Dissertation

Political Motives of Developers for Collaboration on GNU/Linux

submitted for the degree of

MA (Globalization and Communications) 2003/2004

by

Tobias Escher

Centre for Mass Communication Research

University of Leicester

August 2004

17791 words

(plus 1266 words bibliography)

Abstract

This paper examines to what degree developers of the collaboratively produced computer operating system GNU/Linux are politically motivated for their contributions. It first states that software is politically relevant. It then goes on to argue for the political significance of Free Software/Open Source Software (FS/OSS) and discusses the developers ambivalent attitude towards a politicisation of FS/OSS. Centrepiece is a survey carried out with 85 GNU/Linux developers that showed that most of them are conscious of the social relevance of FS/OSS and that their engagement is of a deliberately political nature.

Notes

An online version of this document in the Portable Document Format (pdf) is available at http://www.chronovault.net/websites/tobi/researchID/ together with the questionnaire and the raw data of this survey. User name as well as password to access the site is ***guest***. For more information see Appendix A.6 Online Resources for this Survey.

This work could not manage without the use of some specific vocabulary. For those readers not familiar with the concepts of software programming and free software there is an explanation of some important terms and concepts in the appendix to this work.

Acknowledgements

I want to take the opportunity to express my gratefulness to all the persons who gave me support during this project and provided valuable thoughts and discussions. First of all I want to mention my supervisor Dr. Gillian Youngs at the Centre for Mass Communication Research at the University of Leicester. Furthermore this survey would not have been possible without the great responsiveness of the members of the Free Software / Open Source Software community. Additionally I send a thank you to my pre-testers, namely Björn, Marc, Frank, Christian and Tobias for helping to improve the questionnaire and to fix some bugs in the software. My proofreader Rita shall not be left out of this list who managed the extraordinary task to provide precise advice even in the most stressful situations. I would also like to mention the German Academic Exchange Service (DAAD) without whose financial support this year at the University of Leicester would have been impossible. Last, but by no means least, comes a special word of gratitude to the ones whose support was and still is key to everything and who I better do not name as I know that his would cause them embarrassment.

Table of Contents

Abstract.....i

Notes.....i

Acknowledgements.....ii

Table of Contents.....iii

Table of Illustrations.....v

Table of Graphics.....v

Acronyms.....vii

Chapter 1: Introduction.....1

1.1 Free Software and GNU/Linux.....1

1.1.1 Free Software.....2

1.1.2 GNU and Linux.....3

Chapter 2: The Politics of FS/OSS.....5

2.1 Software as Political Issue.....6

2.2 The Political Significance of Free Software / Open Source.....9

2.3 FS/OSS-Developers as Political Actors.....11

2.3.1 Free Software Foundation versus Open Source Initiative.....12

2.3.2 The Political Developer: Contrasting Findings.....14

2.4 The Problems of Determining FS/OSS Politicisation.....18

Chapter 3: Methodology.....21

3.1 Research Questions.....21

3.2 Research Method.....21

3.2.1 Instrument.....22

3.2.2 Questionnaire Content.....24

3.2.3 Sample.....25

3.2.3.1 Sampling.....25

3.2.3.2 Sample Demography.....27

3.2.3.3 Sample FS/OSS Involvement.....31

Chapter 4: Findings of Survey.....35

4.1 Assessment and Motivation of Developers.....35

4.1.1 High Overall Politicisation in FS/OSS Community.....35

4.1.2 “Free Software is Politics, Open Source is Economics”.....36

4.1.3 Politics just One of Many Motivations.....39

4.2 Political Participation Offline and Online.....42

4.2.1 High General Political Interest and Activity.....42

4.2.2 Relation of Online and Offline Political Behaviour.....44

4.3 Characterising Political Developers.....47

4.3.1 Hacker Values as Main Characteristic for the Politically Motivated Developers.....47

4.3.2 Little Observable Differences between the Movements.....50

4.3.3 No Real Differences for Low/Medium Income Countries.....54

4.3.4 Paid Developers no Less Motivated than Volunteers.....55

4.4 Political Views of FS/OSS Developers.....56

Chapter 5: Conclusion.....60

Bibliography.....64

Appendix.....70

A.1 Explanation of Computer Terms.....70

A.2 Why Access to Source Code Is Important.....73

A.3 Questionnaire.....74

A.4 Invitation Letter.....84

A.5 Technical Issues Concerning the Online Questionnaire.....85

A.6 Online Resources for this Survey.....85

A.7 Internet Addresses of Mentioned Projects.....86

Table of Illustrations

Illustration 1: Motivation of FS/OSS developers. Findings of FLOSS.....16
Illustration 2: Screen Shot Questionnaire.....23
Illustration 3: Compilation Steps.....73

Table of Graphics

Graph 1: Countries of Residence. (82).....27
Graph 2: Continents. (82).....28
Graph 3: Area. (81).....28
Graph 4: Age. (82).....29
Graph 5: Education. (82).....29
Graph 6: Profession. (77).....30
Graph 7: Work Status. (79).....30
Graph 8: Years of FS/OSS Development. (85).....31
Graph 9: Development Projects. (81).....31
Graph 10: Project Initiative. (78).....32
Graph 11: Contribution of Translations. (77).....32
Graph 12: Paid for FS/OSS. (82).....32
Graph 13: Development at Work. (62).....33
Graph 14: FS/OSS Contacts. (83).....33
Graph 15: User OS. (80).....34
Graph 16: Political Motivation. (84).....35
Graph 17: Importance of Political Goals. (84).....35
Graph 18: FS/OSS Politicisation. (83).....36
Graph 19: FS/OSS Differences. (85).....37
Graph 20: FS/OSS Membership. (85).....38
Graph 21: Personal Motives for Contribution.....40
Graph 22: Community Motives for Contribution.....41
Graph 23: IT-Divide Motives for Contribution.....41

Graph 24: Political News Access. (84).....43
Graph 25: Political Interest. (85).....43
Graph 26: Political Action. (83/80).....43
Graph 27: FS/OSS Publicity. (82).....44
Graph 28: Discussion Contributions. (85).....44
Graph 29: Political Goals by Political Interest. (84).....45
Graph 30: Political Motivation by Political Interest. (84).....45
Graph 31: Political Action by Political Interest. (83/80).....45
Graph 32: Income Status by Political Interest. (82).....46
Graph 33: Political Action by Income Status. (81/77).....46
Graph 34: Hackers. (78).....47
Graph 35: Political Goals by Hacker. (77).....48
Graph 36: Political Motivation by Hacker. (78).....48
Graph 37: Hacker Share by Politicisation. (77).....48
Graph 38: Political Action by Hacker. (77/74).....49
Graph 39: Movement by Hacker. (78).....50
Graph 40: Differences FS/OSS by Hacker. (78).....50
Graph 41: Hackers by Community. (78).....50
Graph 42: Movements by Political Goals. (84).....51
Graph 43: Movements by Political Motivation. (84).....51
Graph 44: Movement by Political Interest. (85).....51
Graph 45: Political Action by Movement. (83/80).....52
Graph 46: Movement by Politicisation. (83).....53
Graph 47: Motives by Country Income Status.....55
Graph 48: Political Views.....56
Graph 49: Internet Governance. (75).....58
Graph 50: FS/OSS Hierarchies. (82).....59

Acronyms

ARPA	Advanced Research Projects Agency
DRM	Digital Rights Management
GNU	Gnu is Not Unix
GPL	General Public License
FAQ	Frequently Asked Questions
FS	Free Software
FSF	Free Software Foundation
HIC	High Income Country
ICT	Information and Communication Technology
IT	Information Technology
MS	Microsoft
LMIC	Low/Medium Income Country
OS	Operating System
OSI	Open Source Initiative
OSS	Open Source Software
WWW	World Wide Web

Chapter 1: Introduction

This paper is concerned with computer programmers who develop GNU/Linux (often named simply Linux) which is a computer operating system, the basic piece of software that makes a computer work. GNU/Linux is peculiar because it is Free or Open Source Software (FS/OSS) produced by thousands of volunteers around the world (although in recent times also increasingly by paid programmers). Free Software in this case does not mean that the software comes free of charge (although it often does) but that everyone is free to copy, modify and redistribute it, something that is made possible because the source code of the operating system is available to everyone. It is this fact that marks the difference between GNU/Linux as FS/OSS and for example Microsoft Windows as proprietary software.

The main question of this paper is whether the developers of GNU/Linux attribute FS/OSS a political significance and if so, whether they see their programming contributions as a deliberate act of political action or as an expression of a social idea. This question is considered to be important as this paper argues that software due to its nature and function is of crucial political relevance and that computer programmers therefore have considerable responsibility (Chapter 2.1). The case is made that FS/OSS is of particular political relevance because it challenges established assumptions about production and property (Chapter 2.2). While FS/OSS and GNU/Linux in particular are already widely deployed for political reasons, it will be discussed that the developers' perception of this is ambivalent as well as the findings of previous studies concerning this topic (Chapter 2.3). It is one of the aims of this paper to look for explanations for this contradictory findings (Chapter 2.4). Finally, to answer the main research question of this paper an online survey was carried out (Chapter 3) the results of which will be discussed in Chapter 4.

1.1 Free Software and GNU/Linux

Before exploring in detail the political ideas attached to the operating system GNU/Linux and the motivations of its developers, it seems expedient to make the reader familiar with the idea of FS/OSS and GNU/Linux in particular.

1.1.1 Free Software

First of all one has to understand that modern computer programs are written in a high-level programming language that is relatively easy to understand for humans but useless to computers. In order to make it work on a computer it has to be translated into a low-level language (called binaries) that is understandable for the computer but barely for humans. In order to manipulate an existing piece of software, a computer programmer needs to have access to this high-level version of the software, called source code (see Appendix A.2 for more information). Nowadays software that is bought is usually proprietary which means it can only be obtained in binary format without the source code (which is a closely guarded secret) and often with further restriction that for example prohibit to make further copies of the software. By the time computer programming started around 1960 though, computer programmers used to share their programs freely among each other. They even formed a kind of subculture, referring to themselves as hackers (someone who makes a clever improvement or unintended use of any kind of device) and adhering to some common values that were later written down as the Hacker Ethic (Levy, 1994). It promoted values such as freedom of information, open systems and decentralization.

Richard M. Stallman was one of these hackers and after having experienced the decline of the hacker culture due to the commercialisation of computers, he created the concept of free software in order to overcome the restrictions of proprietary software. Free software describes software that comes with the author's permission for the user to use the program for any purpose, to study how the program works, to redistribute copies of it and to improve the program and release the improvements to the public. So free software does not mean that the software has to be free of charge (although a lot of it is) but that it allows the user certain freedoms. As Stallman (1996: 41) puts it: *"free as in 'free speech', not as in 'free beer'"*. To promote this idea, Stallman founded the Free Software Foundation (FSF)[i] in 1985. To make it easy for other computer programmers to grant their software these freedoms, Stallman sat up a special license, the GNU General Public License (GPL) that applied to a piece of software makes it free software. In effect, Stallman used the copyright law to reverse it.

1.1.2 GNU and Linux

In order to provide people with the option to fully renounce of proprietary software Stallman started the GNU project in 1984. GNU is a recursive acronym for "GNU is not Unix" and was meant to create a free software alternative to a then popular operating system (OS) Unix (for a full account of the GNU project see Stallman, 2002a). An operating system is the very basic software that every computer needs in order to perform the most basic tasks like access the hardware in the computer. The most popular OS nowadays is Windows from Microsoft in its different variants. While GNU was very successful in creating the important tools for an operating system, by the early nineties it still lacked the core component of an OS, a kernel. This gap was filled in 1991 with the creation of a kernel by Finish computer science student Linus Torvalds. This happened independent of the GNU project but with the software it provided (for a full account of the history of GNU/Linux see Moody, 2001). The kernel and the GNU software represent a fully functioning free software OS, today commonly referred to as Linux although to acknowledge the big contributions of the GNU project it should be named GNU/Linux and that is the term that will be used throughout this work.

The revolutionary thing about the Linux kernel was the way in which it was developed. The kernel was and still is programmed cooperatively by developers throughout the world led by Torvalds. Eric S. Raymond (2000) refers to it as the 'bazaar'-style of production. It basically means making your program available to everyone and inviting contributions of them to further develop the software together. This was made possible by the thriving Internet and the openness of the source code, ensured by Stallman's GPL which was applied to the Linux kernel. While the GPL enabled this mode of development, it was not really used for free software development until Linus Torvalds' kernel. As Castells (2001: 45) argues, Richard M. Stallman and his fellows did not realize the full potential of networks for the production of software:

> *"Only a network of hundreds, thousands of brains working cooperatively, with spontaneous division of labor, and loose, but effective coordination, could accomplish the extraordinary task of creating an operating system able to handle the complexity of increasingly powerful computers interacting via the Internet."*

Under the name Open Source this mode of development has gained considerable momentum, not only used to produce software but also applied for other purposes like content creation (see for example Wikipedia[ii], a collaboratively produced free encyclopaedia on the Internet). Furthermore, in recent years a number of less strict alternatives to Stallman's GPL were created – so called Open Source licenses. While all licenses – Free Software as well as Open Source - have in common that they demand the free availability of the source code, the objectives of the licenses can be quite different as will be discussed later. Throughout this work, the term FS/OSS is used to refer to software the source code of which is available and that allows for collaborative production.

Concerning the acceptance of GNU/Linux today, it is difficult to get reliable numbers on current GNU/Linux users as the OS can be obtained mostly free of charge via the Internet and therefore sales figures do not have much expressiveness. An estimate by the community itself is about 18 million users (Alvestrand, 2004). However, GNU/Linux is said to become more and more a threat to Microsoft's operating systems as big computer companies such as IBM and Sun are starting to sell their products with GNU/Linux or huge Internet companies like Google and Amazon are running their sites on the free software OS. While still not strong in the market of home and office PCs, in the market of servers the free operating system has earned already a considerable market share that is estimated to be at around 23% with MS Windows having about 55% (IDC, 2003).

Chapter 2: The Politics of FS/OSS

Before taking a closer look at political explanations of developers' engagement with FS/OSS, it seems appropriate to define the meaning of the term politics in the context of this work. In reference to Heywood's (2002: 21-22) definition, politics is understood very broadly as "*the activity through which people make, preserve and amend the general rules under which they live.*" Through this approach to politics the analysis is not limited to issues concerning the government but brings into the picture other areas such as the economy or the social organisation of people as well.

This project focuses on the political ideas and actions of computer programmers with special reference to software politics. Following the definition of politics proposed in this text, political ideas are defined as any ideas about the kind of *general rules* under which people want to live, that means any desire for how they should live and interact with each other and how to organise society. These ideas do become relevant if they result in an action aimed at raising awareness, convincing others or even implementing these ideas. This work is interested in whether the creation of GNU/Linux and related FS/OSS can actually be perceived as a political action that is an expression of a particular set of political ideas of GNU/Linux programmers. An important point to make here is the question of awareness. An action can be politically significant without the actor actually deliberately pursuing a certain aim. So first it has to be discussed whether the actions of GNU/Linux-programmers are of political significance and if this is positively answered, it has yet to be analysed whether the programmers are aware of it and deliberately work for expressing a political idea. The latter is the main objective of this survey.

The following chapter aims at explaining why software is a political matter at all before setting out in detail why FS/OSS in particular is relevant for society. The main part will be a discussion of the ambivalent attitudes of developers towards a politicisation of FS/OSS and an attempt to explain the reasons for the contradictory findings of previous studies.

2.1 Software as Political Issue

I argue that software is of political relevance because of its crucial interface function in a world in which digital technology has such a prominent position and furthermore because its very nature embodies the challenges the political system faces from digital entities as a whole. In this context software is understood not only as digital data representing instructions that can be executed by a computer but also as standards for data formats and data exchange protocols. It is argued that software poses a number of challenges to established *general rules* by putting into question established economic, social and subsequently political assumptions.

