which labour economics can explain open source's features. In particular, they state that programmers participate in free/open source software, for the same reason as they do for commercial software. That is, only if they derive a net benefit from their activity. They define net benefit as equal to immediate payoff plus the delayed payoff. The former payoff comprise of immediate benefits and costs, while the latter comprise of delayed benefits and costs. In particular:

---

[11] In particular the following are presented: the GNU Public Licence (GPL), the BSD license, the Artistic Licence (Perl license), and the Mozilla licence (MPL).

- Immediate benefits are i) the programmers' personal needs – this is what Raymond (1999) calls "scratching a developer's personal itch", ii) contributing to a free/open source software project may be more enjoyable than performing a routine task at the workplace.

- Immediate costs are *opportunity costs*. That is, i) programmers are using their time in an activity that is not paid for, although they could have done the opposite or rest, ii) as a result, their performance on their paid-for work decreases.

- Delayed benefits. They use "economics of signalling incentives" to discuss this. This is an umbrella term capturing both "ego gratification" and "career concern incentives". The former can be thought of as a synonym of self-actualisation (see Maslow 1970). The latter is based on peer recognition and relates to the fact that contribution to Open Source Software enhances future career prospects for developers.

- Delayed costs are analogous to those that exist in the academic context. In the same way as in the academic research, fads attract the attention of many researchers, programmers may find themselves involved in a project or a projects fork only to be proven that the didn't worth it, in the future. However, it should not be ignored that fads also have benefits.

Apart from implicit economic incentives there are cases where leading companies – like IBM, Netscape, Hewlett-Packard – are paying for OSS development. As described by Jørgensen (2001), 41% of developers are getting paid for their contribution.

Chapter 5: Empirical Research

This section's purpose is to provide awareness, upon the way that the underlying research was conducted. That is, awareness upon its research approach and chosen method. By research approach it is meant the general style of a research endeavour, while by method it is meant the application of a set of distinct techniques (Braa and Vidgen 1997, Cornford and Smithson 1996, Galliers 1992, Myers 1999) – see also [Appendix II](). In order to provide a better understanding of the above, a brief presentation of the research philosophies that are available in social sciences will be made. Consequently the way that the current research was conducted is presented as well.

5.1 Approach

Within the social sciences discipline there are alternative schools that study social reality using different viewpoints. Valid knowledge in not given, in this sense, but questioned depending on the epistemology adopted by the researcher. An early work on this issue that extensively influenced social research was made by Burrell and Morgan (1979) who proposed a continuum of epistemology that runs from positivism to anti-positivism.

After Burrell and Morgan, research approaches have variously been classified as objective versus subjective. Inline with this classification, scholars distinguish between two classes of research approaches, that is *Scientific* – often called positivist – and *Interpretivist* (Galliers 1992, Lee 1999). Another very important classification of research epistemologies can be found in the comparison study of *positivist*, *interpretive*, and *critical* research, by Orlikowski and Baroudi (1991). Positivist**,** generally assume that reality is objectively given and can be described by measurable properties which are independent of the observer (researcher) and his or her instruments. Interpretative, researchers start out with the assumption that access to reality (given or socially constructed) is only through social constructions such as language, consciousness and shared meanings. Critical, researchers assume that social reality is historically constituted and that it is produced and reproduced by people.

Since social systems include people, there is the variability and psychological opaqueness of the human mind and human intention to content with (Cornford and Smithson 1996). Thus the most appropriate path to deal with the Free/Open Source Software phenomenon as a social phenomenon is the interpretative approach.

5.2 Method

Research methods can be classified in various ways, however one of the most common distinctions is between qualitative and quantitative research methods. Quantitative research methods were originally developed in the natural sciences to study natural phenomena. Qualitative research methods were developed in the social sciences to enable researchers to study social and cultural phenomena.

The motivation for doing qualitative research, as opposed to quantitative research, comes from the observation that, if there is one thing which distinguishes humans from the natural world, it is our ability to talk! Kaplan and Maxwell (1994) argue that the goal of understanding a phenomenon from the point of view of the participants and its particular social and institutional context is largely lost when textual data are quantified. Qualitative research methods are designed to help researchers understand people and the social and cultural contexts within which they live.

Qualitative research refers to the use of qualitative data, for instance interviews, documents, and participant observation data, to understand and explain social phenomena. The word "qualitative" is not a synonym for "interpretive" - qualitative research may or may not be interpretive, depending upon the underlying philosophical assumptions of the researcher. Qualitative methods can loosely be defined as "those which eschew metrication and seek other means of capturing and analysing (understanding) data" (Cornford and Smithson 1996).

As mentioned already, the current study follows the interpretative approach. Moreover, it follows a qualitative research method to study FS/OSS contributors' motivations. Given the geographical breadth of FS/OSS participants the most efficient way to conduct the aforementioned research is online. Computer facilitated research has been credited as a robust way to reveal information about users, their attitudes, and their actions, that is otherwise impractical or nearly impossible to gather (Mann

and Stewart 2000). Indeed this approach is imperative in the context of FS/OSS, since their very essence relies on the Internet.

The main research method considered involves an approach through Usenet newsgroups and mailing lists posting. These newsgroups and mailing lists are electronic sources for distribution, inquiry, retrieval, and posting of news articles. Such methods offer rapid dissemination of items of interest such as software bug fixes, new product reviews, technical tips, and programming pointers, as well as rapid-fire discussions of matters of concern to the working computer professional (RFC-977). Furthermore there exists a well defined way of member communication (RFC-1855). This is proven to be an effective way of collaboration as several Newsgroups and mailing lists, nowadays, are specialised in particular topics of interest.

In particular, the current research was conducted using online surveys and online interviews (see Appendices IV, V, and VI, VII respectively). The web-based-survey offered considerable advantages in terms of increased reach by collapsing boundaries of time and space (Mann and Stewart 2000), as well as a fruitful ground for finding interviewees. The interviews then followed a semi-structured fashion. They were fairly formalised, using an interview protocol organised into specific thematic areas. Supplementary questions – sometimes called probes (Mann and Stewart 2000) – were introduced in a spontaneous manner to seek further clarification and elaboration of answers. The main advantage of semi-structured interviews is that the impose "purposive topical steering" (Mann and Stewart 2000); that is, tracking the issues which are more interesting.

Last, but not least important issue of a qualitative research method, is the question of sampling. Indeed this is one of the major issues when conducting social research (Maisel and Persell 1995). A survey can take the form of a census survey where every member of the target population is surveyed. However, when this is not feasible sampling is imperative. Indeed, the latter, was the case in the FS/OSS context. Sampling was focused in the sense that an attempt was made to find individuals who have experienced the FS/OSS phenomenon, in particular by contributing to it. This loose sampling method was considered important, bearing in mind the disadvantage

of selecting a single or a few free/open source software projects. This would introduce

bias to the underlying research (Lancashire 2001).

Chapter 6: Empirical Findings and Critical Analysis

Following the literature study, an invitation for participation to an on-line survey was sent to contributors of free/open source software via email. Their email accounts were reached by means of FS/OSS mailing lists and discussion lists over the Internet. These groups included both general open source communities and specific free/open source software programmers' forums. In particular an email with the URL of the online questionnaire was sent to the following lists: arch@FreeBSD.ORG, open-source-now-list@redhat.com, linux-user@egr.msu.edu, cgl_info@lists.osdl.org, freebsd-chat@FreeBSD.org, current@FreeBSD.org, gllug@linux.co.uk. Twenty-three people responded to the questionnaire and seven declared an interest in being interviewed online; Finally out of the seven contributors that declared an interest to be interviewed, six replied to the interview questions.

6.1 Questionnaire Responses

The purpose of the questionnaire was to collect respondents' demographics and moreover, it was also used as the starting point in to order find volunteers for the interview and consequently to set the interviews in context. The collection and analysis of the questionnaire responses yields the following results:

The entirety of the respondents were male. Most, that is 18 respondents, were between 20 and 30 years old. The majority, 19 of them, had been programmers for between 4 and 10 years, and 3 had been programmers for more than 10 years. All the respondents have been contributing to FS/OSS for less time than they have been programmers. In particular, 16 have been contributing to FS/OSS for a period between 2 and 8 years, while only 2 have been contributing for more than 8 years.

Most participants, that is 20 respondents, possessed an undergraduate degree or higher. Almost three quarters, that is 17 respondents, were professional programmers, while 11 of these characterised themselves as both professionals and hobbyists. One fifth, that is, 5 respondents are directly or indirectly paid for their contribution. The remainder were composed of 3 students and 3 professionals that programme in their spare time.

A very interesting figure is that the majority, that is 20, have initiated a project themselves. In response to the question, "When you first joined a project's community, which were your expectations?", 12 responded "as a means to improve a particular project", 4 "by mere curiosity", and 6 as a means to "improve my programming skills". The minority, that is 2, is getting paid for their contribution. On the questions of expected rewards, 6 explicitly declared that they expect monetary reward for their contribution, and on the question whether they expect other (aside from monetary) pay-offs/rewards 17 responded "rather true" or "true" while 5 responded "rather false" or "false".

Questioned regarding their incentives for participation; 5 responded "to improve a project", 4 "acquiring specialised programming knowledge", 14 "better use of my skills, than in any other Software Engineering context", 19 "future career potentials", and 2 "due to my general principals". In the related to public domain software participation question 13 responded "no, and will not in the future either" while 6 have participated in the past to public domain software. Lastly and very interestingly, the entirety of the sample responded "rather true" or "true" to the question of whether they "abide to the norms and rules of community".

6.2 Interviews

Complementary to the online questionnaires, another qualitative method was used as a means to allow respondents to elaborate their answers; in particular, online interviews. The interviews followed a semi-structured fashion (see for example Appendix VI). The most relevant, to the discussion, parts of the interviews are quoted in Appendix VII.

6.3 Critical Analysis

Perspectives like community's theory, motivational psychology, governance, and economics, are the most popular approaches that have been employed to answer the question of participant's motivation. As it has been claimed already each of those attempts addresses the issues only partially. The objective of this section is to discuss the strong points of each approach, their shortcomings, as well as to justify why they should be used as a composite framework to understand participant's motivation.

6.3.1 Communities

Theory of epistemic communities can adequately answer questions related to the motivational issues of participants in FS/OSS development. Contributors are motivated by a particular need for software. Once they have entered the community the primary motivational factor that keeps them in the community is their sought for reward in terms of reputation and recognition from ones peers. Indeed, in their questionnaire responses 16 out of 23 – that is 69% of the sample – mentioned "recognition from peers" as their motive for participation to FS/OSS development. Furthermore, reputation concerns are obvious in the interviews (Appendix VII).

However, epistemic communities anticipate a power vacuum that is filled in by the community. This power vacuum does not seem to exist in the FS/OSS context. Furthermore, this approach cannot answer the question of how participants are motivated in the first place to enter the community (Edwards 2001). Moreover, each social phenomenon, that involves a community structure, must be studied in respect to its geographical, legal – that is governance, and economic aspects. Given the geographical distribution of FS/OSS contributors, it is not possible to socialise people into the community and create a common understanding and shared frame of reference. Contributors have significantly different mindsets, they are motivated by need and interest in solving the problem, and far less by socialising (Edwards 2001). Hence, the motivation issue cannot be fully approached by community theories.

6.3.1.1 Theoretical Intertwine

According to the Encyclopaedia of the Social Sciences, in the early social sciences literature the concept of community was defined as something that was part of a larger society, a "**geographical area** with definite **legal boundaries**, occupied by residents engaged in interrelated **economic activities** and constituting a politically self-governing unit [emphasis added]" (Lindeman 1962).

Another very interesting question that epistemic communities fail to answer is how do participants choose the software that they participate. Perhaps an answer to this can be found in Hansmann's (1980, 1981, 1987) economic theory and rationale for the non-profit sector.

Furthermore, programmers often identify themselves as part of the free/open software community and align their goals with those of the community. Hars and Ou (2001) call this "community identification" and they categorise it as a motivation in the same group of needs as belonging and love (see Maslow 1970).

Following the above analysis, it is argued that the epistemic communities approach alone, is not adequate to explain participants' motivation and that it is interlinked with issues of motivational psychology (see for example Hars and Ou 2001, Maslow 1970), economics of non-profit (Hansmann 1980, 1981, 1987). Moreover, bearing in mind the definition of community by Lindeman (1962), it can be argued that an obvious link exists among epistemic communities, governance structure and economics.

6.3.2 Motivational Psychology

The understanding of open free/source software development using motivational psychology approach appears useful. Motivational psychology seems fruitful in explaining participants' motivation to contribute software. For their action fulfils, as analysed by (Raymond 2000) their self-actualisation need, as well as their "community identification" need (Hars and Ou 2001).

As presented already *Intrinsic motivation (internal factors)* include the desire of feeling competence and self-determination; *extrinsic motivations (external rewards)* include factors such as direct or indirect monetary compensation, and other's recognition as well (Deci 1975, Hars and Ou 2001); whereas *idealistic motivations* refer to contribution without the expectation to ever receive any individual reward (Franck and Jungwirth 2002). As it was revealed by the interviews conducted, all types of psychological motivation are strong motivators in FS/OSS contribution. In Appendix VII the reader can find citations from participants' statements that can be categorised as intrinsic, explicit, both intrinsic and explicit, and idealistic motivation respectively.

However, as in the case of epistemic communities the question of how contributors choose the software to which they will contribute remains unanswered.

6.3.2.1 Theoretical Intertwine

Frey (1997), possesses that extrinsic motivation (e.g. monetary rewards and regulation) may drive one's intrinsic motivation to perform a task. This phenomenon is called *crowding-out effect*. At a subsequent research work, Frey and Jegen, verified that "the motivation crowding effect suggests that an external intervention via monetary incentives or punishments may undermine (and under different identifiable conditions strengthen) intrinsic motivation" (Frey and Jegen 2001). As a consequence of this research work it can be argued that motivational psychology alone, cannot answer the question of participants motivations in FS/OSS projects since economic matters come into the surface.

In addition, Franck and Jungwirth (2002) following their distinction between, investors and donators, make a very interesting point, in relation to the crowding effect; they argue that "the basic innovation in open source has been the crafting of a governance structure, which enables investment without crowding out donations". Demonstrating thus, that motivational psychology is interlinked with governance issues.

Following the above analysis, it can be argued that the motivational psychology approach alone, is not adequate to explain participants' motivation and that it is interlinked with issues of economics (Frey and Jegen 2001) and governance (Franck and Jungwirth 2002).

6.3.3 Governance

The strong point of code as governance is that, as opposed to previous approaches (e.g. epistemic communities and motivational psychology), it can explain participants' motivation to join a project. That is, if the licensing underlying a particular piece of software is in accordance with his personal objective or worldview, he or she has an incentive to join the development process. Indeed, one of the interview findings (Appendix VII) is that participants are carefully choosing the FS/OS software to which they will contribute.

However this perspective does not answer the question why participants would continue their contribution when another program with similar or identical licensing is launched.

6.3.3.1 Theoretical Intertwine

At this point Frey's work can come into the surface again. While he discusses the consequences or the crowding effect, he also acknowledges that regulations can sometimes be used to prevent free riding and establish equity and fairness. In fact, most regulations are designed to protect the public from the potential ill effects of others' (often intrinsic) motivation (eg. antitrust laws, property rights and criminal laws) (Frey 1997).

Aligned with this argument, Franck and Jungwirth (2002) categorise participants as investors and donators. Following this categorisation, they then draw their main argument, which proposes that a licensing agreement plays a key role in reconciling investors and donators in open source. "Without a governance structure economizing in the symbiotic relationship of investors and donators, open source will not work" (Franck and Jungwirth 2002) . This governance structure is suited to attract donators. "It turns out that the same institutional prerequisite, a specific licensing agreement for software, ultimately enables the design of a rather complex incentive structure for investors based on reputation, and at the same time serves as a device against contract failure in attracting donators" (Franck and Jungwirth 2002).

Following the above analysis, it can be argued that the governance structure of FS/OSS, is not adequate to explain participants' motivation and that it is interlinked with issues of economics as well as psychological motivation (Frey and Jegen 2001).

6.3.4 Economics

Economics seems the most rational approach of all those previously mentioned. However, it can be said that, it attempts to assign extensive rationality in the not very "rational" context of free/open source software development. The explanations provided from this approach can definitely explain why programmers would want to join the free/open source software development. In particular, using the categorisation of Lerner and Tirole (2001), interviewees' comments are presented in Appendix VII.

However, this approach, does not explain how FS/OSS contributors choose the particular software. More importantly it fails to explain why contributors would continue their participation once they have reached the point where they could "sell" their skills. For example Torvalds' statement "I want to continue to try to avoid making money directly off Linux - that keeps me focused on purely technical issues with the Linux kernel" (Ghosh 1998b), is not easily explained by economic incentives.

6.3.4.1 Theoretical Intertwine

As discussed already, participating in free/open source software is a very effective way to improve development skills. "in an economic sense, developers may have nothing to gain in economic terms by keeping development to themselves. Thus, in a game-theoretic sense, there is nothing to lose by opting for OSS, even if any financial benefits do not seem immediately obvious" (Feller and Fitzgerald 2002). As (Raymond 1999) puts it, in other words, there are three primary benefits that accrue to successful contributors of open source projects "good reputation among one's peers, attention and cooperation from others, … [and] higher status … [in the] exchange economy." In Linus Torvalds' words: "I don't exactly expect to go hungry if I decide to leave the University. 'Resume: Linux' looks pretty good in many places" (Ghosh 1998b).