To understand the crucial function of software one has to acknowledge the prime importance of information in contemporary society. The second half of the 20th century saw a global restructuring of capitalism with a general reorientation towards more flexible organisation, what Harvey (1989: 147) called *"flexible accumulation",* a new form of production characterised by the flexibility of *"labour processes, labour markets, products and patterns of consumption".* This reorientation saw the emergence of a truly global market, coupled with a shift in importance from manufacturing to services of first and foremost financial nature (see for example Strange, 1997: 90). This whole transition was both one reason for and a result of the development of a new bright star in the technology sky: Information and Communication Technologies (ICTs) (Schiller, 1999). Computers and digital communication networks, above all the Internet, were used to create a global market place, accessible for 24 hours, seven days a week and from almost everywhere in the world though not for everyone on the world. The most important commodity exchanged via the global communication networks is information. Information has become a prime value in modern capitalist societies as the finance sector unmistakeably illustrates. That leads Castells (2000: 17) to assert a new mode of development, *Informationalism*, which is characterised by the "*action of knowledge upon knowledge itself as the main source of productivity."*

This prime role of information greatly relies on the capabilities of modern ICTs. However, none of these electronic devices that permeate our life today would be

useful without a piece of software in it that controls its functions. Software constitutes the interface between us humans and the powerful machines that play such an important role in our life. As Zimmermann (2004: 12) argues:

> *"In this function software structures the economic and social options open to people in everyday working life."*

However, obviously the impact of software is ubiquitous and not limited to the working environment. The ways in which software is produced and distributed and what kind of usage of it is allowed, define to a large degree how a society is communicating and interacting. Let us take the Internet and the World Wide Web as an example (for a complete account of the invention of the WWW see Berners-Lee, 2000). Both were made up not of new hardware but of new ideas about how the existing hardware should communicate with each other. These ideas were fixed into certain rules, so-called standards, and software was written which made the hardware function according to these rules. First the researchers of ARPA and later the WWW-inventor Tim Berners-Lee made the deliberate decision to put all the standards they defined to the public domain as they wanted to ensure a general standard that everyone would be able to adopt and to use. If these researchers would have decided differently, the probability is high that a number of companies would have tried to develop their own rivalling standards as was the case throughout the history of computer networking till the widespread adoption of the Internet in the early 90s. So instead of the truly interconnected space the Internet constitutes today, the likely result would have been a fragmented network landscape. The story of the Internet and the WWW serves as an example of both how important a piece of software can be for society and of how influential the chosen form of distribution is. However, also the use software allows defines the options open to society as may be illustrated by software for Digital Rights Management (DRM), software that can restrict the user even from printing a copy of what he or she actually sees on the screen.

Yet the political significance of software arises not only out of its function but also out of the very nature of software itself as it poses a challenge to established assumptions of production and ownership. If we understand *"technology as an expression of human creativity"* (Youngs, 2001:382), then software is this human

creativity in its purest form. There is nothing material, nothing tangible to disguise the idea. It is for this reason that in software code there is embodied the whole problematic of our legal and political systems to deal with digital entities in general – be it music, books or any other kind of digitalised information. They all have in common that the only cost arises from their actual creation, that they are endlessly reproducible at marginal costs and that they travel over digital communication networks without respect to national borders. The result is that the traditional notions or property and copyright are attacked. In the case of the recording industry or software companies, whole business models are put into questions by the free distribution of digital data via "pirate" copies and peer-to-peer networks. For instance, in China more than 90% of all copies of Microsoft's Windows are not legally acquired by its users (Aldermann, 2002; Ghosh 2003). The implications are not only economic but directly political as the besieged are huge economic players which form a strong lobby demanding political reactions and extending existing copyright regulations to cyberspace.

The ongoing discussions about how to regulate the electronic sphere indicate that a policy for the digital realm including ways to deal with software and its social implications, has still to be found (see for example Doctorow, 2004). Originated in the programming sphere itself, FS/OSS offers a concept of how to produce, distribute and use software. It is of special political relevance because this concept is in discrepancy to established forms of production and ownership. As such it has been embraced by a number of actors with - as will be shown - diverging objectives. But as the analysis will outline, developers are contradictory in their acknowledgement of the political implications of FS/OSS and previous studies do not agree on whether developers' collaboration on FS/OSS goes beyond a pure technical activity as a means to promote and practise a certain social idea. I argue that the ideas and opinions of computer programmers are of great importance as they are the actual creators of software. How far reaching their impact can be can again be seen in the example of Tim Berners-Lee. But computer programmers and their opinions are not only of relevance because they are authors of software but also because they form a highly educated elite in an information centred society as they are able to cope with the demands of a world in which

information and its management is of prime importance. They can be expected to understand its structure and inner workings so their opinion can be considered to be an informed one and should have momentum.

2.2 The Political Significance of Free Software / Open Source

The FS/OSS development model has produced a number of highly successful projects (apart from GNU/Linux for example the Apache web server[iii] and the Mozilla web browser[iv]) and got, especially in recent times, increasing news coverage. The popularity of the FS/OSS model and its licenses is growing, having found already a variety of applications and imitations outside the production of software and beyond the technology sector. Examples for this are Wikipedia, a collaboratively produced encyclopaedia on the net, the Creative Commons licensing model[v] or MIT's OpenCourseWare[vi]. As different authors (Holtgrewe, 2004; Tuomi, 2002) have pointed out, the FS/OSS model represents a technological as well as a social innovation. On the technology side, the FS/OSS-programmers have not only created a number of software products that are seen by many as superior to proprietary alternatives (although there are also critical voices, see for example Levesque, 2004). They also managed to technically facilitate the large scale cooperation of people spread over the whole world. However, the social innovation is that these people actually work together to produce and give away a product for free that huge companies are used to charge for. They did not only introduce basic principles of scientific collaboration into software production, but also a new approach to ownership which is based on the premise that no one should own the source code and that is protected by suitable licenses which enable and encourage the distributed collaborative production of volunteers.

The ideas connected and hopes inscribed to the FS/OSS model depend very much on which of the both innovations is valued more. For economic actors the technical merits are most important and they are looking for a way to incorporate this development model. The social implications of a collaborative production of open code are therefore more an unintended consequence than a desired objective. A number of studies have been interested in the advantages of FS/OSS for innovation (Tuomi, 2002) and efficient software production, especially under the question of its viability for

companies (Bonaccorsi and Rossi, 2003; Wagner et al., 2003; Stewart and Gosain, 2003; Raymond, 1999). On the other hand, the FS/OSS model has gained a lot of proponents from alternative movements. For critics of capitalism, the technical innovation is only a proof that alternative methods of production and exchange are possible, fuelling expectations on the FS/OSS model as a counter concept to a capitalist society (Meretz, 2000). So it is no wonder that the collective action and the produced software has found friends among a lot of – mostly left-wing - political activists as Riemens (2003) highlights. Less radical proponents hope the model is leading to a more responsible and accountable form of scientific progress (Holtgrewe, 2004) as the development processes are transparent and open for participation of a broader public.

Both the economic approach to FS/OSS as well as the more social one have political implications on a number of different levels: By actually giving away highly valuable software for free the movement puts into question the foundation of a whole industry. In fact, it already changed part of the game with dedicated GNU/Linux-firms like RedHat and Suse as well as industry giants like IBM and Sun switching to GNU/Linux. While this is already of profound influence on the economy, the significance in economic as well as political terms comes especially out of the challenge this alternative system poses to established rules of intellectual property protection like copyrights and patents because the idea of free software is that no one actually owns the code. Coleman (forthcoming: 11) argues that FS/OSS is a *"cultural critique through contrast"* because it *de-familiarises* people with as given and effective accepted intellectual property laws by setting up a working alternative that is both highly visible and open to participation. As was explained, there are a lot of people who are most interested in the FS/OSS ideas of freedom, collaboration and sharing and who are especially attracted by its opposition to the giant transnational company Microsoft. They perceive it as part of a greater social concept and so new social movements like the globalization critics often go together with use and promotion of FS/OSS, adding further to its political impact. Take therefore the following quote from an activist of the alternative Internet news project Indymedia (Coleman and Hill, 2004):

> *"Earlier, someone said that [Indymedia] is a revolutionary project, and free software is a revolutionary tool for it."*

Another dimension to the political significance of FS/OSS is the option of technological empowerment that it offers developing countries (Reijswoud and Topi, 2003). As Gosh (2004) points out, both the FS/OSS software and the development model are highly attractive to countries that are struggling to build up their IT infrastructure. FS/OSS offers an ideal opportunity to obtain high value software at low costs and build an IT infrastructure while respecting intellectual property rights. Therefore GNU/Linux has been the operating system of choice in a number of developing countries (The Register, 25.09.2003) and special institutions were sat up in order to support FS/OSS there (see for example the *Free Software and Open Source Foundation for Africa,* FOSSFA[vii]). Additionally, FS/OSS is promoted by a number of development organisations. The proponents argue that it is not only the money that makes FS/OSS interesting for developing countries. The software also can prevent countries from being locked into a single vendor, especially one that is American. Other major reasons are the open standards and the high adaptability of FS/OSS, making it possible to produce a localized version of the software for markets that are not profitable for big software vendors (see for example The IndLinux Project[viii], an effort to create a GNU/Linux-distribution that supports a variety of Indian Languages). Furthermore it helps to develop the IT skills of local people and enables them to participate in global software development. Finally, an important factor in low-cost environments, GNU/Linux is said to perform significantly better on old (and therefore cheap) hardware than Windows.

2.3 FS/OSS-Developers as Political Actors

While having made the case for the political significance of FS/OSS as software in particular and as concept in general, it is the aim of this study to focus on the developers and to examine whether this political significance is acknowledged by them and whether it is a deliberate concern of the community or just an unintended side effect. The following section will outline two extreme contrasting positions in the community and subsequently discuss the findings of previous studies in order to make out a general tendency in favour or disfavour of FS/OSS as an idea and practise with political relevance.

2.3.1 Free Software Foundation versus Open Source Initiative

Concerning a political approach to FS/OSS, the Free Software Foundation (FSF) and the Open Source Initiative[ix] (OSI) represent two contrasting poles in the community. At the political end of the opinion spectrum is the Free Software Foundation with its GNU project, founded by Stallman. While the people participating in it are often highly skilled and value good code, Free Software was and still is about much more than creating software. As Stallman (2001: 167) puts it:

> *"[...] speaking for the free software movement, I talk about issues of ethics, and what kind of a society we want to live in, what makes for a good society, as well as practical, material benefits. They're both important. That's the free software movement."*

The FSF states that no one should actually own a program. While they see economic disadvantages in the use of proprietary software, the main argument against proprietary software is a social one. The evil of copyright would be that it destroys people's solidarity, the ability 'to help your neighbour', by prohibiting modifying and copying programs. Stallman (Moody, 2001: 28): *"I consider that immoral."*

In fact, the people of the free software movement extend the idea of sharing software beyond the pure technical issue to a general question about the basic values of our society. One can clearly identify political claims in the philosophy of the FSF, claims that are very much based on the libertarian ideas Stallman identifies himself with and that are informed by the hacker ethic, as is illustrated by the emphasis that is placed on the *freedom* of the user by preserving the users' rights and not restricting him. The GNU software therefore is one means of transporting a political idea. Again Stallman (2002a: 22):

> *"Even if GNU had no technical advantage [...], it would have a social advantage, allowing users to cooperate, and an ethical advantage, respecting the user's freedom."*

But what kind of political idea is the FSF promoting? While Stallman was criticising big corporations that relentlessly pursued their profit to the disadvantage of the people, his libertarian belief is not anti-corporate or anti-capitalistic (even if his critics often connect him to communism). What Stallman and the Free Software movement are not denying is the fact that a programmer should be paid, but not for the

software itself but for the work he or she is putting in it. It is therefore totally acceptable to charge money for free software as the FSF in fact does. What is more, the freedom of free software also leaves total freedom to use it for any commercial purpose, including for example military ones as has already happened (Slashdot.org, 01.03.2003). So in this sense the ideas of the FSF might not have that much in common with for example the ones of globalization critics that promote and use FS/OSS.

While it was shown, that the proponents of free software have a political idea, this political engagement is by no means shared or even welcomed by the whole community. The rejection of a politicization of Free Software went so far, that in 1998 the Open Source Initiative was founded in what was essentially an attempt to establish an apolitical alternative to the Free Software movement. Co-founder Eric S. Raymond (1998b: 212) sums up the main reasons:

> *"It seemed clear to us in retrospect that the term "free software" had done our movement tremendous damage over the years. Part of this stemmed from the well-known "free-speech/free-beer" ambiguity. Most of it came from something worse – the strong association of the term "free software" with hostility to intellectual property rights, communism, and other ideas hardly likely to endear themselves to an MIS manager."*

The Open Source Initiative was eminently a try to make the movement more attractive to business by putting emphasis on the openness and common production of the source code and by disengaging with many of the ideas that went further than that. The OSI is less interested in the social implications of FS/OSS but sees economic disadvantages in the use of proprietary software as it prevents the software from being used and developed in the best and most effective way. The OSI is promoting the bazaar-style development as it was used for the Linux-kernel while not making too many claims about the kind of software that should be the result. Therefore the FSF claims (Stallman, 2002b: 55):

> *"Open source is a development methodology; free software is a social movement."*

An apolitical approach to FS/OSS is also purported by no one less than the major publicly prominent figure of the movement, Linux initiator Linus Torvalds:

> *"I can't totally avoid all political issues, but I try my best to minimize them. When I do make a statement, I try to be fairly neutral. Again, that comes from me caring a lot more about the technology than about the politics, and that usually means that my opinions are colored mostly by what I think is the right thing to do technically rather than for some nebulous good." (Diamond, 2003)*

2.3.2 The Political Developer: Contrasting Findings

So with the FSF and the OSI at the two opposing ends of the political spectrum, which approach favours the majority of the developers? There are many authors that argue that the foundation of the OSI is just one sign of a deeply in the community embedded concern about a politicisation of the work of the developers. In fact, there are several authors who assert that the majority of the developers is simply centred around the technology they are producing rather than its social benefit. Hannemyr (1999) argues that

> *"[...] the hacker's fascination with technology is not because they believe that technology will bring about great and revolutionary changes (or make any societal difference whatsoever). Hackers love technology for its own sake."*

Coleman (forthcoming: 7) argues similarly that the main motivation of FS/OSS programmers is to protect software and the technical process of open development because it ensures a better product as well as the ability to hack. Yet she adds that FS/OSS developers in fact see that free software is beneficial for society but they deny any political motivation. Coleman sees several reasons for the insistence on political neutrality: First, the whole process is seen as purely technological and the openness simply ensures better software while political ideas could prevent the free circulation of thought. Second, to promote a particular political view would mean to diminish the freedom of people with other opinions who want to use the software. Third, it is feared that a politicisation would prevent others from participating in the movement. Finally, politics is considered as uncool and ineffective. Coleman believes that it is this insistence on political neutrality that allows the concepts its great impact. The neutrality

of FS/OSS would have enabled its wide application across a variety of diverse backgrounds – from counter-cultural groups like Indymedia to world-class capitalist business such as IBM. Also Tuomi (2002: 214) asserts the community an advantage by focusing on the technical product instead of its social implications:

> *"The culture of hacking is probably the most perfect and frictionless implementation of modernity, and therefore it also produces technological products effectively. There are no deep internal conflicts that would compromise its competence-based efficiency. As long as it builds itself around those technological artefacts that it produces, it is able to avoid many of those conflicts that make similar efficiency difficult in broader social contexts."*

Tuomi's argument is illustrated by another statement of Torvalds:

> *"I think of myself as an engineer, not as a visionary or "big thinker". I don't have any lofty goals. I just want to have fun making the best damn operating system I can."* (Diamond, 2003)

Torvalds also puts the focus on the fact that the participation of high-skilled computer programmers to take part in creating a product that for most of them offers no direct financial reward, does not have to be motivated by *"lofty goals"*. FS/OSS offers a lot of personal benefits for the developers, first and foremost to do something they enjoy: programming. A survey of Lakhani et al. (2002) with about 700 selected participants observed that intellectual stimulation or enjoyment is the main motivator of programming. A number of other authors also emphasise that contribution to FS/OSS has by no means to be altruistic. In fact, Lancashire (2001) is convinced that the application of economic theory to FS/OSS collaboration explains better why people contribute to FS/OSS than hacker ethic or post-scarcity gift economies. He argues that for most people FS/OSS involvement is a means to improve their skills in order to get a (better) job. A survey by Robles et al. (2001) supports this assumption as about 70% of the 5,500 participants say they have already profited professionally from their engagement with FS/OSS or they expect to do so. Another study by Ghosh (1998) explores the same topic. He also argues that FS/OSS should not be understood as a gift economy. Instead the contributors know that they will get something back for their efforts, for example in the form of other free software they can use or the help of other people in developing software. Ghosh (1998) believes that FS/OSS resembles *"an*

implicit barter economy with asymmetric transactions". While the contributor may not get an immediate reward, the decision to offer one's own work is an absolute rational one as in fact there will be a service in return that, as Ghosh (1998) states, is in sum even greater than the contribution.