Thus, while some of benefits conferred from participation in open source projects may be less concrete in nature, there also appear others that – although delayed – are quite tangible rewards. It is well known that employers scan the credit lists of OSS products to identify promising talents, and thus programmers increase their value in the commercial software market. Speaking in concrete figures, Eric Raymond is estimated to have become wealthy to the tune of $47m (£31m) when VA Linux went public in 1999, while Linus Torvalds is estimated to worth $20m (£13m) (Feller and Fitzgerald 2002).

Following the above analysis, it can be argued that economics is not an adequate perspective to explain participants' motivation. There is an obvious link between peer recognition, in the epistemic community sense, and delayed benefit in the economic sense (Edwards 2001, Lerner and Tirole 2001), external psychological rewards can be

considered as economic rewards (Frey and Jegen 2001, Lerner and Tirole 2001), and furthermore, governance of FS/OSS is a definitive factor of economic incentives and external rewards (Franck and Jungwirth 2002).

6.4 A Depiction of Motivations Intertwine

For the reader's convenience as well as for clarification of the above analysis, the following matrix illustrates the intertwined nature of the presented approaches towards FS/OSS participants' motivation.

	Communities	Motivational psychology	Governance	Economics
Communities				
Motivational psychology	(Hars and Ou 2001, Raymond 2000)			
Governance	(Lindeman 1962)	(Franck and Jungwirth 2002, Frey 1997)		
Economics	(Edwards 2001, Hansmann 1980, Lerner and Tirole 2001, Raymond 1999)	(Edwards 2001, Frey 1997, Frey and Jegen 2001, Frey and Osterloh 2000, Lerner and Tirole 2001)	(Franck and Jungwirth 2002, Frey 1997)	

Chapter 7: Discussion

The fact that none of the approaches discussed herein can adequately answer the question of FS/OSS participants' motivation should not come as a surprise. It is indeed a proof that a new phenomenon is rising – or at least that a new and growing interest emerges for social sciences research.

Human beings create new knowledge through what they already know (Bruner 1990, Bruner, *et al.* 1986). Our understanding consists in grasping the place of an idea or fact in some more general structure of knowledge. When we understand something, we understand it as an exemplar of a broader conceptual principle or theory (Bruner 1990). Furthermore, new knowledge in order to be spread, it must be articulated through the use of language, which is both descriptive and constituent (Giddens 1976).

Scientific discovery is not different in this perspective; it needs to be expressed upon previous knowledge, using language, as its expression medium. Hence, it is formed through a process of metaphor[12] and analogy[13] (Leatherdale 1974, Nonaka 1994, Tsoukas 1989, Tsoukas 1991). Given that there are always things that we are still exploring and others that we have explored already, and hence we possess knowledge about them, it is to be expected that metaphors and analogies will always be used in scientific discourse (Tsoukas 1991, Weick 1989). In effect, FS/OSS is one of those things that, in social sciences, we are still discovering.

In simple words, free/open source software development is a relatively new field of social science research. We use – for example – epistemic communities as a means to explain it only because epistemic communities preceded free software. If the opposite were the case we would try to explain epistemic communities using free software theory! Of course, we should not expect a theory that was developed to explain one phenomenon to be as effective to explain another. Thus, we must be aware that no

---

[12] "The essence of metaphor is understanding and experiencing one kind of think in terms of another" (Lakoff and Johnson 1981).

[13] "Analogy allows the functional operation of new concepts or systems to explore the reference to things that are already understood" (Nonaka 1994).

previous conceptualisation would be adequate to explain free/open source software contribution. Rather a composite perspective, based on metaphors and analogies, perhaps enhanced with new elements, is imperative.

Chapter 8: Conclusions

When open source programmers volunteers their time into writing code for an FS/OSS project, they freely give away the code they write for the good of the FS/OSS community; at the same time, they benefit from the contributions of others within this community by getting really good software, and so the whole movement benefits and grows. People identified with the community take pride in their work. They receive satisfaction in knowing they have contributed an excellent feature, or elegant piece of code. Comparing this to commercial programmers illustrates the major differences that underlie software development. The conclusion that this study reaches is that the consequences of these different sources of motivation lead to very different social arrangements that call for a multidisciplinary study.

Researchers have extensively studied the motivations for those that volunteer their time into building and writing code for free. This study attempted to explore the consistency or variance of the given answers to the question of FS/OSS participants' motivation. It has been found that 89% of the approaches adopted by published studies are divided among four major categories; namely, theory of communities, motivational psychology, licences as a means of code and ultimately governance, and economics.

However a comprehensive answer to participants' motivation does not seem that easy. It has been shown that different approaches address different aspects of the process and that each approach needs to employ others in order to provide a more complete answer. In particular, in this paper it has been argued:

Firstly, that epistemic communities anticipate a power vacuum that does not seem to exist in the FS/OSS context. Furthermore, this approach cannot answer the question of how participants are motivated in the first place to enter the community. Moreover, each social phenomenon, that involves a community structure, must be studied in respect to its geographical, legal – that is governance, and economic aspects. Given the geographical distribution of FS/OSS contributors, it is not possible to socialise people into the community and create a common understanding and shared frame of

reference. Hence epistemic communities approach alone, is not adequate to explain participants' motivation. It must be rather studied in accordance with issues of motivational psychology, and economics of non-profit. Moreover, bearing in mind the definition of community by Lindeman (1962), it can be argued that an obvious link exists among epistemic communities, governance structure and economics.

Secondly, that motivational psychology, as in the case of epistemic communities cannot adequately answer the question of how contributors choose the software to which they are contributing, in the first place. Moreover it is argued that the motivational psychology approach alone, motivation and that it is interlinked with issues of economics, through the concept of extrinsic motivation - external rewards as well as with governance, since licensing is a determining factor of whether contributor will be able to satisfy their psychological needs through their contribution to a particular piece of FS/OSS software.

Thirdly, that governance does not answer the question why participants would continue their contribution when another software with similar or identical licensing is launched. Hence, it can be argued that the governance structure of FS/OSS, is not adequate to explain participants' motivation and that it is interlinked with issues of economics as well as psychological motivation.

Lastly, that the economics approach does not explain how FS/OSS contributors choose a particular software. More importantly it fails to explain why contributors would continue their participation once they have reached the point where they could "sell" their skills. Hence it can be argued that economics is not an adequate perspective to explain participants' motivation. There is an obvious link between peer recognition, in the epistemic community sense, and delayed benefit in the economic sense. External psychological rewards can be considered as economic rewards. Furthermore, governance of FS/OSS is a definitive factor of economic incentives and external rewards.

The conclusion that this study arrives to, after studying the literature on motivations and conducting both quantitative and qualitative empirical research, is that a multidisciplinary approach is required to understand participants' motivations in

FS/OSS projects. This conclusion should not be surprising. On the contrary, since FS/OSS is a new area of interest for the social sciences we must expect that the developing knowledge that concerns its essence will be articulated through metaphors and analogies of already developed theories. Hence, neither of the perspectives, presented herein, can be used exclusively to explain the issue of FS/OSS participants' motivation, rather they are inextricably linked and must be studied as such.

Chapter 9: Research Limitation and Further Research

9.1 Research Limitation

The arguments presented in this paper, derive from earlier studies in the field, as well as from theoretical thinking and empirical research, which was conducted by the author. This combination of approaches, gives to the current work an added value, and a strong standpoint in order to defend its arguments. However, there are limitations that derive from this process which are discussed bellow.

The major limitation of this study is that, the author did not participate in or observe FS/OSS development activities directly. The conclusions come from the literature review as well as from questionnaire and interview data. However ethnographic data would offer more grounded accounts of participants' motivation. Researching by means of questionnaires and interviews is an adequate starting point to begin a research project. However safer conclusions can be drawn be means of participation and observation. Participation in FS/OSS can bring to this research a more in depth perspective on the reasons of participation as well as a means to elaborate further the research questions.

Additionally, the subject of the empirical research can be considered small, especially given the fact that the research wasn't focused in any particular FS/OSS project. However, not choosing a particular project was deemed crucial, since doing otherwise would introduce bias to the research. Nevertheless given the small size of the sample, statistically conclusive findings cannot be drawn from this research.

However, participation in FS/OSS and questioning more volunteers for the empirical research, would not have been possible in the context of a summer dissertation project. Participating in FS/OSS, as well as collecting and analysing further empirical data, would have exceeded the required time-space of the current research assignment. It is strongly argued though that such an approach can further reinforce the stand taken in this paper, and hence it is perceived as a field for further research.

9.2 Further Research

The Free/Open Source software production seems to be only the beginning of a new way of working which can be extended into all areas of economic activity. Although promising, currently the model is only observed in 'digital goods' and research environments, such as the genome project or nuclear physics. Most of the work in the field of Free/Open Source has focused on technology and engineering issues, as well as the motivation driving individual developers to participate in open source projects.

Recently, several researchers are discussing ways that the paradigm of free/open source software development might influence the future of employment, and rewards. For example, Feller and Fitzgerald (2002), discusses potential ways that the FS/OSS paradigm be transported to other industries. Indeed this is a promising field for further research, which again calls for a multidisciplinary approach. In particular, it requires analysis of contributors' motivation, analysis of industrial and organisational cultures as well as widespread observation and empirical research.

Recently, several researchers are, discussing potential ways that the FS/OSS paradigm could be transported to other industries (Feller and Fitzgerald 2002). For example, Dinkelacker and Garg (2001), and Masum (2001) are discussing ways that the paradigm of free/open source software development might influence the future of employment, and rewards. Indeed this is a promising field for further research, which again calls for a multidisciplinary approach. In particular, it requires analysis of contributors' motivation, analysis of industrial and organisational cultures (Schein 1992) and formative context (Ciborra and Lanzara 1994, Unger 1987), as well as widespread observation and empirical research.

Appendix I – Open Source Timeline

Year	Event
1950s and 1960s	Software source code is distributed without restrictions in IBM and DEC user groups, ACM's Algorithms Section etc.
1969	Ken Thompson writes the first version of UNIX. Its source code is distributed freely throughout the seventies.
1978	Donald Knuth (Stanford) publishes TEX as free software
1979	Following AT&T's announcement to commercialize UNIX, UC Berkeley begins with the creation of its own version of UNIX, BSD (Berkeley Software Distribution). Eric Allmann, a student at UC Berkely develops a program that routes messages between computers over ARPANET. It later evolves into Sendmail.
1983	Stallmann publishes GNU Manifesto calling for free software, and establishes Free Software Foundation.
1986	Larry Wall creates Perl (Practical Extraction and Report Language), a versatile programming language used for writing CGI (Common Gateway Interface) scripts.
1987	Developer Andrew Tanenbaum releases Minix, a version of UNIX for the PC, Mac, Amiga, and Atari ST. It comes with complete source code.
1991	Linus Torvalds publishes version 0.02 of a new UNIX variant that he calls Linux in a Minix newsgroup.
1993	FreeBSD 1.0 is released. Based on BSD Unix, FreeBSD includes networking, virtual memory, task switching, and large filenames. Ian Murdock creates a new linux distribution called Debian Linux.
1994	Marc Ewing forms Red Hat Linux. It quickly becomes the leading Linux distributor. Bryan Sparks founds Caldera with backing by former Novell CEO Ray Noorda.
1995	The Apache Group builds a new Web server, Apache, based on the National Center for Supercomputing Applications' (NCSA's) HTTPd 1.3 and a series of patch files. It has become the dominant HTTP server today.
1998	Netscape not only gives away Communicator 5.0 (Mozilla) but also releases its source code.
1999	Number of Linux users estimated at 7.5 Million.
2000	More software companies such as Novell and Real release versions of their products which run on Linux.

Adopted from Hars & Ou (2001)

Appendix II – Research Approaches for Social Systems

The Four Paradigms of Social Theory

	Objective	**Subjective**
Regulation	FUNCTIONALIST	INTERPRETIVIST
Radical Change	RADICAL STRUCTURALIST	RADICAL HUMANIST

Adopted from (Burrell and Morgan 1979).

Information Systems Research Approaches

Scientific	**Interpretivist**
Laboratory experiments	Subjective/argumentative
Field experiments	Review
Surveys	Action research
Case studies	Descriptive/interpretive
Theorem proof	Futures research
Forecasting	Role/game playing
Simulation	

Information Systems research approaches in the context of scientific and interpretivist philosophies. Adopted from (Galliers 1992).

Appendix III – Literature Research Findings

British Library of Political & Economic Science – Electronic Journals

The articles considered are coming from various academic resources. In particular the following journals were considered, in alphabetical order: European Journal of Information Systems, Inform - The Journal of Policy, Regulation and Strategy for Telecommunications Information and Media, Information Systems Journal, Journal of Industrial Economics, Journal of Information Technology, Oxford Review of Economic Policy, and Proceedings of the 34th Hawaii International Conference on System Sciences. Nine articles were considered and based on their main argument they can be classified as follows:

- Motivation as depending on specific elements of the development life cycle (Jørgensen 2001)

- Academic communities: (Bergquist, et al. 2001)

- Idealistic motivations: (Forge 2000)

- Internal factors (e.g., intrinsic motivation, altruism) and external rewards (e.g., expected future returns, personal needs) Hars and Ou (2001)

- Governance: (Ciborra and Andreu 2001, Kogut and Metiu 2001)

- Economics: (Lerner and Tirole 2001)

- Gift economy: (Bergquist and Ljungberg 2001, Ljungberg 2000)

Free/Open Source Research Community

In particular twelve articles were considered. Based on their main argument they can be classified as follows:

- Communities: (Butler, et al. 2002)

- Innovation Communities: (von Hippel 2001)

- Epistemic communities: (Edwards 2001)

- Communities of practice: (Faraj and Wasko 2001, Franke and Shah 2001)

- Intrinsic motivation: (Dafermos 2001b, Lakhani and von Hippel 2002, von Hippel 2002)

- Governance: (Franck and Jungwirth 2002)

- Gift economy: (Metiu and Kogut 2001)

- Conventional economics: (Lerner and Tirole 2000)

- *Theory of Professionals* and *Economics of clubs*: (Garzarelli 2002)

- *Private Investment* and *Collective Action* in what they call "private-collective" innovation model (von Hippel and von Krogh 2002).

First Monday

In particular fifteen articles were considered. Based on their main argument they can be classified as follows:

- Communities: (Kuwabara 2000)

- Hacker Communities: (Hannemyr 1999)

- Epistemic communities: (Bezroukov 1999, Kelty 2001)

- Communities of practice: (Moon and Sproull 2000, Tuomi 2001)

- Intrinsic motivation (includes reputation): (Dafermos 2001a, Ghosh 1998a, Ghosh 1998b, Raymond 1998a)

- Governance: (Söderberg 2002)

- Gift economy: (Barbrook 1998, Raymond 1998b)

- Conventional economics: (Lancashire 2001)

- Leadership: (Edwards 2000)

Appendix IV – Research Contributors

The identification and a brief description of those who contributed to this research can be found below. Please note that only those research contributors that agreed to have their names revealed are listed in alphabetical order.

Patrick Cable (pcable@slaudiovis.org)

Patrick Cable is a teenager and still at school. He is currently working on LiveWriter, which is a tool for high school English teachers & students - students submit their essays online and teachers grade them online, and it keeps a digital portfolio of sorts.

Daniel Carrera (dcarrera@math.toronto.edu)

Daniel is a PhD student in Mathematics, he has been a programmer for 8 years and he is contributing to FS/OSS for the last 2 years. He has participated to the "Mono" project, which is a free implementation of the .NET framework.

Philip Hands (phil@hands.com)

Philip Hands (http://www.hands.com) is the founder of "hands.com ltd" which has been providing "consultancy and annual support contracts to discerning clients since 1993, specialising in Free Software, and Unix-like operating systems". He is a strong proponent of Free Software, which in his words "has recently reached wider public awareness under the name of 'Open Source' or 'OSS' ". He is responsible for the compilation of official Debian CD images and – among other projects – he is a frequent contributor to the following projects: Debian, rsync, samba, mgetty+sendfax, LRP, OpenSSH.

Matt Rowland (matt@paperlove.org)

Matt has been a programmer for 2 years. He is involved in the FS/OSS community for less than a year, "but here to stay". He has initiated, in his words, "just another content management system" and he is also contributing to GOVIA and SchoolForge.

James Strachan (james_strachan@yahoo.co.uk)

James Strachan (http://www.apache.org/~jstrachan) has been a programmer for over 20 years and a frequent Open Source contributor. He participates in numerous Jakarta projects such as: jelly, maven, commons, taglibs as well as other projects such as dom4j, jaxen, saxpath. He is a professional software developer and his employer (http://www.spiritsoft.com) supports his contributions.

James Strachan (james_strachan@yahoo.co.uk)

Appendix V – Questionnaire

Online Questionnaire

The contributors were able to fill in an online questionnaire, then by clicking on "send form" the questionnaire would be forwarded to my email account at LSE. That would reveal the sender's email address, so in order to ensure confidentiality a .txt based questionnaire was available as well. Following is the questionnaire as it appeared on: http://www.lse-students.ac.uk/tzouris/oss/.

Free Software / Open Source Questionnaire Version 2.0

Dear contributor,

Thank you for volunteering to complete this questionnaire, which results will be used for my MSc dissertation. Your help is appreciated and acknowledged.

The questionnaire is designed in a way to require approximately **10 minutes to fill in**. Please be assured that any information you provide will be used with the strictest of confidence, and you will not be identified in any way unless agreed by you (please see the Data Protection Statement).

Please answer all questions.

Thank you very much for your help,

Menelaos Tzouris

PS: This questionnaire will pop up your e-mail client and send an e-mail to me on your behalf, for your convenience. However if you prefer, you can fill in a text based questionnaire and manually email it to me.