The following graphic (Ghosh, 2004) gives a summary of a survey with 2,800 participants by Ghosh et al. (2002) and its findings concerning the diverse reasons that motivate developers to contribute to FS/OSS:

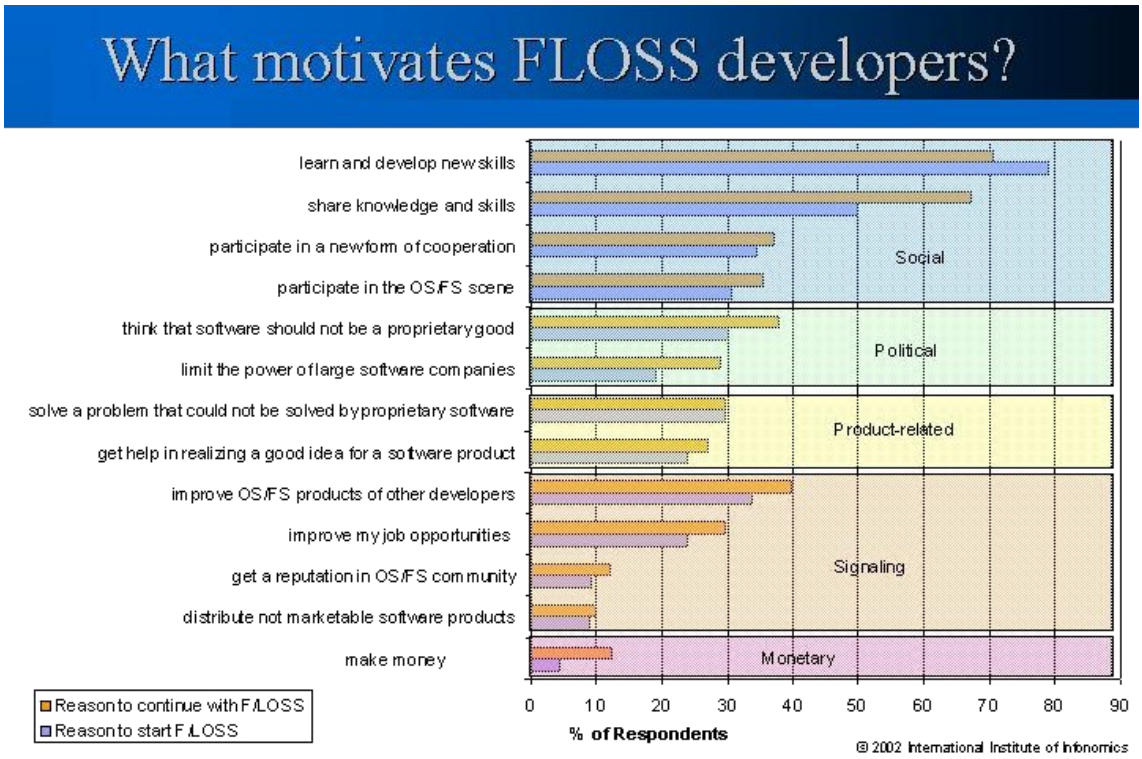


Illustration 1: Motivation of FS/OSS developers. Findings of FLOSS.

The overview illustrates that despite the often non-paid voluntary nature of most developers engagement with FS/OSS, programmers can gain a lot from FS/OSS-development. Apart from learning and developing new skills, programmers can benefit of FS/OSS participation by getting help for realizing a software product, solving a problem that could not be solved by proprietary software or improving their job opportunities. On the other hand, that hackers participate in FS/OSS out of personal reasons can hardly be surprising and is by no means immoral. The question is simply whether the reasons for participation stop at the personal benefit or if there is something more beyond.

Indeed, while this overview confirms that programmers participate in FS/OSS projects for their personal benefit, it also shows that another big share is held by motivations which cannot (or not completely) be explained by the pursuit of a personal benefit. About 50% (Ghosh et al.) of the programmers are motivated to join the community because they want to share their knowledge and skills and that motivation gets even stronger once the people participate. Other reasons, shared by about one third of the developers, are the wish to participate in the FS/OSS scene and to take part in a new form of cooperation. According to Ghosh et al. (2002), slightly less people are part of the scene because they want to limit the power of big software companies. The survey by Lakhani et al. (2002) further reveals that about 30% write FS/OSS because they feel an obligation from their use of FS/OSS. So obviously *"selfish"* motivations are not the whole picture. Furthermore, even hardcore "no-politics" evangelists like OSI co-founder Eric S. Raymond (1998a: 29) admit that the community as a whole has been increasingly active to lobby the political arena due to governmental proposals of tougher communications or software regulations. In response to this, starting with Salin (1991), there is a growing understanding of source code as speech (Coleman, forthcoming: 2), hence source code should by law be secured under the constitutional guaranteed freedom of expression, an argumentation that for example the FSF is promoting. The claims made by the community have already led some authors to attribute FS/OSS the status of a social movement. In his analysis, Zimmermann (2004) asserts FS/OSS developers a certain degree of organization, a collective identity (around hacking values) and actions that are aimed at a profound social change in society. Another finding of Coleman and Hill (forthcoming) is, that while most developers do not have political or ethical reasons to join the community, through participation and discussion among their fellow hackers they come to value the principle of free software as well as its validity for other realms of life:

> *"The technical success of free software projects and the personal gains from participation jointly reinforce the hacker belief in openness and information sharing with relevance and applicability in other domains of social and political life."* (Coleman and Hill, p. 280)

2.4 The Problems of Determining FS/OSS Politicisation

I would argue that the difficulties of previous studies to deliver a coherent picture of the politicisation of FS/OSS developers mainly stem from three different sources that every new study has to acknowledge in order to produce a valid account of the politicisation of FS/OSS.

First, there is the problem of the diversity of motivations: As the motivational studies have shown, there are various reasons for FS/OSS developers to contribute to FS/OSS which need not necessarily be altruistic. However, the important question is not whether a programmer can benefit from FS/OSS but whether this is the sole motivation. Furthermore, the diversity of motivations is increased by the very openness of the community as it allows for a variety of ideologies that FS/OSS can be used for. If one considers that entities as different as Indymedia (that uses and develops its own free software content management system, called Mir[x]) and IBM use and develop FS/OSS solutions, it is clear that these do not operate on the same ideological ground. I am critical of the distinction between apolitical hackers and political users as Coleman argues. Similarly Riemens (2003) urges for a distinction between hackers and activists: While there would exist common concerns, for example the freedom and independence of the Internet, hackers that produce the software would generally not subscribe to the (mostly left-targeted) greater political aims of the activists that only use the software. I would argue that for FS/OSS this distinction is difficult to make as the openness of FS/OSS blurs the very distinction between producer and user by empowering software users to take part in the development process. So the political *users* of FS/OSS can become part of the community of *developers* with all their ideas about society and the role FS/OSS should play in it. Indymedia was already named as such an example, another one would be the Association for Progressive Communications[xi] (APC). Both are not only using FS/OSS with a clearly political motivation but also producing their own FS/OSS. So every study of the motivations and opinions of FS/OSS developers has to be aware that there can be no single, “right” opinion. To acknowledge this diversity means not to conclude from the findings of a certain group of developers to the whole

community: that is valid for the political motivations of the FSF as well as for the "just an engineer" - vision of a Linus Torvalds. The existence of these extremes clearly demonstrates that FS/OSS cannot solely be either the one or the other.

Second, another difficulty in getting to grips with the political assessment of FS/OSS by the community itself is that the developers often might not be self-aware of the political ideas that are the base for their statements. Coleman (forthcoming: 4) asserts that the developers' emphasis on freedom is not only the result of the practical needs of programming:

> *"While technical or economic rationalities are often the native explanation for FOSS, a taken for granted form of cultural liberalism and the pragmatics of programming mutually inform and reinforce the hacker aesthetic distaste for politics."*

So it can indeed be said that the FS/OSS concept is based on certain political values and assumptions, namely liberal ones, and therefore it also transports these values, although many developers might simply be not aware of this. Weber (2004: 5) for example argues that both movements, Free Software and Open Source, are politically informed by a libertarian ideology although they interpret it differently. The apolitical self-image of the OSI can be ascribed to the fact that, as van Dijk (2000: 45) notes, the libertarian ideology often sees itself as apolitical as it champions the freedom of the individual and the retreat of the state to the role of a night-watchman. However, it is clearly a political ideology and as such of relevance for the actions of the developers. So what can be concluded is that it is indeed possible to see an ideological underpinning of FS/OSS independent of whether its proponents itself actually acknowledge this. Therefore there are limits to the conclusions that can be drawn from the self-assessment of developers.

Finally, when trying to assess the role politics plays for FS/OSS developers, it is important to acknowledge that the digital sphere in which programming takes place, is different from the traditional public sphere and that subsequently the definitions we used to have for political participation might not be suitable to register the actions of computer programmers in terms of their political/social objective. I would argue that the

developers already express a particular social idea by participating in a global network of voluntarily collaborating people to produce a technical product that is given away for free. As Everard (1999: 160) puts it:

> *"The Internet is a cultural artefact. As such it encodes within its structures, its technology and the language surrounding it, a world-view or philosophical outlook. New habits of thought emerge from the way the Internet structures information."*

That the culture of hackers exists in the virtual worlds of electronic communication networks does not mean that its impact would not be real. The whole identity of these people who actually *"live"* in that *cultural artefact* is considerably shaped by their daily practice and experience online. However, there does not (yet) exist a coherent ideological framework that would sum up the values and beliefs of most FS/OSS programmers. Apart from it being a relatively recent phenomenon (at least in this degree of participation), one of the reasons for this is what Stewart and Gosain (2003) call a hackers' belief in practical work being more beneficial than theoretical discussion. But that does not necessarily have to be a weakness. While the programmers' ideas do not come out of a thoroughly thought-through theoretical approach to society, they come out of their very practical experiences around the technology they use and create. They champion certain ways of life because they have experienced them to be useful and beneficial in their computer-centred world. To sense how they promote them, we might not get too far if we simply look for signs of traditional forms of political participation such as party identification, protest action or public manifestations. In this sense the computer programmers might be simply apolitical, though I would argue that they are politically active. A first clue can be the hacker ethic, that also states: "*Computers can change your life for the better"* (Levy, 1994: 45). The first hackers gave their successor on the way a firm belief in modern technologies as a force for good. There was and still is the belief that these technologies can be used to overcome social problems. So hackers' aims for society do not directly translate into political action in a form we would expect. Rather they are mediated by technology as they are pursued mostly in a technological realm (nowadays predominantly the Internet) and promote certain technologies as a means to achieve a certain social behaviour.

Chapter 3: Methodology

3.1 Research Questions

The main objective of this work was to find out about the political motivations of FS/OSS developers, in particular developers of GNU/Linux. Despite the number of studies that are concerned with the subject and that were outlined above, there was yet no study that focused exclusively on political explanations for developers' motivation. Therefore this research project was an attempt to close this gap. However, as the problems of the other studies have illustrated, that is no minor task and so due to the rather limited scope of this project, this study can just be a start. While it was discussed that the concept of FS/OSS is indeed of political significance, the developers' perceptions were contrasting. So the main questions to be answered were as follows:

1. How do developers assess FS/OSS in terms of political relevance, especially in relation to the contrasting views of FSF and OSI?
2. Is the work of FS/OSS developers partly politically motivated?

In response to the outlined problems of previous studies due to the diversity of opinions in the community and the new forms of political participation, further questions were:

3. How does political participation offline relate to political participation online and the other way around?
4. What distinguishes developers with a high political motivation from ones with low or no political motivation?
5. What are the political views of developers?

3.2 Research Method

To achieve the data necessary to answer the research questions I decided to carry out a survey of GNU/Linux developers. Therefore I sat up an electronic self-completion questionnaire that was put online and to which developers were invited via announcement on relevant mailing lists or message boards. (Please refer to the Appendix for a listing of the questionnaire and the invitation letter). In addition I created

several computer scripts that collected the information from the questionnaire form and stored it in a text file that later could be easily imported into the statistical program SPSS to analyse the data.

3.2.1 Instrument

The use of a survey design for this research immediately brings up the issue of the limits of self-awareness that was discussed earlier as one of the problems of previous studies. That I decided for it none the less was due to the fact that the developers with their personal opinions and actions stand in the centre of the research questions. To ask them directly seemed to be the most straightforward way of acquiring the necessary information. What is more, this kind of data did not yet exist. Another main advantage of the survey design was the possibility to acquire more data from a greater audience than for example a participant observation or interviews would have been able to generate. A great number of participants was considered necessary due to the outlined diversity of motivations in the community. To cope with the problem of inaccurate self assessment, the researcher must not totally rely on the expressiveness of the answers and try to access issues not (only) by straightforward asking but by a broader inquiry into the greater framework of opinions and actions.

The application of a self-completion questionnaire online offered convenience for both the participants and the researcher. For the participants there was no need to synchronize with the researcher in time and space as the questionnaire was online 24 hours a day and accessible from every place in the world that is connected to the Internet. Furthermore, as the questionnaire consisted mostly of closed questions its completion required less effort from the participants. For me as researcher there were a number of benefits: First, there was no need to personally engage with the participants and after setting up the survey it could run without interference from the researcher. The closed questions of the questionnaire provided me with standardised and therefore comparable data. Nevertheless, after consultation with the supervisor some open questions were built in as well as a general comments field at the end of the questionnaire to give participants opportunity to add their own views apart from the standardised questions. That also provided the researcher with more possibly valuable

material that otherwise would have been lost. Finally, a major advantage of a computer-based questionnaire is that the generated data is already in digital form and therefore easy to process for a statistic program. A whole work step – the coding of information – is omitted.