Declaration

Please tick this box if you wish your identification details to be disclosed? ☐

Would you be interested in receiving a copy of the final report? ☐

Identification

Name: [   ] (optional)

Software Freedom, Open Software and the Participant's Motivation
A Multidisciplinary Study

Surname: [   ] (optional)

Nickname: [   ]

Age (in years) [14] [Please choose one: ▼] (optional)

Gender[15] [Please choose one: ▼] (optional)

Nationality: [   ] (optional)

Section A: Background

Education[16]: [Please choose one: ▼]

other: [   ]

Occupation[17]: [Please choose one: ▼]

other: [   ]

How long have you been a programmer? (in years)[18] [Please choose one: ▼]

How long have you been contributing to FS/OSS? (in years)[18] [Please choose one: ▼]

Section B: Main Questionnaire

1) Have you ever initiated a project yourself? If yes please give brief details.

[No.   ]

2) To which projects are you participating/or have you participated in the past? (please name)

---

[14] Possible entries: less than 18; 19-22; 23-26; 26-29; above 30

[15] Possible entries: Male; Female

[16] Possible entries: High School; Undergraduate Degree (BSc, BA, etc); Graduate Degree (MSc, MA, MPhil, PhD, etc); Other: please specify

[17] Possible entries: Software Engineer; Student; Other, please specify

[18] Possible entries: less than 2; 3-5; 5-7; 7-9; more than 10

[text area]

3) When you first joined a project's community, which were your expectations[19]

Please choose one: ▼

other: [   ]

4) Would you describe yourself as a "Hobbyists" instead of a "Professional" Computer Programmer?[20]

Please choose one: ▼

Are you paid for your contribution? Please answer questions in both columns if applicable.	
If YES	If NO
5) Which company(ies) support(s) your contribution? [text area]	7) You **do expect** that **other pay-offs (aside from money)** will turn your effort into a profitable investment.[21] Please rank the truth of this statement: ▼
6) Please provide brief description of this placement. [text area]	8) Please describe what other rewards you are expecting, if applicable. [text area]

9) Please specify the incentives for your participation in order of importance. (a:most important, e:least important) [22]

a) Please choose one: ▼

other: [   ]

---

[19] Possible entries: None/ Mere Curiosity; Improving my programming Skills; Meeting people with similar interests; Improving a particular project; Other: please specify

[20] Possible entries: Hobbyists; Professional; Both

[21] Possible entries: (1):True. "Yes, I do expect other rewards."; (2):Rather True.; (3):Neither true, nor false.; (4):Rather False.; (5):False. "No, I do not expect other rewards."

[22] Possible entries: (1):Recognition from your peers.; (2):Future career potentials.; (3):Better use of your skills , than in any other software engineering context.; Other: please specify

b)

Please choose one: ▼

other:

c)

Please choose one: ▼

other:

d)

Please choose one: ▼

other:

e)

Please choose one: ▼

other:

10) Would you identify your self as a contributor to Free Software or Open Source Software? [23]

Please choose one: ▼

11) Have you ever participated in a Public Domain Software, rather than Free Software / Open Source Software? [24]

Please choose one: ▼

12) You are aware and you abide to the norms and rules of the project's community? [25]

Please rank the truth of this statement: ▼

13) Any other comments you would like to make on your participation?

Section C: Open Ended Questions (optional)

---

[23] Possible entries: Free Software; Open Source Software; I am indifferent

[24] Possible entries Yes; No; "No, and I don't think I will in the future either"; "No, but I might"; I don't really know the difference

[25] Possible entries (1):True.; (2):Rather True.; (3):Neither true, nor false.; (4):Rather False.; (5):False.

This part of the questionnaire is optional. However, you are kindly requested to provide brief answers to all or some of the following questions.

1)   How do you perceive other FS/OSS contributors? In particular those who participate to the same project as you are. Please describe briefly.

2)   What influenced you to write free / open source code?

3)   According to what criteria you choose the project in which you will contribute?

Debriefing

If you like, you can send me your comments, concerning this questionnaire. Per example things that you did not like, others that you did, vague questions, etc.

Are you interested in being interviewed online?

Please tick this box ☐ if you are interested in being contacted by me for an online interview as a supplement to this questionnaire.

Submit Questionnaire via Email

To: m.tzouris@lse.ac.uk

Cc:
(please add your e-mail here if you are interested in receiving a copy of the generated e-mail)

Subject: Free Software / Open Source Questionnaire Version 2.0

Send Form

Thank you very much for your time,
Menelaos Tzouris

When the form is submitted the contents also appear here.

Data Protection Statement

Details of the names and email addresses of people who answer this online questionnaire are kept for the purpose of email correspondence. They will only be revealed if explicitly agreed by you to the final version of my dissertation; as well as in electronic or other forms that may derive by it. Likewise, other personal information that is sometimes collected as well, such as details of your job, your gender, etc. will be treated as strictly confidential and will not be published unless explicitly agreed by you.

If you have recently submitted your details to me through email, or other avenues, and you would like to be removed from the research contributor's list, please send me an email stating that. I will remove your details from my files and send you an email confirming that I have done so. This does not include your non-personal details which will be used for the research that they were submitted in the first place.

Thank you,

Menelaos Tzouris.

Appendix VI – Semi-structured Interview Layout

As mentioned already the interviews had a semi-structured form. Below is presented a representative example of the content of the interview questions.

```
From: Tzouris,M
Sent: Tue 13/8/2002 8:35 pm
To: <undisclosed>
C.c.:
Subject: Free Software/Open Source Online Interview.


Dear <undisclosed>,


Following your response to the online questionnaire on the 25th of
July, I contact you again for the inline interview as you requested.


First and most of all I would like to thank you, once more, for you
contribution towards the completion of my research.


---------------------------------------------------------


Here starts the actual interview:...
You do not have to answer all questions. However please try to be
somehow extensive at the answers that you give.


Please note that you have answered some of the below already in the
questionnaire. However the interview is a means to elaborate more on
your thoughts.


Question 1:
-----------
As I can see from the questionnaire you filled in, you are
participating in Linux on Sega Dreamcast. Why did you choose to
participate to this particular project?


Question 2:
-----------
```

You are a PR Professional and a Hobbyist computer programmer. How does this two align? How did you learn programming?

Question 3:

You say that you expect a "bit of kudos" from your participation. However you rank "future career potentials" as your 3rd incentive for your contribution. Can you elaborate on this please?

Question 4:

Can you elaborate on your statement "wrote a c++ programme 9 years ago and released the binaries but not the source. Don't know why now." Did you realise later that you didn't do the right think? If so, did this influence your future as a programmer?

Question 5:

Which is the license that underlines the projects you contribute to (Linux on Sega Dreamcaster)? Have you studied it in depth?

Question 6:

You say that you abide to norms and rules of the community. Can you please elaborate on this?

Question 7:

What is your relationship with other contributors?

Question 8:

Whom do you consider your allies and whom your enemies in the FS/OSS "game"?

Question 9:

You say that you started contributing to FS/OSS "for fun and because it was a challenge". Is it still a challenge? What keeps you in the active in the FS/OSS.

```
Question 10:
-----------
Do you have the sense FS/OSS is a competitive or collaborative
environment? How would you back your answer?


Thanks again for your time,
Menelaos.


PS:
The first draft of my work is now ready and can be found at
http://www.geocities.com/tzmnlaos/oss/draft.html
Please fell free to comment on it, if you like. However, please be
informed that this is the first draft and that it hasn't been
properly proofread yet.



You can check the above URL to see how my research is progressing,
since I will start to incorporate in it the results from the
interviews. I will email you again with the URL of the final report.
```

Appendix VII – Interviews

Communities

Reputation concerns are obvious in the following interview citations:

> ```
> "I expect a little bit of kudos […] well you never know
> when those skills will be needed"
> ```
>
> <div align="right">(FS/OSS Contributor, mcm)</div>

> ```
> "The good thing about the community is that your status is
> based on your skills [...] Who cares if you're a dwarf or a
> goblin, as long as you contribute well to the project? It's
> all about respect, and if you prove your skills, you earn
> the respect [emphasis added]."
> ```
>
> <div align="right">(FS/OSS Contributor, LoneWolf)</div>

Motivational Psychology

Below the reader can find citations from participants' statements that can be categorised as intrinsic, explicit, both intrinsic and explicit, and idealistic motivation respectively:

Intrinsic motivation (internal factors)

> ```
> "Of course, you'll get people who want to change the world
> (like RMS) [Richard M. Stallman] but if you hear about what
> motivated him in the first place, it was about being
> frustrated in his goal of fixing a printer.  Most techies
> are obsessive about fixing things. If I come across
> something that's broken and obviously fixable, I pretty
> much have to fix it."
> ```
>
> <div align="right">(FS/OSS Contributor, Philip Hands)</div>

Extrinsic motivations (external rewards)

> ```
> "Someone offers something else I need in return for my
> effort/code (good oldtime bartering) "
> ```
>
> <div align="right">(FS/OSS Contributor, LoneWolf)</div>

Both intrinsic and extrinsic motivation

> "I do it because I like doing what I do. Sure, in a way the experience I gain when working on a certain project gives me skills that I may later use in a "normal" paid job, but that's more of a bonus than anything else. I live as long as I learn, and I'm planning on living for quite some time yet! ;-)"
>
> (FS/OSS Contributor, LoneWolf)

Idealistic motivation

> "I could be wrong or just unique, but I don't think contributing to Free/Open source software is driven by rewards. It's far more based on interest and a willingness to share. Sharing benefits us all in the long run - better software is the obvious result of it. It's also about honour, it's not right for me to use software created by others without contributing something back."
>
> (FS/OSS Contributor, LoneWolf)

Governance

One of the interview findings is that participants are carefully choosing the FS/OS software to which they will contribute, for example see the following statement made by an interviewee:

> "BTW another thing worth mentioning is that of licences. The stuff I do is ASF (Apache Software Licence) related. Its often referred to as a BSD-type licence. All the open source work I do, and the people I work with, all do ASF stuff too. I have nothing to do with GPL. Most people I know are the same. Reason? So that we can use the open source work as part of our commercial day jobs. So I can embed all the code I write under open source, inside my employers products."
>
> (FS/OSS Contributor, James Strachan)

Economics

using the categorisation of Lerner and Tirole (2001), interviewees' comments are presented in below:

Immediate benefits:

> "personal enjoyment, getting software for free that meets my needs, giving back after having received so much high-quality free software"
>
> (FS/OSS Contributor, hp)

> "Knowledge and experience, rewards I am already enjoying. As well as the pleasure of being involved in interesting things."
>
> (FS/OSS Contributor, Matt Rowland)

Immediate costs:

> "So pretty much everything I work on, I use as part of my day job*, while being paid to do something else. This leads to interesting & difficult time management issues :-) [...]*paid job or 'day job' as OS folks often refer to it"
>
> (FS/OSS Contributor, James Strachan)

Delayed benefits:

> "I've always hated the feeling that I'm rewriting a program that has been written before --- I think many programmers have this highly attuned form of laziness, and it offends us to be redoing things. Free Software means that does not need to happen."
>
> (FS/OSS Contributor, Philip Hands)

> "[Reward I am expecting:] perhaps a recognised name in the industry"
>
> (FS/OSS Contributor, tjr)

> "We all live in the real world and we all need to eat and sleep. I contribute to open-source as a hobby without expecting any direct pay-off in return. However, the experience gained from being involved in open source software will obviously help in the commercial world. A

> developer who does not spend time developing software will
> stagnate and their skills will erode. Developing software
> is a craft which is finely honed by experience - and is not
> a skill which one can learn in a classroom."
>
> (FS/OSS Contributor, atc)

Delayed costs:

> "Also, the cannon of Free Software programs has always
> reminded me of some sort of Darwinian evolutionary
> scenario. The fittest program tends to get the community
> mind share, and thus gets improved faster. If someone
> comes up with a better idea, people are quite likely to
> switch. This seems likely to produce the best possible
> solution over time, even if the route is sometimes a little
> erratic[26]."
>
> (FS/OSS Contributor, Philip Hands)

---

[26] Fads attract the attention of many programmers. However fads may wore out, and contributors may find themselves involved in a project or a project's fork only to be proven that it was not worth the effort, in the future.

References

Barbrook, R. (1998) *The Hi-Tech Gift Economy* First Monday, 3(12) Last accessed: June 2002 Last updated: 7th December 1998 Address: http://www.firstmonday.dk/issues/issue3_12/barbrook/.

Behlendorf, B. (1999) "Open Source as a Business Strategy" in *Open Sources - Voices from the Open Source Revolution*, (DiBona, C., S. Ockman and M. Stone eds) O'Reilly & Associates, Inc, Sebastopol, CA.

Bergquist, M. and J. Ljungberg (2001) "The Power of Gifts: Organizing Social Relationships in Open Source Communities", *Information Systems Journal,* **11** p. 305–320.

Bergquist, M., J. Ljungberg and U. Lundh-Snis (2001) "Practising Peer Review in Organizations: A Qualifier for Knowledge Dissemination and Legitimization", *Journal of Information Technology,* **16** p. 99–112.

Bezroukov, N. (1999) *Open Source Software Development as a Special Type of Academic Research (Critique of Vulgar Raymondism)* First Monday, 04(10) Last accessed: June 2002 Last updated: 4 October 1999 Address: http://www.firstmonday.dk/issues/issue4_10/bezroukov/.

Blakely, E. (1989) "Theoretical Approaches for a Global Community" in *Community Development in Perspective*, (Christenson, J. A. and J. J. W. Robinson eds) Iowa State University Press, Ames, IA.

Braa, K. and R. Vidgen (1997) "An Information Systems Research Framework for the Organizational Laboratory" in *Computers and Design in Context*, (Kyng, M. and L. Mathiassen eds) MIT Press, Cambridge, MA., pp. 381-401.

Brooks, F. (1975) *The Mythical Man Month: Essays on Software Engineering,* Addison-Wesley,

Bruner, J. S. (1990) *Acts of Meaning,* Harvard University Press, Cambridge, MA.

Bruner, J. S., J. J. Goodnow and G. A. Austin (1986) *A Study of Thinking,* Transaction Books, New Brunswick, NJ.

Burrell, G. and G. Morgan (1979) *Sociological Paradigms and Organisational Analysis,* Heinemann, London.

Butler, B., L. Sproull, S. Kiesler and R. Kraut (2002) *Community Effort in Online Groups: Who Does the Work and Why?* Forthcoming in: Leadership at a Distance, S. Weisband & L. Atwater (Eds.). Last accessed: July 2002 Last updated: February 2002 Address: http://opensource.mit.edu/papers/butler.pdf.

Castells, M. (1996) *The Rise of the Network Society,* Blackwell Publishers Inc., Cambridge, Massachusetts.

Ciborra, C. and G. F. Lanzara (1994) "Formative Contexts and Information Technology: Understanding the Dynamics of Innovation Is Organizations", *Accounting, Management and Information Technology,* **04 (02),** pp. 61-86.

Ciborra, C. U. and R. Andreu (2001) "Sharing Knowledge across Boundaries", *Journal of Information Technology,* **16 (02),** pp. 73-81.

Cornford, T. and S. Smithson (1996) *Project Research in Infrmation Systems - a Student's Guide,* MacMillan Press ltd, London, UK.

Dafermos, G. N. (2001a) *Management and Virtual Decentralised Networks: The Linux Project* First Monday, 6(11) Last accessed: July 2002 Last updated: 5 November 2001 Address: http://www.firstmonday.dk/issues/issue6_11/dafermos/.

Dafermos, G. N. (2001b) *Management and Virtual Decentralized Networks: The Linux Project* Durham Business School Last accessed: August 2002 Last updated: 09 January 2002 Address: http://opensource.mit.edu/papers/dafermoslinux.pdf.

Dalle, J.-M. and N. Jullien (2001) *Libre' Software: Turning Fads into Institutions?* Last accessed: 5th July 2002 Last updated: Address: http://opensource.mit.edu/papers/Libre-Software.pdf.

Deci, E. (1975) *Intrinsic Motivations,* Plenum Press, New York, NY.

Dempsey, B. J., D. Weiss, P. Jones and J. Greenberg (2002) "Who Is an Open Source Software Developer? - Profiling a Community of Linux Developers", *Communication of the ACM,* **45 (02)**.

Dinkelacker, J. and P. K. Garg (2001) "Corporate Source: Applying Open Source Concepts to a Corporate Environment". in *Making Sense of the Bazaar: 1st Workshop on Open Source Software Engineering was held on May 15, 2001 as part of The 23rd International Conference on Software Engineering (ICSE 2001),* Toronto, Ontario, Canada,

Edwards, K. (2000) *When Beggars Become Choosers* First Monday, 5(10) Last accessed: July 2002 Last updated: 2 October 2000 Address: http://www.firstmonday.dk/issues/issue5_10/edwards/.

Edwards, K. (2001) *Epistemic Communities, Situated Learning and Open Source Software Development* Department of Manufacturing Engineering and Management, Technical University of Denmark, Denmark Last accessed: 28 June 2002 Last updated: 11June 2001 Address: http://opensource.mit.edu/papers/kasperedwards-ec.pdf.

Faraj, S. and M. M. Wasko (2001) *The Web of Knowledge: An Investigation of Knowledge Exchange in Networks of Practice*, MIT - Open Source Research Community, Last accessed: 13 February 2002 Last updated: Address: http://opensource.mit.edu/papers/Farajwasko.pdf.

Feller, J. and B. Fitzgerald (2000) "A Framework Analysis of the Open Source Software Development Paradigm". in *Proceedings of the twenty first international conference on Information systems,* Brisbane, Queensland, Australia,

Feller, J. and B. Fitzgerald (2002) *Understanding Open Source Software Development,* Addison Wesley,

Forge, S. (2000) "Open Source: The Economics of Giving Away Stuff, and Software as a Political Statement", *inform - the journal of policy, regulation and strategy for telecommunications information and media,* **02 (01),** pp. 5-7.