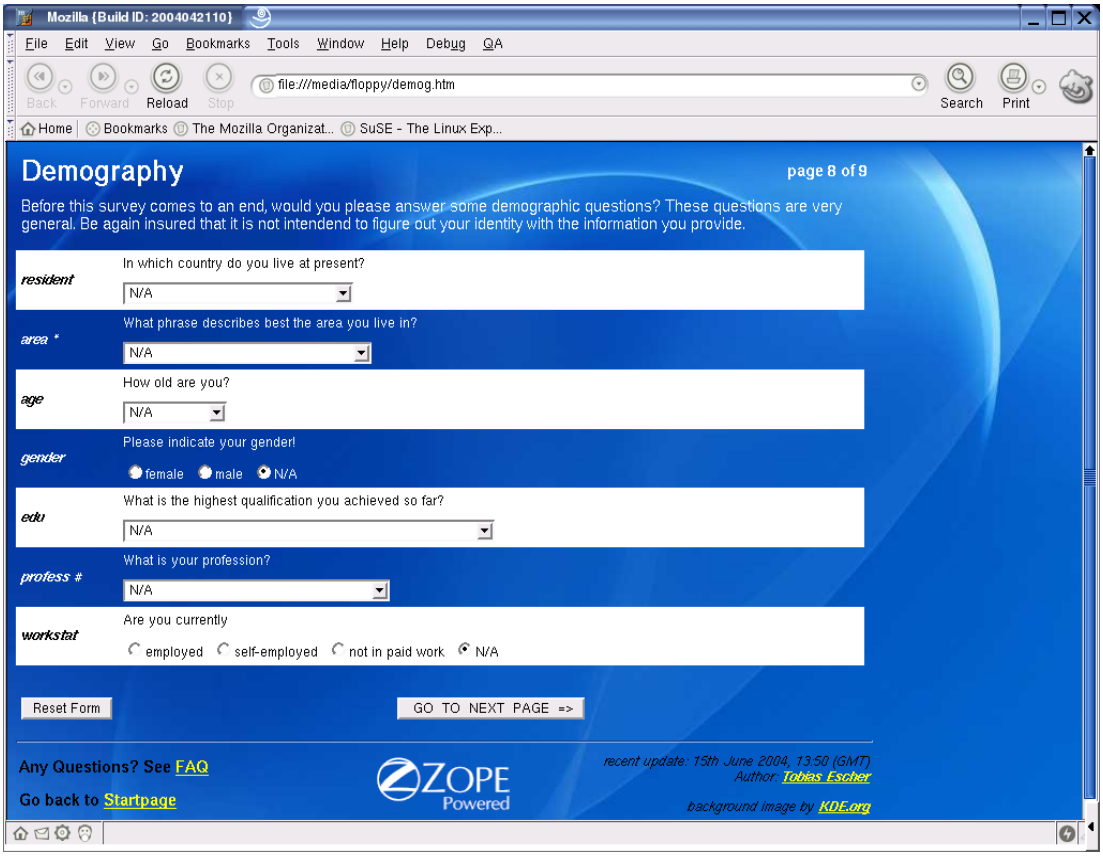


Illustration 2: Screen Shot Questionnaire

However, the main argument to choose the Internet as the medium for carrying out a survey on GNU/Linux developers was to adapt to the developers' usual environment and to get access to the community in order to reach as many respondents as possible. This was ascribed special importance because, as Bryman (2001: 131) points out, self-completion questionnaires often have problems to find enough respondents as they lack the opportunity to motivate people into actually taking part in the survey. It was hoped to create a kind of insider status for the researcher that could encourage participation by using free software to run the questionnaire and the incorporation of a special page answering possible questions, a feature well known in the programming scene (Frequently Asked Questions, FAQ). This was further pursued by putting the generated data to the free use of others (respondent's agreement

provided), in other words to apply some of the principles of FS/OSS to this survey. Apart from that participants were ensured anonymity and it was made transparent which data is collected. Finally, the questionnaire on the Internet did not request the developers to engage in any unusual activity (like filling out a paper questionnaire) and therefore hopefully added further to the reliability of the data.

3.2.2 Questionnaire Content

The questionnaire as it is presented here and was finally deployed for the survey is the result of several refining processes due to discussions with the supervisor and points raised by pre-testers familiar with the topic. A major improvement of the questionnaire during that process was the inclusion of several open questions instead of a completely standardised questionnaire.

The main interest of the survey was the developers' assessment of the political relevance of FS/OSS and inasmuch they are politically motivated to contribute to FS/OSS. It was tried to achieve this information in several ways: On the one hand, developers were plainly asked how important the political goals of FS/OSS are for them, if there are any at all, and how much of their motivation to contribute to FS/OSS is political. One the other hand, the respondents were provided with a catalogue of statements that expressed certain objectives for participation in FS/OSS – some self-related, some aimed at the community, some concerned with the special requirements of low/medium income countries – and which the developers should rate according to the importance for their personal motivation. The idea was that rather apolitical developers might be motivated more by self-interest than community goals. Furthermore, developers were asked for their assessment of Free Software in comparison to Open Source Software and to state which movement they feel they belong to. As was shown, both movements have different views on a politicisation of the community and therefore a choice of one of them would also state a certain attitude of the developer. Another aim was to investigate the relationship of online and offline political participation and therefore participants should state their interest in politics in general and how often they access political news. Most importantly, questions were asked about the participation in any form of political protest in the last half year, separately for offline and online action.

While equipped with these questions the survey should have been able to provide some useful results, I was also interested in the nature of the political motivations, as a political motivation alone says nothing about what kind of politics the developer favours. However, this objective proved difficult as political opinions are a private matter and asking for them could prevent possible respondents from taking part in the survey as my supervisor pointed out. So finally I decided to stick to technology related statements and questions from which I hoped to get some insight into the programmers' opinions, for example how they consider the hierarchies in FS/OSS projects or who should be responsible for Internet governance. Of special concern in this complex was whether the values of the hacker ethic are still important for the respondents. Finally, to be able to recognise in what ways developers with different opinions differ from each other as well as to gather some background information about the developers, also the usual demographic questions were asked (education, profession, etc.) as well as some FS/OSS specific ones that could give an insight into respondents involvement in FS/OSS (number of projects, paid for development of FS/OSS, etc.).

3.2.3 Sample

3.2.3.1 Sampling

This survey was interested in people who are taking part in the development of GNU/Linux. As the GNU/Linux operating system nowadays consists of so many different FS/OSS programs, it was difficult to distinguish between FS/OSS developers who should be part of the sample and ones who should not. Yet, as the focus was on people developing not only for a single program but with the whole operating system in mind, I contacted developers that were involved with the production of particular distributions of GNU/Linux.

Rather than contacting developers directly I decided to approach them via the project's mailing lists or message boards because these could be accessed more easily and provided a good means to reach a broad audience without setting individual programmers under pressure. To respect the groups environment and not to annoy possible respondents, mails were mostly sent to the administrator of the project with the plea to forward it to the mailing list if it was considered appropriate. That also helped to

increase the responsiveness of the developers. Another advantage of this approach that became clear during the research is that they are open for other interested readers who might provide useful comments on the research even when not actually participating as developers. To ensure that only invited people could take part, the survey was password protected with each project obtaining a different login. That allowed to measure response rates and also to later filter the data according to project if necessary.

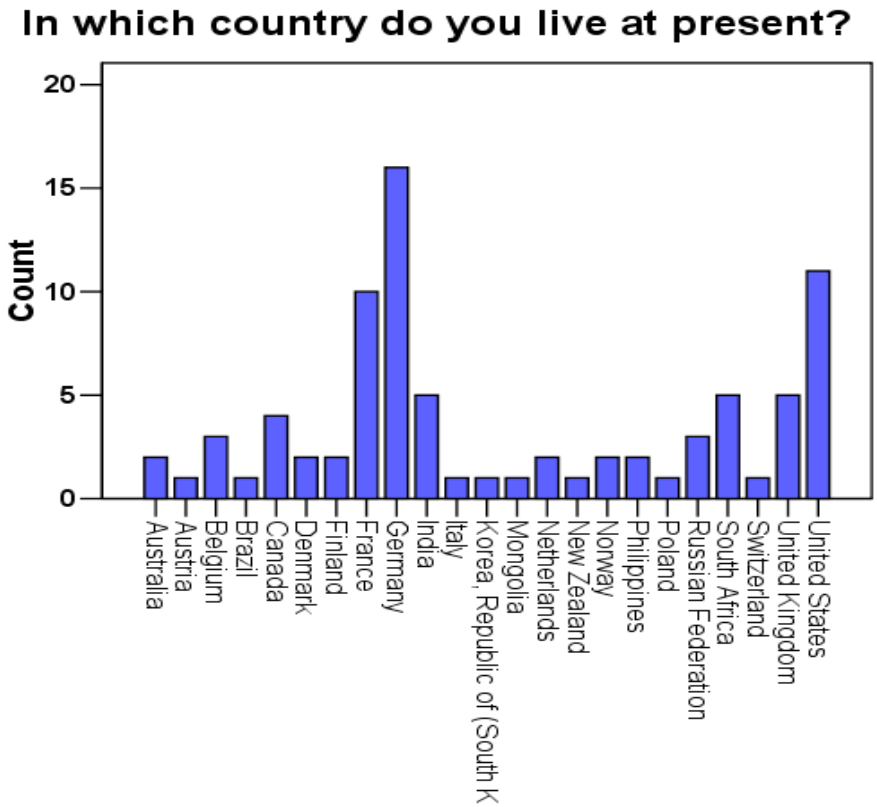
In order to achieve a broad overview about the different motivations of the community, the selection of the sample had to be ascribed great importance. Generally, a sample of completely randomly selected respondents would be desirable to allow to infer from the sample to the overall population. That seemed impossible to realise as neither the overall population could be clearly identified nor a process implemented to randomly select participants. To make matters worse, a self completion questionnaire always relies on the cooperation of the targeted person so in any case a self selection process takes place that might lead to the sample just representing a certain subgroup of developers such as the ones who are politically motivated and want to express that. To tackle this problem the announcement explicitly said that also developers are invited who considered themselves as not politically motivated. Apart from this, as has already been discussed, poorer countries have special incentives for the use of GNU/Linux, and to get the views of developers from these countries I deliberately tried to contact GNU/Linux distribution projects that were initiated there. To add further to the diversity of the sample I also tried to include not only projects of volunteers but also developers from commercial GNU/Linux distributions.

Finally data was collected during the time of 15th to 24th June. During this time 85 developers from 10 out of 23 contacted projects responded. The overall response rate was even higher with more than 200 participants by the beginning of August. However, the majority of these responses came too late to be included into this research as an invitation was forwarded belatedly by an administrator. The following section will give an overview about the main characteristics of the sample. Please note that values in brackets found in the captions indicate the number of respondents that answered the question. Two asterisks indicate that a value is significant at a 0.99 significance level,

one asterisk signals a 0.95 significance level (always 2 tailed). Correlation always means Pearson correlation coefficient and mean differences were tested with a t-test for independent samples.

3.2.3.2 Sample Demography

The data analysed in this work covers the answers from 85 participants from 23 different countries. The highest number of respondents came from Germany (16), followed by the USA (11) and France (10). Fortunately, the questionnaire reached a lot of programmers from low/middle income countries and so the fourth place is occupied by South Africa and India with five developers each.

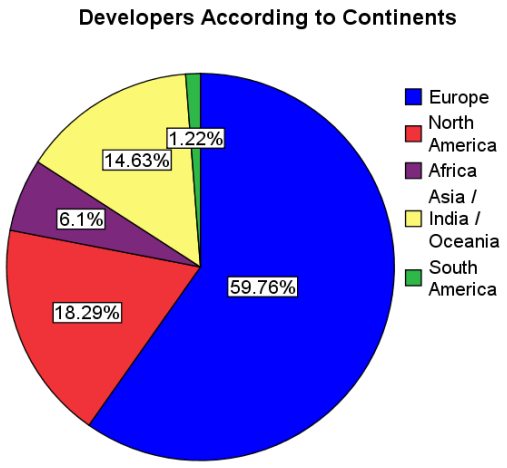


Graph 1: Countries of Residence. (82)

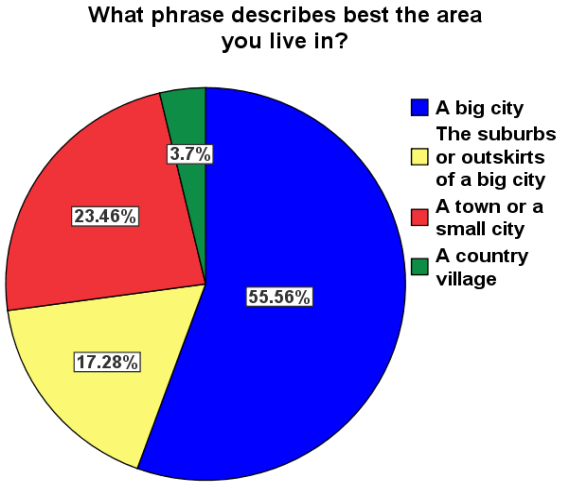
The highest number of developers in the sample came from Europe (60%), followed by North America (18%) and countries in Asia, India or Oceania (15%). South America (1%) is only poorly represented with one programmer from Brazil. Overall, the majority of respondents (78%) came from high-income countries (HICs) while programmers from low/middle income countries (LMICs) accounted for 22% of the sample[1]. Compared to the survey of Ghosh et al (2002), in which roughly about 7%

1 The classification was made according to the Human Development Report 2003 of the United Nations

came out of LMICs, the aim of the survey to include more developers from these states was achieved. Almost all developers (96%) live in urban settings, in big cities or their outskirts (69%), while only about 4% live in a country village. According to the United Nations World Urbanization Prospects (2004: 35), 74.5% of the population of developed regions live in urban areas while that is only the case for 42.1% in less developed regions. Either way, the urbanization rate of the sample is much higher, giving testimony to the digital divide not only taking place between countries but also in states between urban and rural areas.



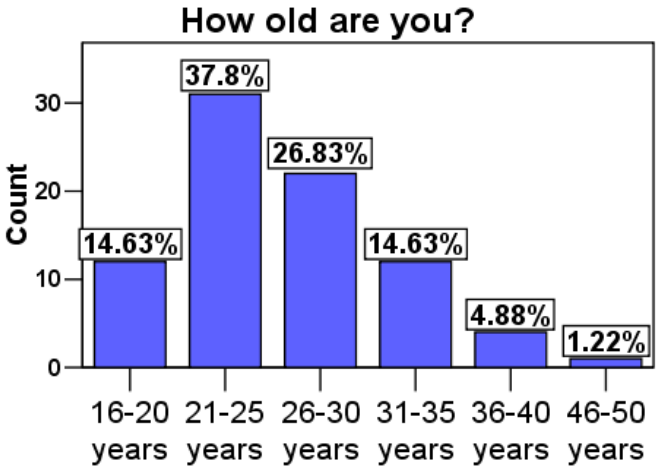
Graph 2: Continents. (82)



Graph 3: Area. (81)

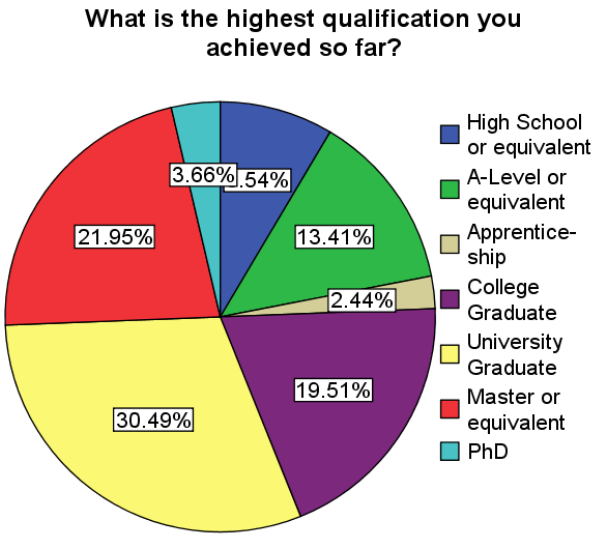
As could be expected after the findings of previous studies, almost all participants were male with only two female respondents. The age structure corresponds with the findings of the mentioned studies with the majority of developers (79%) aged between 16 and 30.

Development Programme (UNDP). Low income countries (Gross National Income per capita less than $745 in 2001) are India and Mongolia. Middle income countries (GNI/capita in 2001: $746-$9,205) are Brazil, Philippines, Poland, Russian Federation and South Africa. The report is available online at: http://hdr.undp.org/reports/global/2003/ [07.07.2004]



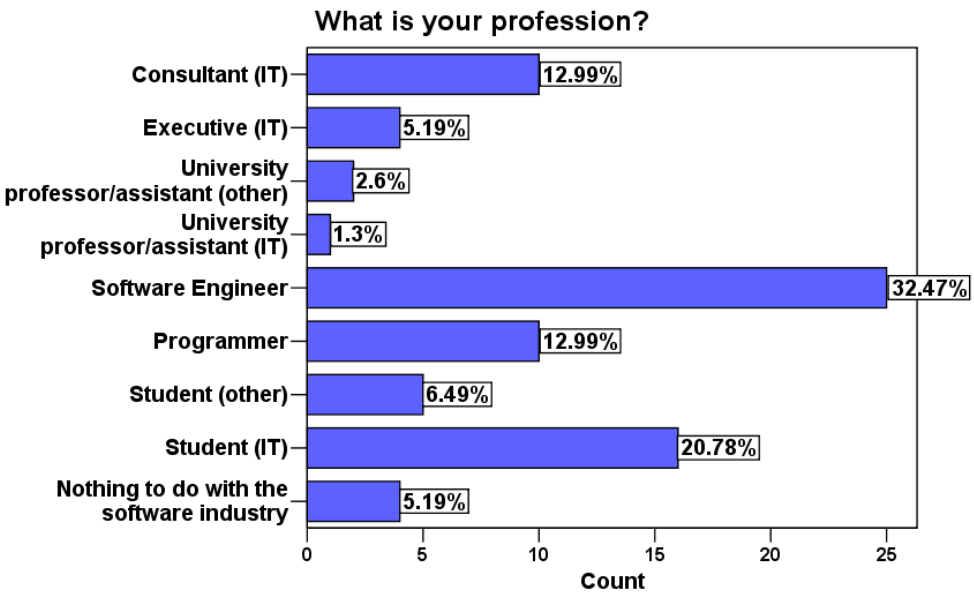
Graph 4: Age. (82)

Also the findings of previous studies concerning the high level of education are confirmed in this survey. University degrees (University Graduate, Master, PhD) made up for 56% of the sample.