Franck, E. and C. Jungwirth (2002) "Reconciling Investors and Donators - the Governance Structure of Open Source", *University of Zurich, Working Paper Series*.

Franke, N. and S. Shah (2001) *How Communities Support Innovative Activities: An Exploration of Assistance and Sharing among Innovative Users of Sporting Equipment* Sloan Working Paper #4164 Last accessed: 6 March 2002 Last updated: August 2001 Address: http://opensource.mit.edu/papers/frankeshah.pdf.

Frey, B. S. (1997) *Not Just for Money: An Economic Theory of Personal Motivation,* Edward Elgar Publishing Limited, Cheltenham.

Frey, B. S. and R. Jegen (2001) "Motivation Crowding Theory: A Survey of Empirical Evidence", *Journal of Economic Surveys: forthcoming.*

Frey, B. S. and M. Osterloh (2000) "Motivation, Knowledge Transfer, and Organizational Forms", *Organization Science,* **11** pp. 538-550.

Galliers, R. D. (1992) "Choosing Appropriate Information Systems Research Methods" in *Information Systems Research: Issues, Methods and Practical Guidelines*, (Galliers, R. D. ed.) Blackwell, Oxford.

Garzarelli, G. (2002) *Open Source Software and the Economics of Organization* Last accessed: August 2002 Last updated: April 2002 Address: http://opensource.mit.edu/papers/garzarelli.pdf.

Ghosh, R. A. (1998a) *Cooking Pot Markets: An Economic Model for the Trade in Free Goods and Services on the Internet* First Monday, 3(3) Last accessed: July 2002 Last updated: 2nd March 1998 Address: http://www.firstmonday.dk/issues/issue3_3/ghosh/index.html.

Ghosh, R. A. (1998b) *Interview with Linus Torvalds: What Motivates Free Software Developers?* FirstMonday, 3(3) Last accessed: 3 July 2002 Last updated: 2 March 1998 Address: http://www.firstmonday.dk/issues/issue3_3/torvalds/index.html.

Giddens, A. (1976) *New Rules of Sociological Method : A Positive Critique of Interpretative Sociologies,* Hutchinson, London.

Haas, P. M. (1992) "Introduction: Epistemic Communities and International Policy Coordination", *International Organization,* **46 (01)**.

Halloween_Document (1998) *Halloween Document I* Last accessed: 11 August 2002 Last updated: Address: http://www.opensource.org/halloween/.

Hamerly, J., T. Paquin and S. Walton (1999) "Freeing the Source - the Story of Mozilla" in *Open Sources - Voices from the Open Source Revolution*, (Stone, M. ed.) O'Reilly & Associates, Inc, Sebastopol, CA.

Hannemyr, G. (1999) *Technology and Pleasure: Considering Hacking Constructive* First Monday, 4(2) Last accessed: July 2002 Last updated: 1st February 1999 Address: http://www.firstmonday.dk/issues/issue4_2/gisle/.

Hansmann, H. B. (1980) "The Role of Nonprofit Enterprise", *Yale Law Journal,* **89** pp. 835-901.

Hansmann, H. B. (1981) "Nonprofit Enterprise in the Performing Arts", *The Bell Journal of Economics,* **12 (02),** pp. 341-361.

Hansmann, H. B. (1987) "Economic Theories of Nonprofit Organization" in *The Nonprofit Sector: A Research Handbook*, (Powell, W. W. ed.) Yale University Press., New Haven, CT, pp. 27-42.

Hars, A. and S. Ou (2001) "Working for Free? – Motivations of Participating in Open Source Projects". in *Proceedings of the 34th Hawaii International Conference on System Sciences,*

Hecker, F. (1999) *Mozilla at One: A Look Back and Ahead* Last accessed: 3 July 2002 Last updated: July 5, 2000 Address: http://www.mozilla.org/mozilla-at-one.html.

Herzberg, F. (1982) *The Managerial Choice: To Be Efficient and to Be Human (Rev. Ed.),* Olympus, Salt Lake City, UT.

Hinds, P. and S. Kiesler (1995) "Communication across Boundaries: Work, Structure, and Use of Communication Technologies in a Large Organization", *Organization Science,* **6** pp. 373-393.

Hoffman, E. (1988) *The Right to Be Human: A Biography of Abraham Maslow,* J. P. Tarcher, Los Angeles.

IEEE (1990) "Ieee Standard Glossary of Software Engineering Terminology" in *Ieee Std 610.12-1990 (Revision and Redesignation of Ieee Std 729-1983),* IEEE Computer Society.

Jørgensen, N. (2001) "Putting It All in the Trunk: Incremental Software Development in the Freebsd Open Source Project", *Information Systems Journal,* **11 (04),** pp. 321-336.

Kallinikos, J. (2001) *The Age of Flexibility - Managing Organizations and Technology,* Academia Adacta, Lund, Sweden.

Kaplan, B. and J. A. Maxwell (1994) "Qualitative Research Methods for Evaluating Computer Information Systems" in *Evaluating Health Care Information Systems: Methods and Applications*, (Anderson, J. G., C. E. Aydin and S. J. Jay eds) Sage, Thousand Oaks, CA.

Kelty, C. M. (2001) *Free Software/Free Science* First Monday, 6(12) Last accessed: July 2002 Last updated: 3 December 2001 Address: http://www.firstmonday.dk/issues/issue6_12/kelty/.

Klandermans, B. (1997) *The Social Psychology of Protest,* Blackwell, New York.

Kogut, B. and A. Metiu (2001) "Open-Source Software Development and Distributed Innovation", *Oxford Review of Economic Policy,* **17 (02),** pp. 248-264.

Kuwabara, K. (2000) *Linux: A Bazaar at the Edge of Chaos* First Monday, 5(3) Last accessed: August 2002 Last updated: 6 March 2000 Address: http://www.firstmonday.dk/issues/issue5_3/kuwabara/.

Lakhani, K. R. and E. von Hippel (2002) *How Open Source Software Works: "Free" User-to-User Assistance* MIT Sloan School of Management, Cambridge, MA, USA Last accessed: August 2002 Last updated: June 2002 Address: http://opensource.mit.edu/papers/lakhanivonhippelusersupport.pdf.

Lakoff, G. and M. Johnson (1981) *Metaphors We Live By,* University of Chicago Press, Chicago.

Lancashire, D. (2001) *Code, Culture and Cash: The Fading Altruism of Open Source Development* First Monday, 6(12) Last accessed: 10 August 2002 Last updated: 3 December 2001 Address: http://www.firstmonday.dk/issues/issue6_12/lancashire/.

Lave, J. and E. Wenger (1991) *Situated Learning : Legitimate Peripheral Participation (Learning in Doing : Social, Cognitive and Computational Perspectives),*

Leatherdale, W. H. (1974) *The Role of Analogy, Model and Metaphor in Science,* North-Holland, Amsterdam.

Lee, A. S. (1999) "Researching M.I.S." in *Rethinking Management Information Systems*, (Galliers, R. D. and W. L. Currie eds) Oxford University Press, Oxford, pp. 7-27.

Lerner, J. and J. Tirole (2000) *The Simple Economics of Open Source* Last accessed: June 2002 Last updated: March 2000 Address: http://opensource.mit.edu/papers/Josh%20Lerner%20and%20Jean%20Triole%20-%20The%20Simple%20Economics%20of%20Open%20Source.pdf.

Lerner, J. and J. Tirole (2001) "Some Simple Economics of Open Source", *Journal of Industrial Economics,* **50 (02),** pp. 197-234.

Lessig, L. (1999) *Code : And Other Laws of Cyberspace,* Basic Books, New York, N.Y.

Lindeman, E. C. (1962) "Community" in *Encyclopaedia of the Social Sciences Iii*, pp. 102-104.

Ljungberg, J. (2000) "Open Source Movements as a Model for Organising", *European Journal of Information Systems,* **2000** p. 208–216.

Madanmohan, T. R. and S. Navelkar (2002) *Roles and Knowledge Management in Online Technology Communities: An Ethnography Study* Indian Institute of Management-Bangalore, Digital Last accessed: Julu 2002 Last updated: June 2002 Address: http://opensource.mit.edu/papers/madanmohan2.pdf.

Maisel, R. and C. H. Persell (1995) *How Sampling Works,* Pine Forge Press, Thousand Oaks, CA.

Mann, C. and F. Stewart (2000) *Internet Communication and Qualitative Research - a Handbook for Researching Online,* SAGE Publishing, London.

Marshall, G. (Ed.) (1998) *A Dictionary of Sociology,* (2nd) Oxford University Press, Oxford.

Maslow, A. H. (1970) *Motivation and Personality,* (2nd) Harper and Row, New York.