Graph 5: Education. (82)

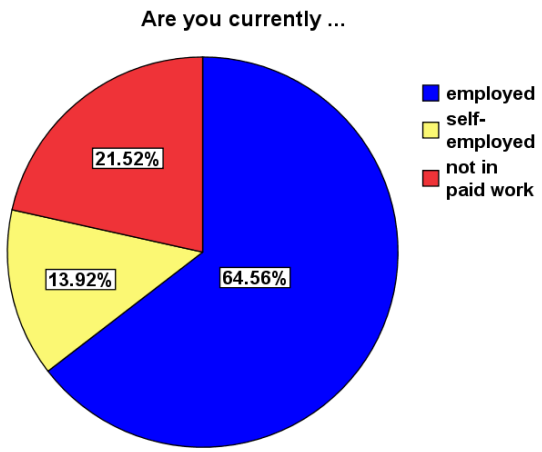
The majority of the respondents (86%) is involved in information technologies either by work or by studies. The biggest shares are held by software engineers (32.5%) and IT-students (21%). Only 5% earn their money outside of the software industry. Despite the openness of the community, participation still requires skills that are obviously not easily acquired by people without an IT background.



Graph 6: Profession. (77)

A view on the current work status shows that over three quarter of the participants are in paid work with the majority of this share being employed (65%). The comparatively big percentage of respondents that are not in paid work (21.5%) can probably be explained by the high number of students and young people in the sample.

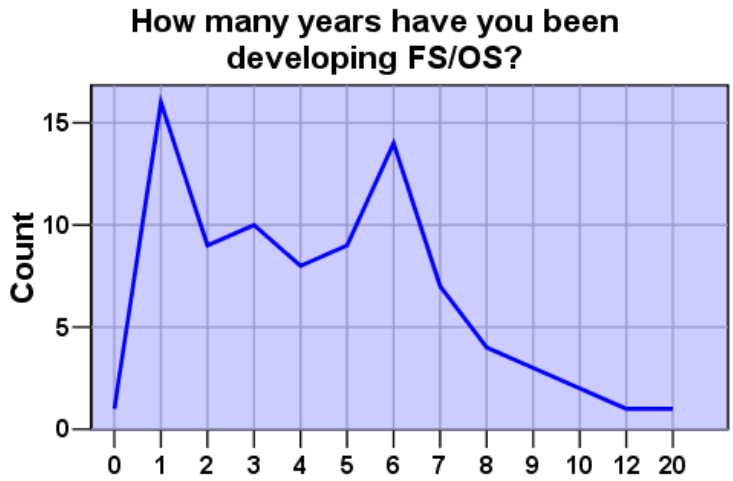
An overall comparison of the demographic findings with the ones of previous studies shows that despite the small sample size the major characteristics of the population of FS/OSS developers could be affirmed. The main difference to previous findings is the increased number of developers from low/middle income countries that was a deliberate feature of this study. Apart from that the sample seems to be rather representative for the overall population of FS/OSS developers.



Graph 7: Work Status. (79)

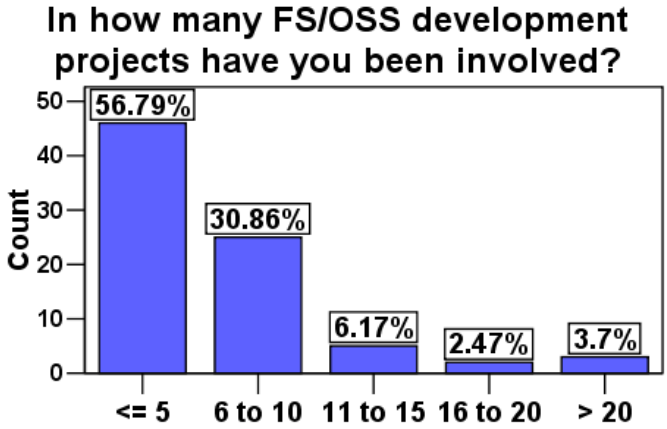
3.2.3.3 Sample FS/OSS Involvement

The majority of the respondents (77.7%) has been developing FS/OSS for about one to six years. The average involvement in FS/OSS is 4.5 years. Only few people have been involved for eight years or longer. That might be explained on the one hand by the fact that the public involvement starting with the Open Source movement is still quite recent and on the other hand that FS/OSS, as the age structure shows, is a predominantly youth phenomenon.



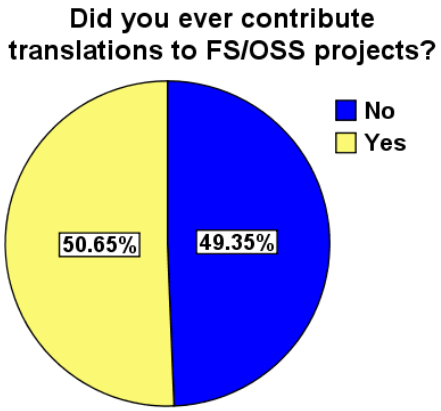
Graph 8: Years of FS/OSS Development. (85)

More than 55% of the participants have been involved in up to five projects so far and the majority of 88% has not been contributing to more than 10 projects. The ones who have been part of the movement for longer are also more likely to be involved in more projects (correlation: 0.445**).

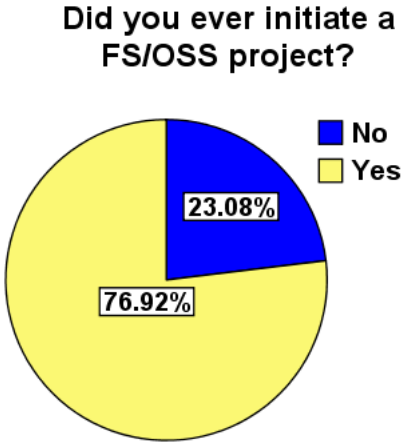


Graph 9: Development Projects. (81)

While contributing translations to FS/OSS projects does not seem to be too popular, the big majority of the developers have themselves already initiated a FS/OSS project. So many respondents of this sample are not solely contributors to GNU/Linux-distributions but have already been active maintainers of their own FS/OSS project.

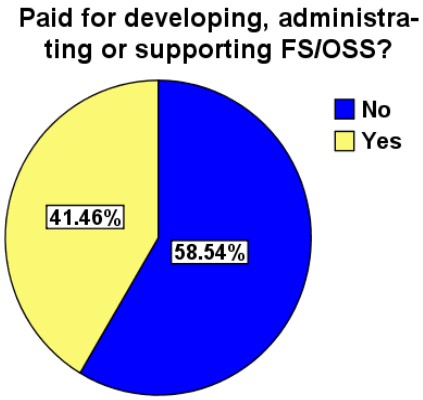


Graph 11: Contribution of Translations. (77)



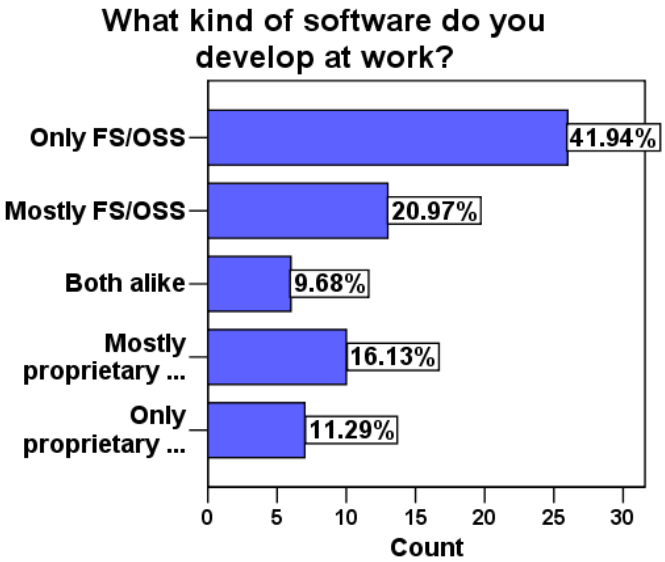
Graph 10: Project Initiative. (78)

About 40% of the questioned developers earn money from FS/OSS, either by developing, supporting or administrating FS/OSS. Previous studies do not agree on the share of paid developers on the whole community. The figures vary from 20% to 30% (Lakhani et al., 2002; Robles et al., 2001) for programmers that are paid only for developing FS/OSS, to around 50% if income from administration or support of FS/OSS is included (Ghosh et al., 2002). However, the latter figure also included non-monetary rewards such as job promotion, hence the real figure could be lower. Therefore the observation of this survey might not be too far from the true share of paid developers, although it has to be remembered how the sample was achieved.



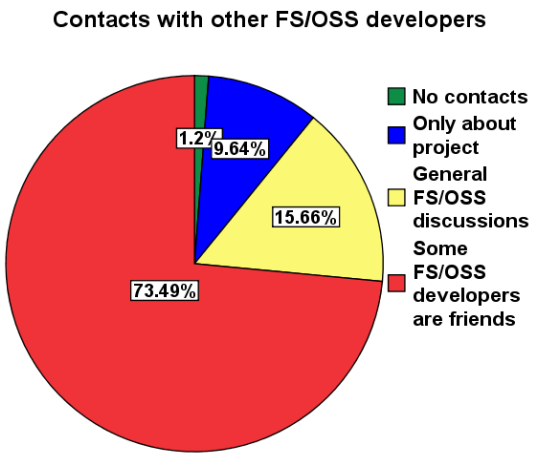
Graph 12: Paid for FS/OSS. (82)

However, 40% of the ones who stated not to earn money by FS/OSS are developing at least some FS/OSS software at work. While five of these twenty-one people are students, the remaining sixteen obviously develop FS/OSS at work although it is not part of their job. Ghosh et al. (2002) already showed that it is not uncommon for FS/OSS programmers to code FS/OSS during their working time, often without the knowledge of the employer. Although the question was designed to aim only at programmers who develop FS/OSS as part of their job, it seems it was not formulated understandably enough because I do not have another explanation for this result.



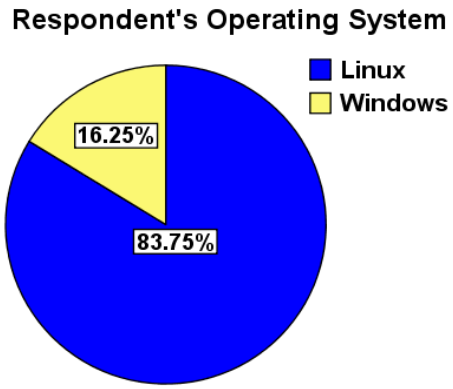
Graph 13: Development at Work. (62)

The findings about the relationship the respondents have with other developers show that one can speak really of a community as more than 70% of the respondents have friendly relationships with other FS/OSS developers.



Graph 14: FS/OSS Contacts. (83)

As could be expected the majority of the respondents is using GNU/Linux as operating system, at least that is the information the Internet browser is revealing. Nevertheless, 16% were using Windows when answering the questionnaire. Whether this is a deliberate choice or just the computer used at work cannot be determined for sure although the former seems rather improbable.



Graph 15: User OS. (80)

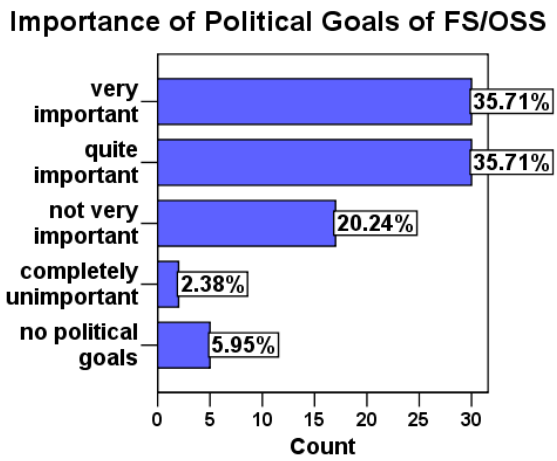
Chapter 4: Findings of Survey

4.1 Assessment and Motivation of Developers

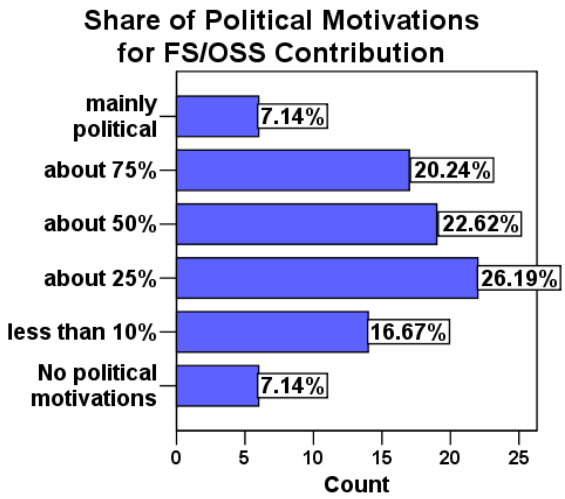
4.1.1 High Overall Politicisation in FS/OSS Community

"We have a great OS here, and even greater ideals behind it."

This quote of one participant expresses an approach to FS/OSS that understands it as more than just a technical matter and this approach is shared by the majority of the respondents in this sample. Only 6% of the questioned developers think that the FS/OSS community has no political goals while over 70% rate the political goals as quite or even very important to them. This high importance directly translates into a high political motivation for contributions to FS/OSS (correlation: 0.63**). Half of the developers state that at least 50% of their motivation is political. I would argue that also developers whose share of political motivation is about 25% are still considerably politically motivated given the variety of other motivations for contribution. Therefore about 75% of the developers have political motivations for their work on FS/OSS.



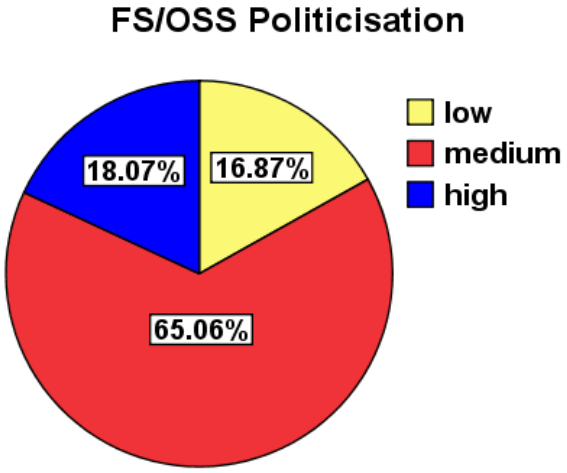
Graph 17: Importance of Political Goals. (84)



Graph 16: Political Motivation. (84)

If we set up an index to indicate the degree to which a developer politicises FS/OSS, it is again obvious that the vast majority of developers assigns FS/OSS at least some political importance. In this index, high FS/OSS politicisation stands for assigning political goals quite or very important and at the same time having stated a share of political motivation of 75% and higher. According to this index low politicised

developers are the ones who stated the political goals would not be very important for them if at all and whose political motivation is less than 10%. The rest is accounted for as medium politicised. With about one fifth of the sample being highly politicised and another fifth lowly or not at all, the majority exhibits an approach to FS/OSS that is driven by a perception of FS/OSS as being important for political reasons but not exclusively.



Graph 18: FS/OSS Politicisation. (83)

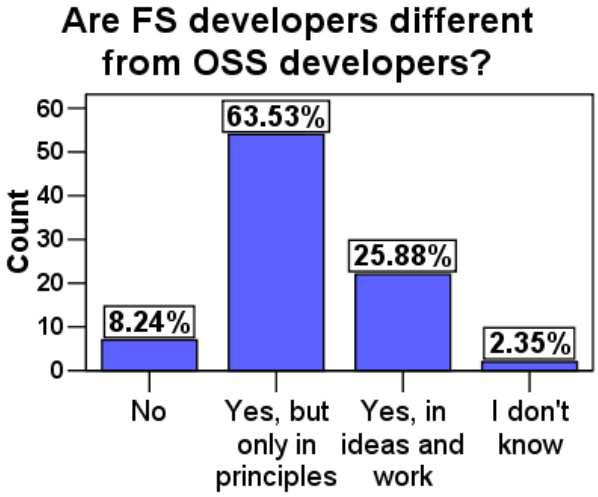
That political motivations do indeed exist in the community is also expressed by the reasons some of the respondents cited for their involvement in FS/OSS:

> *"the world is presently divided into haves and have nots i think floss i a great equalizer"*
>
> *"I feel that by releasing software that is useful to me, I can help my fellow man"*
>
> *"Spread knowledge of software programming around the world, especially to poorer peoples."*

4.1.2 "Free Software is Politics, Open Source is Economics"

The heading is actually a quote from one respondent and it confirms the finding of the this survey that the political and ethical goals of the community are largely attributed to the FS movement. However, the findings also add to the impression that the distinction gets less and less important. If we first have a look at how the differences

between both movements are perceived, then we see that while the vast majority of developers (89%) sees differences between Free Software and Open Source, the greatest part (63%) thinks these differences only exist in principles while the work is the same.



Graph 19: FS/OSS Differences. (85)

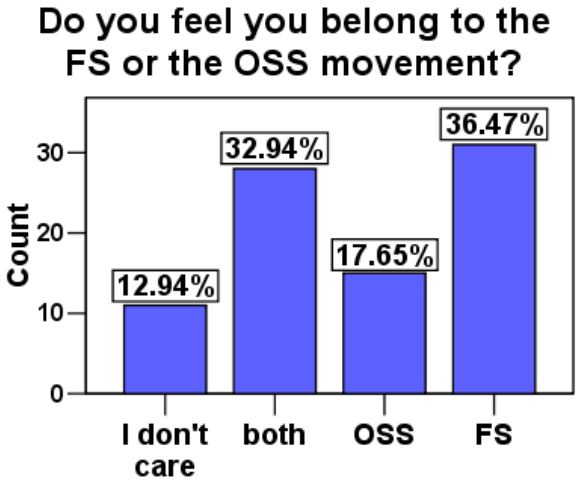
But what are the differences? The comments made concerning this distinction see the difference as it was outlined in the discussion of FSF and OSI: Free Software is the movement with a philosophical position that produces software not only out of technical but also out of ethical / social concerns while Open Source is less ideological and mainly aimed at producing qualitative software, especially to make it more attractive for businesses. This assessment is embodied in one respondent's statement:

> *"Open Source software is just a split of original free software movement, stripped from all political concerns in order to gain more broad recognition in business."*

Developers also mentioned very often that FS has an ethical or moral aspect:

> *"Freedom and ethics in software takes precedence over simply producing quality software."*
> *"I do not believe that Open Source software is a moral issue. I feel that this is the important difference between OSS and FS."*

But what do the developers prefer, politics or business? As the following diagram shows, more people favour the FS movement (36.5%) or say they belong to both (33%) while exclusive OSS members are rare (18%).



Graph 20: FS/OSS Membership. (85)

The FS members are the group who predominantly cited political or ethical reasons for the involvement with FS:

> *"I think I have a social role to play as a developer, so I consider this as being more a FS vision than an OS one."*
> *"Free software is politics, Open Source is economics. The end result is the same, but in this case, I prefer the politics."*

In contrast, one member of the Open Source movement voiced this view:

> *"The FS movement has some resemblance to Socialism, which is a system that I don't support"*

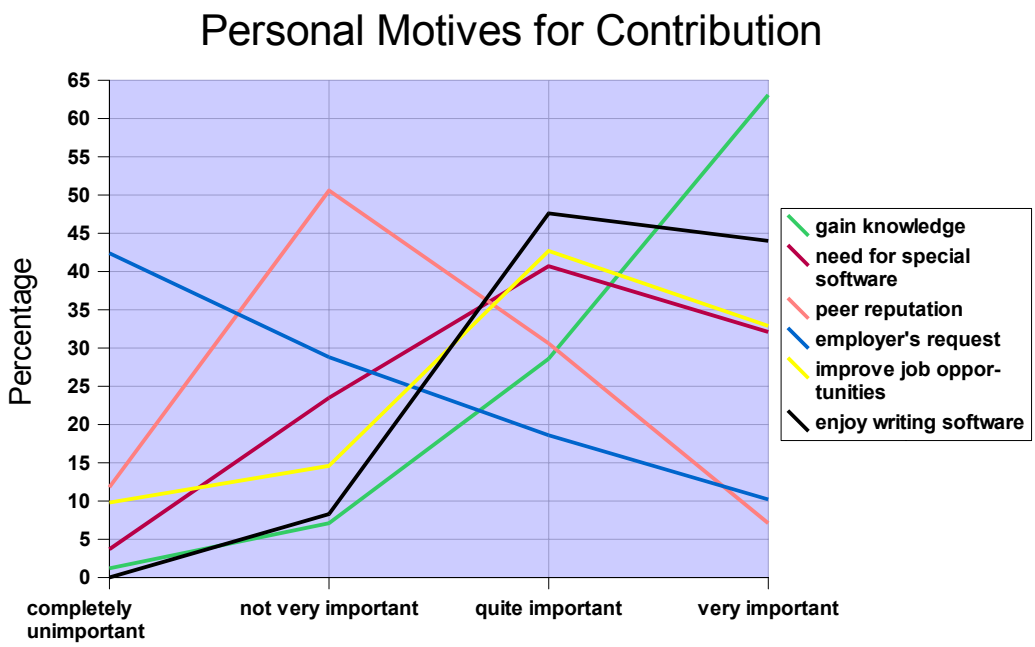
But whatever the differences, the data also reveals that the distinction is not that important for the developers. After all, for about half of the questioned developers, despite admitting differences, the distinction is irrelevant as they say they belong to both movements (33%) or they simply do not care (13%). Furthermore, to distinguish between FS and OSS is in danger of getting meaningless because what becomes clear from the qualitative data is that there exists considerable confusion about what the differences really constitute. Seven developers either interpreted free software as software that has to come free of charge or saw in OSS a development methodology of collaborative software production that would not be possible with free software. As a result, these developers have aligned with a certain movement while their approach to FS/OSS might have placed them better in the other group. Therefore the distinction between both movements gets blurred and subsequently increasingly unusable for the

researcher. However, what catches the eye is that these developers usually align with the OSS movement or with both movements, indicating that FS developers are not only more politically motivated but also better informed about the ideas of the community. That their beliefs are very strong is nicely illustrated by the fact that all the four negative remarks about the license used for releasing the collected data after the survey, came from members of the FS movement. They criticised that it is not a truly free license (as it does exclude commercial use) while others obviously did not care too much about this. Furthermore, the developers who belong exclusively to FS are also the group with the biggest share of people (42%) that see fundamental differences between both movements.

So despite the definition problems and even while about half of the developers do not want to align with one movement exclusively, the distinction is still helpful as it was shown that not only is FS generally seen as more political but also voiced the developers of the FS movement more often political motivations.

4.1.3 Politics just One of Many Motivations

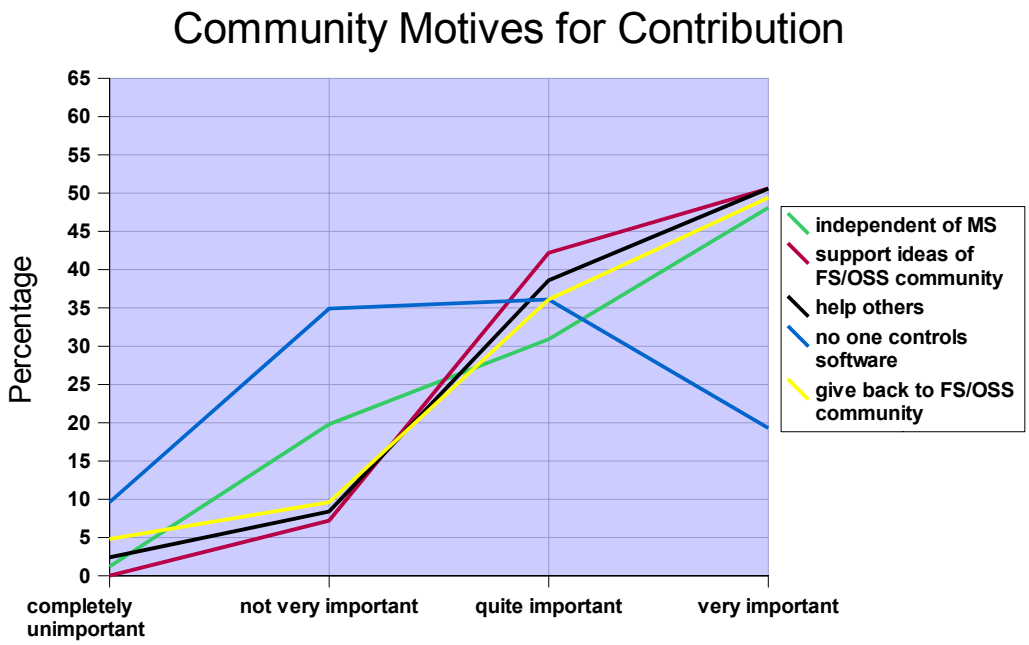
Despite the observed high politicisation of the movement, the developers also cited a lot of very practical motivations for their contribution to FS/OSS. In fact, the survey could confirm the findings of earlier studies that learning and increasing knowledge is the prime motivator for FS/OSS developers. More than 90% stated that this motivation is quite or even very important (mean 2.54). In the same way the finding of Lakhani et al. (2002) that fun and intellectual stimulation are very important can be observed here as the enjoyment from writing software is rated as highly motivating. Almost of equal importance is the improvement of job opportunities and the need for special software. The wish for peer reputation is rated low as more than 60% say this is not important. However, this might be difficult to determine by self-assessment as I would expect that this is a greater share of motivation than the developers admit. The least important personal factor for motivation was the request of the employer to produce FS/OSS, already signalling that even for the 40% share of paid developers writing FS/OSS is nothing they would have to be forced to.



Graph 21: Personal Motives for Contribution.

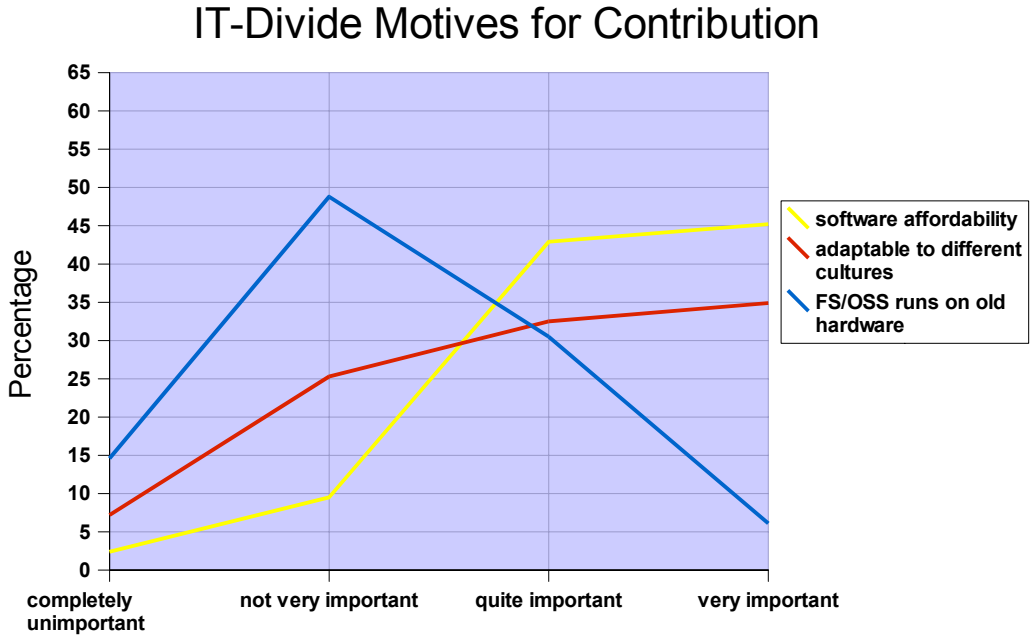
It would be wrong to conclude that the reasons for writing FS/OSS are solely *"selfish"*. As the next diagram shows, community motives have an equally big importance. About half of the sample is quite or even very motivated by the wish to give something back to the community, to help others with their knowledge and to support the FS/OSS community and its ideas. A motive that was assigned unexpectedly low importance (mean 1.65) was that no one would be in ultimate control over the software. This contrasts with the big motivational factor that the independence from huge software companies like Microsoft represents (mean 2.26). My interpretation is that while the objective of the former statement was to find out about the importance of freedom and independence of FS/OSS, it might have been perceived by some developers as a more negative statement about FS/OSS being a disorganized software development. I cannot find another explanation as I would have expected that both statements get similar ratings.

Finally, as this paper made claims about the importance of FS/OSS and GNU/Linux in particular for developing countries, it was asked for the importance of some features that are of special significance in these countries. Strikingly these motives were rated much lower in importance than the previously mentioned ones. The only motive that could reach a rather high importance was the affordability of FS/OSS. The



Graph 22: Community Motives for Contribution

two other statements – the adaptability to different cultural contexts and the possibility to run it on old hardware – got importance ratings that placed them in the least important third of all motivations (mean 1.95 and 1.28 respectively). This overall small concern for the demands of poorer countries might be a sign of the Western dominance in the community but it remains yet to be seen whether the assessment is different for developers from LMICs.



Graph 23: IT-Divide Motives for Contribution

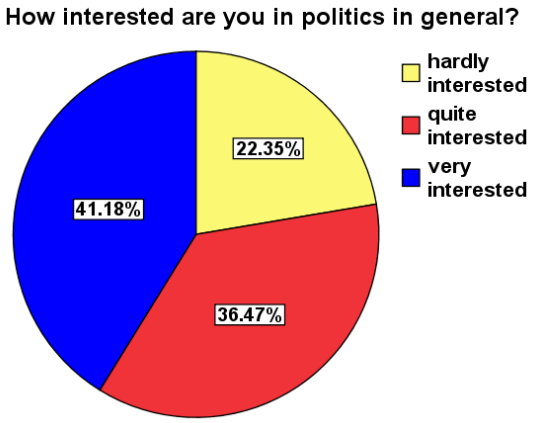
So are there motivational differences? After all, while most of the respondents in the sample were politically motivated, self-related motivations were rated as very important although not considerably higher than the most important community-related motivations. What is more, a comparison of the motivations of developers according to their FS/OSS politicisation shows that the only motivations mentioned significantly more often by highly politicised developers are the wish to support the community and its ideas along with the appreciation of the cultural adaptability of FS/OSS. What can be concluded is that the importance of self-related motivations does not signal an apolitical developer. Rather we have to understand politics as one among many different motivations that include gaining personal benefit as well as trying to help the community. This is also underlined by the qualitative data. While the following respondents all stated a high political motivation, they also cited very practical reasons for their involvement in FS/OSS:

> *"why make a hundred proprietary developers*
> *recreate the wheel 100 times over when*
> *we can make one wheel and have it be perfect ;)"*
>
> *"I want to create an easier environment to develop other applications,*
> *and free software is a good way to do it."*
>
> *"Believe that Open Source software commodtises [sic] functionality,*
> *allowing people to use this base to create new functionality, whether*
> *Open Source or proprietary, moving us forward."*

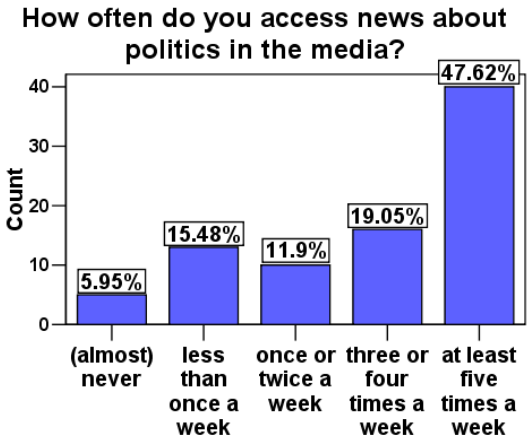
4.2 Political Participation Offline and Online

4.2.1 High General Political Interest and Activity

This survey did not only find a high politicisation of the community, it also showed that among the participants in this sample there is a high interest in politics in general: about three quarter of the respondents are quite (36.5%) or even very (41%) interested in politics and access political news in the media at least once a week but with the majority of them almost on a daily basis (correlation 0.759**).