Masum, H. (2001) "Reputation Layers for Open-Source Development". in *Making Sense of the Bazaar: 1st Workshop on Open Source Software Engineering was held on May 15, 2001 as part of The 23rd International Conference on Software Engineering (ICSE 2001),* Toronto, Ontario, Canada,

Metiu, A. and B. Kogut (2001) *Distributed Knowledge and the Global Organization of Software Development* University of Pennsylvania Last accessed: July 2002 Last updated: February 2001 Address: http://opensource.mit.edu/papers/kogut1.pdf.

Mockus, A., R. T. Fielding and J. Herbsleb (2000) "A Case Study of Open Source Software Development: The Apache Server". in *Proceedings of the 22nd international conference on Software engineering,* pp. 263 - 272, ACM Press New York, NY, USA, Limerick, Ireland, June 2000.

Moody, G. (2001) *Rebel Code - Linux and the Open Source Revolution,* Penguine Books Ltd, London.

Moon, J. Y. and L. Sproull (2000) *Essence of Distributed Work: The Case of the Linux Kernel* First Monday, 5(11) Last accessed: August 2002 Last updated: 5 November 2000 Address: http://www.firstmonday.dk/issues/issue5_11/moon/.

Myers, M. D. (1999) *Qualitative Research in Information Systems* IS World Net Last accessed: 29 July 2002 Last updated: unknown Address: http://www.qual.auckland.ac.nz/.

Netcraft (2002) *Netcraft Web Server Survey* Last accessed: 10 August 2002 Last updated: July 2002 Address: http://www.netcraft.com/survey/.

Neus, A. (2001) *Managing Information Quality in Virtual Communities of Practice*, MIT - Open Source Research Community, Last accessed: 14 Feb. 2002 Last updated: Address: http://opensource.mit.edu/papers/neus.pdf.

Nonaka, I. (1994) "A Dynamic Theory of Organizational Knowledge Creation", *Organization Science,* **05 (01),** pp. 14-37.

O'Reilly, T. (1999) "Lessons from Open-Source Software Development", *Communications of the ACM,* **42 (04)**.

Orlikowski, W. J. (2000) "Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations", *Organization Science,* **11 (04),** pp. 404-428.

Orlikowski, W. J. (2002) *Knowing in Practice: Enacting a Collective Capability in Distributed Organizing* Forthcoming: Organization Science 2002 Last accessed: July 2002 Last updated: May 2002 Address: http://opensource.mit.edu/papers/orlikowski.pdf.

Orlikowski, W. J. and J. J. Baroudi (1991) "Studying Information Technology in Organizations: Research Approaches and Assumptions", *Information Systems Research,* **2** pp. 1-28.

OSI (1998) *Open Source Initiative Launch Announcement* Last accessed: 28 July 2002 Last updated: 22 November 1998 Address: http://www.opensource.org/pressreleases/osi-launch.php.

Porter, M. E. (2001) "Strategy and the Internet", *Harvard Business Review,* **March 2001** pp. 63-78.

Porter, M. E. and V. E. Millar (1985) "How Information Gives You Competitive Adbantage", *Harvard Business Review,* **July - August 1985** pp. 149-160.

Raymond, E. S. (1996) *The New Hacker's Dictionary,* The MIT Press,

Raymond, E. S. (1998a) *The Cathedral and the Bazaar* First Monday, 3(3) Last accessed: July 2002 Last updated: 2nd March 1998 Address: http://www.firstmonday.dk/issues/issue3_3/raymond/index.html.

Raymond, E. S. (1998b) *Homesteading the Noosphere* First Monday, 3(10) Last accessed: July 2002 Last updated: 5 October 1998 Address: http://www.firstmonday.org/issues/issue3_10/raymond/.

Raymond, E. S. (1999) *The Cathedral and the Bazaar,* O' Reilly, Sebastopol, CA.

Raymond, E. S. (2000) *Homesteading the Noosphere: An Introductory Contradiction* Last accessed: 22 June 2002 Last updated: 24 August 2000 Address: http://tuxedo.org/~esr/writings/cathedral-bazaar/homesteading/.

RFC-977 *Network News Transfer Protocol: A Proposed Standard for the Stream-Based Transmission of News* Last accessed: Last updated: Address: http://www.ietf.org/rfc/rfc0977.txt.

RFC-1855 *Netiquette Guidelines* Last accessed: Last updated: Address: http://www.ietf.org/rfc/rfc1855.txt.

Rheingold, H. (1994) *The Virtual Community : Homesteading on the Electronic Frontier,* HarperPerennial, New York, N.Y.

Scacchi, W. (2002) *Understanding the Requirements for Developing Open Source Software Systems* IEE Proceedings--Software, 149(1), 24-39 Last accessed: April 2002 Last updated: December 2001 Address: http://opensource.mit.edu/papers/Scacchi.pdf.

Schein, E. H. (1992) *Organizational Culture and Leadership,* (2nd) Jossey-Bass, San Francisco.

Söderberg, J. (2002) *Copyleft Vs. Copyright: A Marxist Critique* First Monday, 7(3) Last accessed: August 2002 Last updated: 4 March 2002 Address: http://www.firstmonday.dk/issues/issue7_3/soderberg/.

Stallman, R. (1985) *Gnu Manifesto* Last accessed: 25th June 2002 Last updated: Address: http://www.gnu.ai.mit.edu/gnu/manifesto.html.

Stallman, R. (1999) "The Gnu Operating System and the Free Software Movement" in *Open Sources: Voices from the Open Source Revolution*, (DiBona, C., S. Ockman and M. Stone eds) O'Reilly & Associates, Inc, Sebastopol, CA.

Syme, S. and L. J. Camp (2001) *Code as Governance, the Governance of Code* Last accessed: 13 August 2002 Last updated: Address: http://ksgnotes1.harvard.edu/Research/wpaper.nsf/rwp/RWP01-014/$File/rwp01_014_camp.pdf.

Tiemann, M. (1999) "Future of Cygnus Solutions: An Enterpreneur's Account" in *Open Sources - Voices from the Open Source Revolution*, (DiBona, C., S. Ockman and M. Stone eds) O'Reilly & Associates, Inc, Sebastopol, CA.

Torvalds, L. (1999) "The Linux Edge" in *Open Sources: Voices from the Open Source Revolution*, (DiBona, C., S. Ockman and M. Stone eds) O'Reilly & Associates, Inc, Sebastopol, CA.

Torvalds, L. and D. Diamond (2001) *Just for Fun: The Story of an Accidental Revolutionary,* Harper Business,

Tsoukas, H. (1989) "The Validity of Idiographic Research Explanations (in Special Forum on Theory Building)", *The Academy of Management Review,* **14 (04),** pp. 551-561.

Tsoukas, H. (1991) "The Missing Link: A Transformational View of Metaphors in Organizational Science", *The Academy of Management Review,* **16 (03),** pp. 566-585.

Tuomi, I. (2001) *Internet, Innovation, and Open Source: Actors in the Network* First Monday Volume 6, Number 1 Last accessed: July 2002 Last updated: January 2001 Address: http://www.firstmonday.dk/issues/issue6_1/tuomi/.

Tzouris, M. (2002) *Rethinking Communities of Practice in the Networked Organisation - an Organisational Learning Study* Last accessed: 20 June 2002 Last updated: 15 April 2002 Address: http://www.geocities.com/tzmnlaos/cop/cop.pdf.

Unger, R. M. (1987) *False Necessity,* Cambridge University Press, Cambridge, UK.

von Hippel, E. (2001) *Open Source Shows the Way: Innovation by and for Users – No Manufacturer Required!* Sloan Management Review, Summer, 2001 Last accessed: July 2002 Last updated: March 2001 Address: http://opensource.mit.edu/papers/evhippel-osuserinnovation.pdf.

von Hippel, E. (2002) *Horizontal Innovation Networks - by and for Users* MIT Sloan School of Management Last accessed: July 2002 Last updated: June 2002 Address: http://opensource.mit.edu/papers/vonhippel3.pdf.

von Hippel, E. and G. von Krogh (2002) *Exploring the Open Source Software Phenomenon: Issues for Organization Science* Last accessed: August 2002 Last updated: April 2002 Address: http://opensource.mit.edu/papers/hippelkrogh.pdf.

Wasko, M. M. and S. Faraj (2000) ""It Is What One Does": Why People Participate and Help Others in Electronic Communities of Practice", *Journal of Strategic Information Systems,* **2000** pp. 155-173.

Weick, K. E. (1989) "Theory Construction as Disciplined Imagination", *Academy of Management Review,* **14 (04),** pp. 516-531.

Weinberg, G. M. (1971) *The Psychology of Computer Programming,* Van Nostrand, New York.

Zhang, W. and J. Storck (2001) *Peripheral Members in Online Communities in Proceedings of the Seventh Americas Conference on Information Systems (Amcis 2001), Boston, Massachusetts* Last accessed: 6 March 2002 Last updated: Address: http://opensource.mit.edu/papers/zhang.pdf.

Zuboff, S. (1988) *In the Age of the Smart Machine : The Future of Work and Power,* Basic Books, New York.