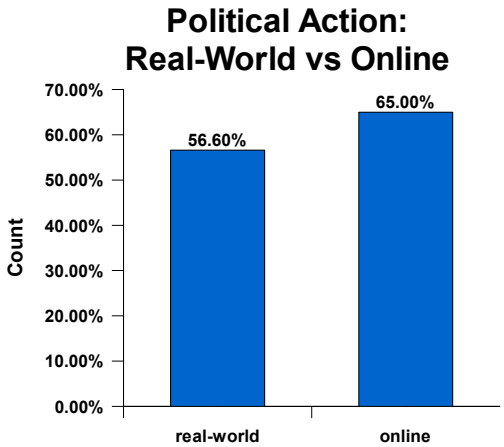


Graph 25: Political Interest. (85)



Graph 24: Political News Access. (84)

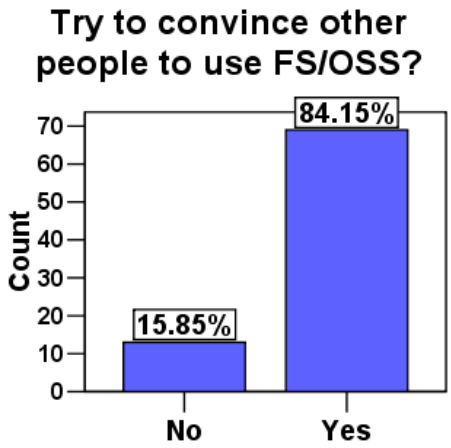
The survey also observed signs of political activities: More than half of the respondents have taken part in a protest action in "the real world" in the last half year and about two thirds did the same online.



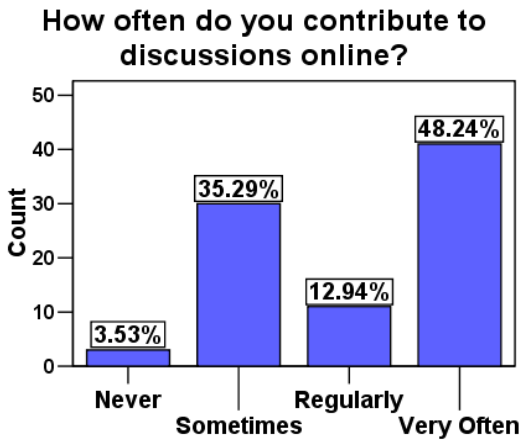
Graph 26: Political Action. (83/80)

Both offline (58%) and online (67%) software patents were the main target of protest. The findings support Coleman's (forthcoming) analysis that FS/OSS developers have been increasingly politicised by governmental actions that were perceived as threats to the community and its technology. But developers were also interested in other issues, although considerably less often: 15% of the developers protested against the war in Iraq while product boycotts, gender equality, gay rights and environment issues were named by one or two developers each. Furthermore, the big majority of the programmers stated that they try to convince other people to use FS/OSS. I would argue that, coupled with the political significance many programmers ascribe FS/OSS, this can

also be considered to be a political activity. It underlines the claim made about hackers aiming at influencing others by using technology. Finally, many developers are also active communicators, at least online with almost half of the respondents very often contributing to discussions online and 60% doing so at least regularly. Of course we do not know what is talked about in this forums and it will by no means be political in general but I would argue that the frequent contributors also communicate some of their values within these general discussions and therefore they can make an impact in contrast to the ones who do not take part in online discussions very often.



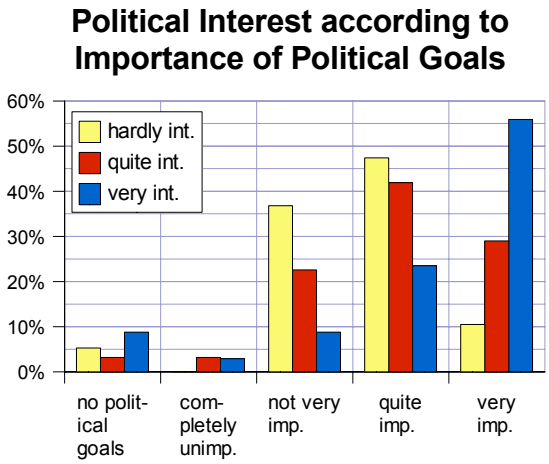
Graph 27: FS/OSS Publicity. (82)



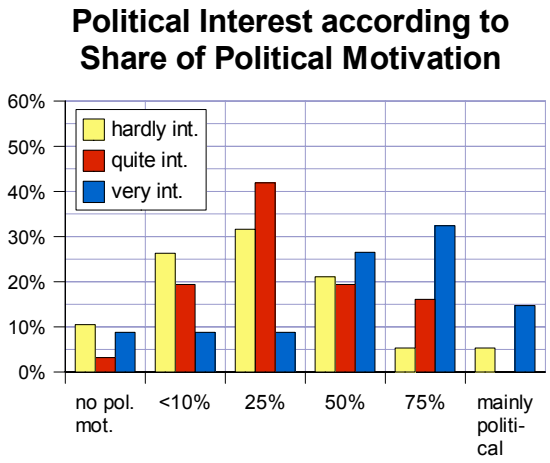
Graph 28: Discussion Contributions. (85)

4.2.2 Relation of Online and Offline Political Behaviour

Another aim of this survey was a clarification of the connection between political behaviour online and offline. What the results indicate is that a high FS/OSS politicisation (high importance of political goals and big share of political motivation) does not depend on a high general political interest. Therefore it would be wrong to conclude from a developer with an apolitical attitude concerning traditional politics that he or she does not attribute the contributions to FS/OSS a political relevance either. However, the results also make clear that a high interest in politics is usually accompanied by a stronger FS/OSS politicisation. This relation is expressed by a correlation coefficient that is significant though not too high: 0.348** for importance of political goals (if "there are no political goals" is excluded) and 0.328** for the share of political motivation for contributing to FS/OSS.



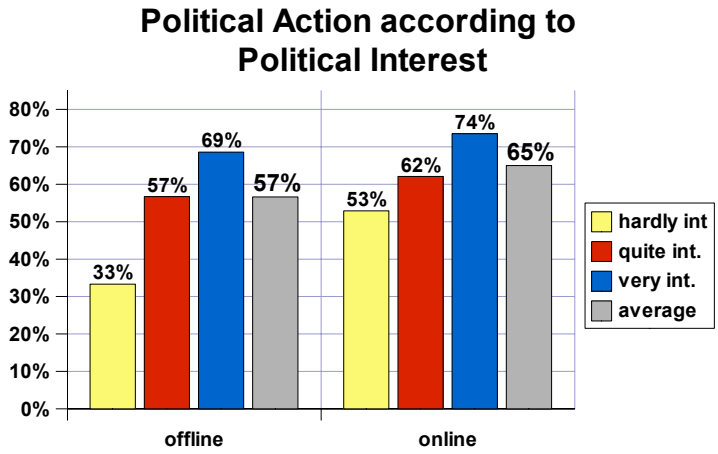
Graph 29: Political Goals by Political Interest. (84)



Graph 30: Political Motivation by Political Interest. (84)

The sample shows another interesting issue: three respondents who said they are very interested in politics in general argued that there would be no political goals in the FS/OSS community. I think that the opinion of these developers is of interest as it does not seem to represent a general apolitical attitude but a qualified assessment of politically interested people. Unfortunately the sample size is too small for analysing this group of programmers separately but it would be interesting so see whether this share of people is significantly increasing within a bigger sample.

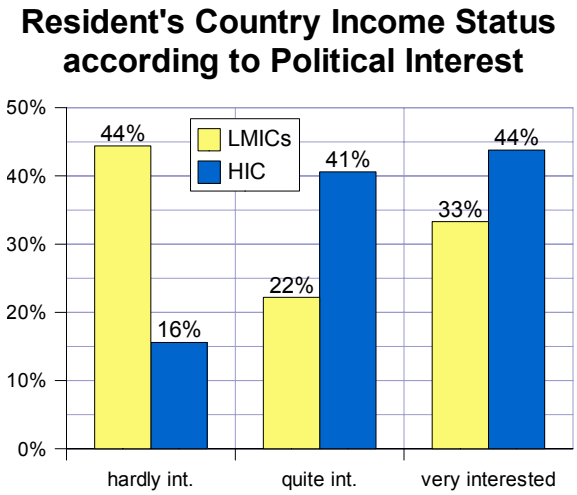
Similar findings for the influence of general political interest can be found for political protest action. While generally protest action increases with political interest, for the online realm protest action is not dependent on a high political interest.



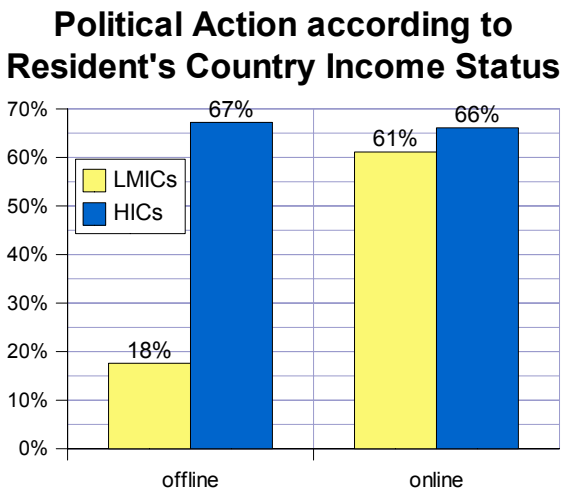
Graph 31: Political Action by Political Interest. (83/80)

Overall there are more developers taking actions online than offline. Especially dramatic is this difference for developers from LMICs: While only 18% did protest in

"the real world", more than 60% took part in protest actions online. Furthermore the example of LMICs again highlights that a general political interest is only of relevance for traditional forms of political action: While developers from LMICs are significantly less interested in politics than their HIC counterparts, they only differ in protest actions offline while the participation online as well as their political motivation is almost the same.



Graph 32: Income Status by Political Interest. (82)



Graph 33: Political Action by Income Status. (81/77)

One possible explanation for the lower offline politicisation of LMICs developers might be a different political situation that allows for less political participation in general. But it as well becomes obvious that the Internet is a medium that allows for a broader political participation, even of those people who are not too much concerned with "traditional" politics. What the results tell us is that programmers are often politically interested and this interest also has an impact on their FS/OSS politicisation. Yet one must not draw conclusions too quickly as this relation is only observable in one direction. A developer that is highly politically motivated for working on FS/OSS can still be completely uninterested in traditional forms of politics.

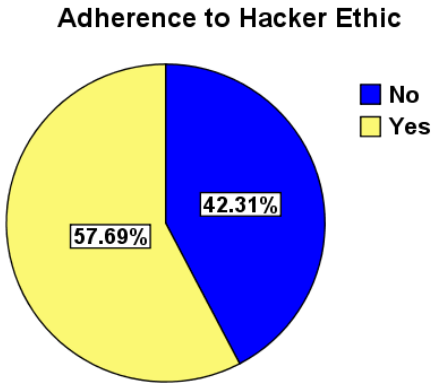
4.3 Characterising Political Developers

4.3.1 Hacker Values as Main Characteristic for the Politically Motivated Developers

> *"I know "geeks" are easily assumed to be ignorant to politics -- but I think it is not because they are geeks, but because they are relatively young, live in relatively politically-stable countries, studied sciences/technologies instead of human/cultural sciences, etc."*

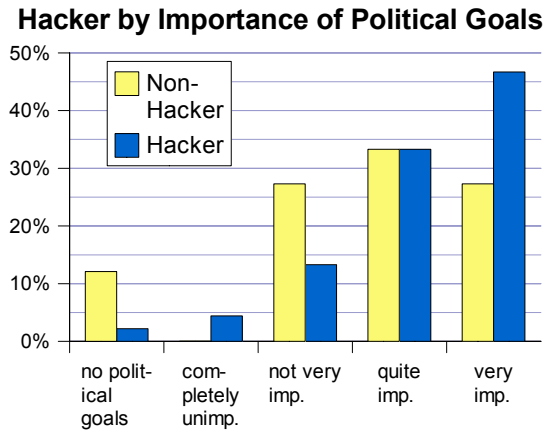
In contrast to this opinion voiced by one participant, this survey did not find significant evidence for the assumption that the politicisation of GNU/Linux developers is connected to their age or to the country they come from. While it was already shown that the developers in general are not ignorant of politics, I would argue that it is exactly these "geeks" who are among the most politicised according to this survey, at least if we understand "geeks" as hackers according to the hacker ethic that was discussed earlier.

In this survey, respondents were asked to state their approval with two statements that expressed the *Information should be free*-credo of the hacker ethic and its emphasis on open systems. The statements read: "*All information should be accessible to everyone."* and *"Generally, software should not be proprietary"*. According to the answers of the developers, it is possible to divide the sample into two groups, developers who strongly adhere to the hacker values (agree largely or completely with the two statements) and that I will label hackers (58%), and the ones who do not (42%). Apart from their assessment of the hacker values both groups exhibit the same characteristics, so the hacker ethic is nothing that would only appeal to students, programmers from LMICs or paid developers.

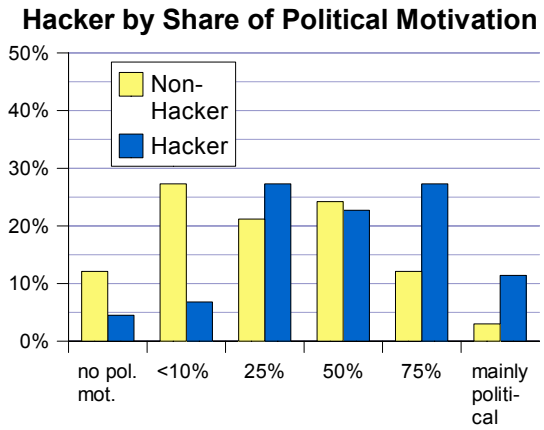


Graph 34: Hackers. (78)

Whether one believes in the hacker values or not is a significant indicator for the degree to which the FS/OSS activities are perceived as political. The following two diagrams show that the political goals of FS/OSS are more important for hackers (correlation: 0.239*, mean difference hacker vs. non-hacker significant*) and accordingly political motivations make for a higher share of hackers' motivation than for non-hackers (correlation: 0.320**, mean difference significant**).

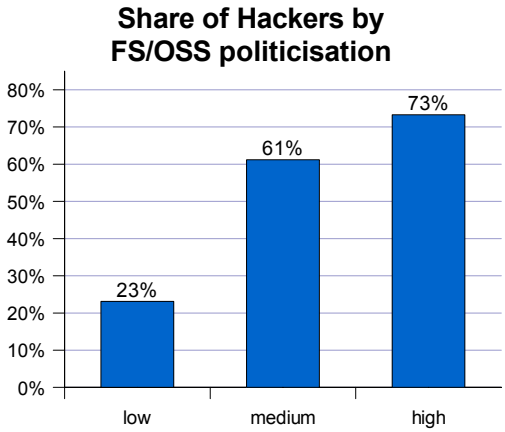


Graph 35: Political Goals by Hacker. (77)



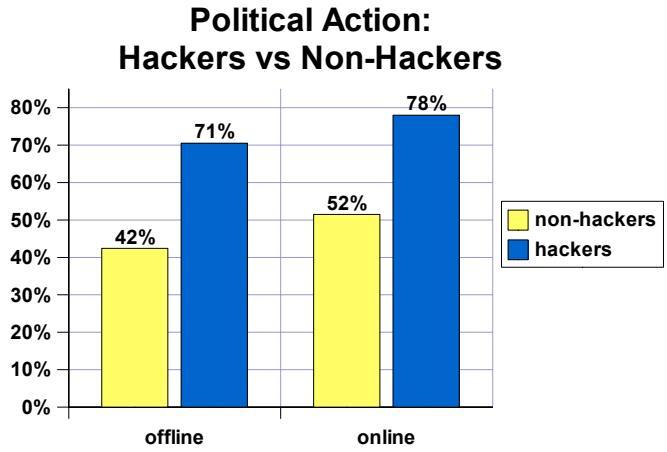
Graph 36: Political Motivation by Hacker. (78)

Subsequently, if we distinct the developers according to the degree of their FS/OSS politicisation, it gets obvious that hackers account for the major share of the more political developers while the group which attributes FS/OSS (almost) no political significance consists predominantly out of non-hackers.



Graph 37: Hacker Share by Politicisation. (77)

The same higher attribution of political issues can also be observed for protest action: Hackers tend to act more often than developers who do not adhere to the hacker values, both offline (correlation: 0.281*, mean difference significant*) and online (correlation: 0.279*, mean difference significant*).



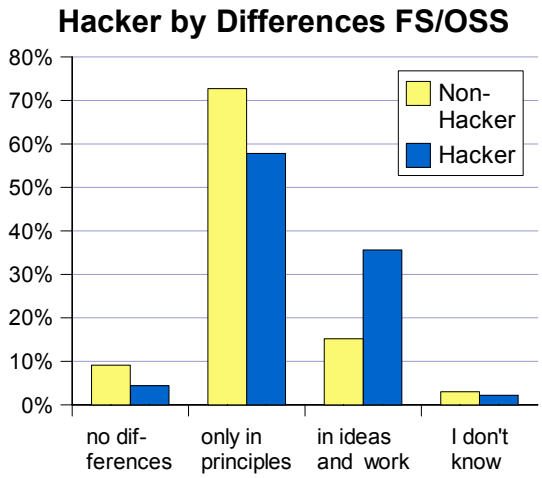
Graph 38: Political Action by Hacker. (77/74)

Finally, the hackers might have a higher FS/OSS politicisation but obviously there is no difference in all the other motivations for contribution, again highlighting that for highly politically motivated developers self-related motivations have the same relevance as for less politically motivated developers.

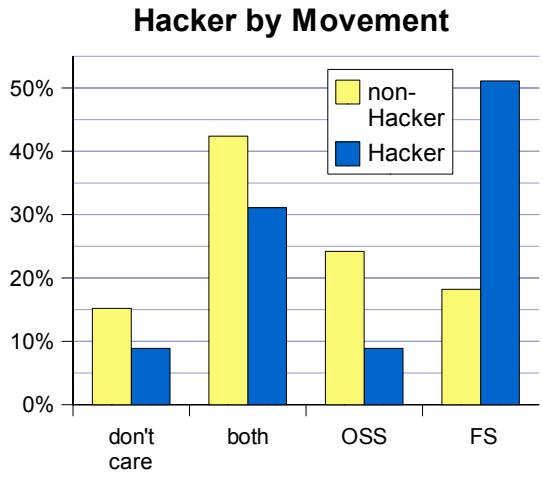
The results tell us several things: First, despite its openness for various motivations, the majority of the FS/OSS community still consists of people that once formed the movement: programmers with certain values, first and foremost freedom of information and openness of computer systems. Second, hackers tend to cite more often political motivations for their involvement with FS/OSS than non-hackers although it is important to make clear that this survey can of course not prove a causal relationship between hacker values and political motivation. Third, the results also give us a hint for how to assess hacking in general in terms of political motivation. However, the historical relationship of hackers and political participation has yet to be analysed.

This section ends with an overview about how hackers align with the two movements FS and OSS. First, there is again an indication of a stronger politicisation of the hackers as they do more often tend to see major differences between the two movements. The majority of the hackers aligns with FS that obviously suits their political ambitions better, leading to the FS movement having the highest share of

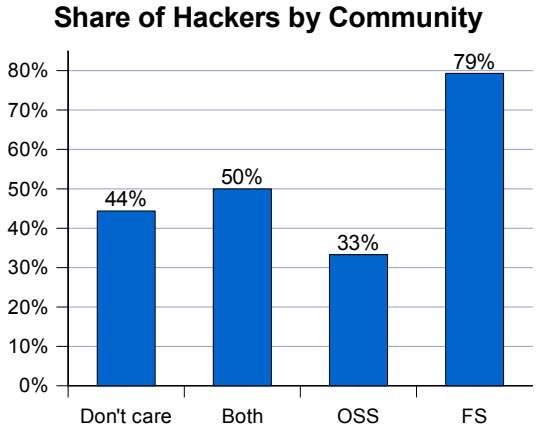
hackers (79%), compared to OSS with the lowest share (33%). It will be the issue of the next section to analyse whether these differences between both movements are also expressed in a different degree of politicisation.



Graph 40: Differences FS/OSS by Hacker. (78)



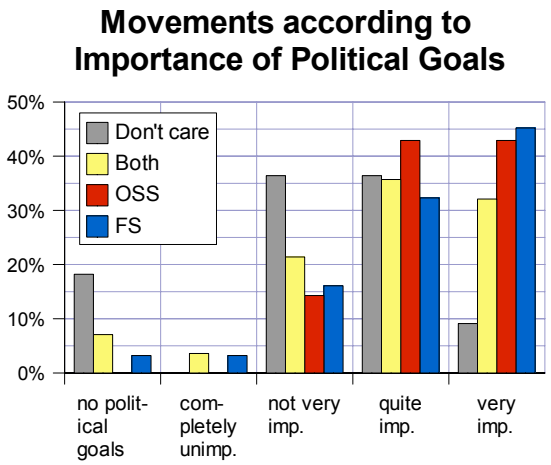
Graph 39: Movement by Hacker. (78)



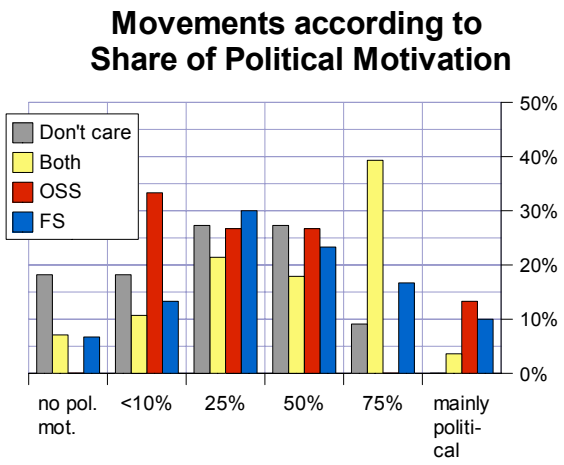
Graph 41: Hackers by Community. (78)

4.3.2 Little Observable Differences between the Movements

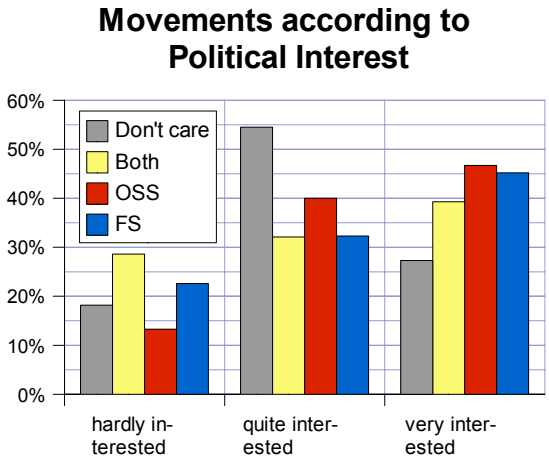
The earlier sections already made the case for FS as the openly political movement with politicised developers, in contrast to OSS. Despite of these findings, a comparison of both movements shows that there is surprisingly little difference between FS and OSS in their assessment of the importance of the political goals. What is more, while the FS movement shows a stronger political motivation, the difference to OSS is not really significant. There are also no obvious differences in general political interest.



Graph 42: Movements by Political Goals. (84)

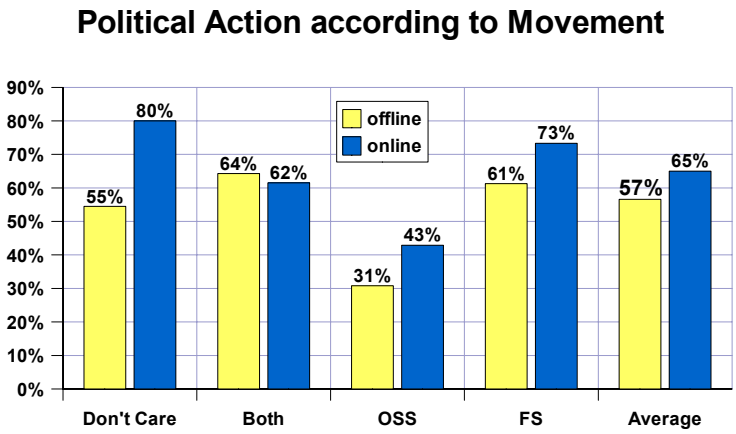


Graph 43: Movements by Political Motivation. (84)



Graph 44: Movement by Political Interest. (85)

So far, there is only some quantitative evidence for a higher politicisation of FS. The diagram below shows how many percent of the developers of each movement took part in a protest action offline and online respectively. It points out clearly that FS members in this sample are about twice as much likely to act, both in the real as well as in the virtual world. Despite the low case number the mean difference for FS versus OSS is almost significant. (significance 0.067 and 0.069 for offline and online respectively).



Graph 45: Political Action by Movement. (83/80)

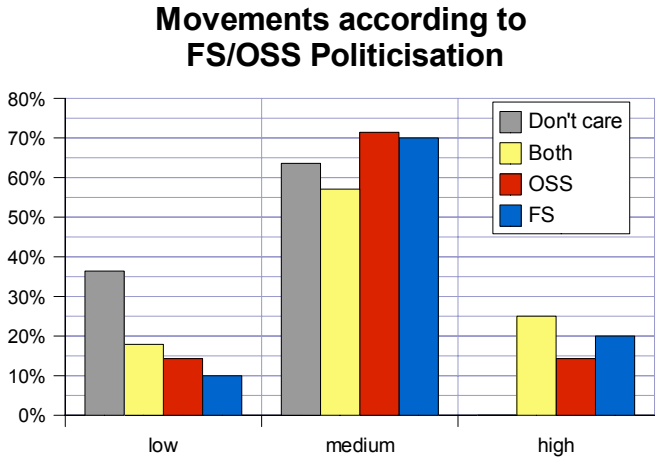
So overall, despite of the developers' own assessment of FS as the more political movement along with a considerably higher share of politically motivated hackers in contrast to OSS, there are almost no significant differences in the political motivation of both movements. I see two main reasons for the obvious equality: First, the comparatively small sample size. Differences are getting less significant if the sample of 85 cases is divided into four subgroups, leaving the number of OSS developers in this sample with only 15 developers. Second, as the analysis of the qualitative material indicates, of these 15 developers four have selected the OSS movement for the wrong reasons as they did not clearly understand the difference between the two. For example one of the OSS developers stated that:

> *"Free Software means provide software for free, but not the code.*
> *I believe it's important to have free code."*

This is clearly wrong and his arguments could place him equally well (or even better) in the FS movement. The findings for the other three respondents are similar and as a result the differences between both movements get blurred. The case of the four respondents is especially relevant as all four assigned the political goals high importance and were rather highly politically motivated with 25% or 50%. Placing them in another movement could make the difference in the overall comparison. With reference to the results of the qualitative data and relating the contrasting quantitative findings to the mentioned problems, I would conclude that the FS movement can be expected to see

programming as more than just a technical activity and to consider FS/OSS as socially relevant, in contrast to OSS developers. However, more enlightenment can only come from an increased sample size.

Having reviewed the differences for FS and OSS, what about the two other groups? While the ones saying they belong to both movements are usually representing the average for most of the variables, the ones who say they do not care for the movements are a group with distinctive properties. That they do not care is expressed in the fact that 27% of them saw no difference between FS and OSS compared to an average of 5% for the other three groups. Concerning the politicisation of their FS/OSS contributions, they attribute significantly less importance to the political goals of the movement (mean "don't care": 2.18, mean for the other three groups together: 3.08, mean difference significant*). The results are similar for the share of political motivation (mean "don't care": 1.91, mean for the other three groups together: 2.58). Additionally this group is less often very interested in politics than the other developers. The comparison for the politicisation of the different groups then also yields corresponding results. Developers who do not care for the movements are the ones who are most often not politicised and none of them was highly politicised.



Graph 46: Movement by Politicisation. (83)

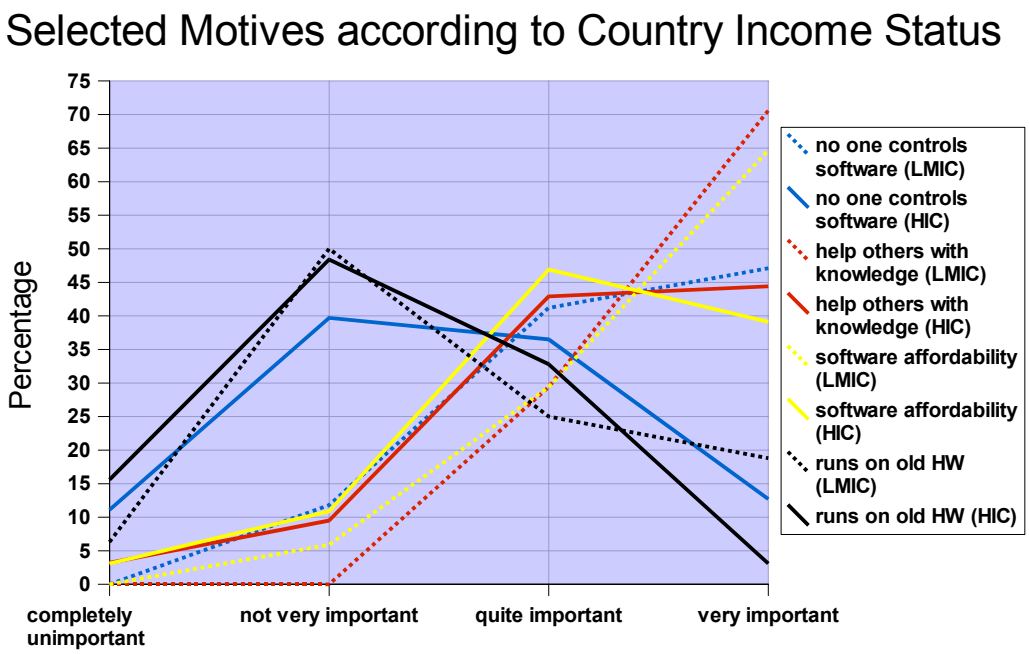
Concluding this section I would say that while with the quantitative data of this sample there might be some problems to prove a higher politicisation of FS developers compared to OSS developers, one thing is rather clear: the developers who do not care about aligning with a certain movement, also often do not care about politics.

4.3.3 No Real Differences for Low/Medium Income Countries

"the world is presently divided into haves and have nots i think floss [Free/Libre and Open Source software] is a great equalizer"

It was outlined earlier that FS/OSS is of particular relevance for poorer countries as it can help to overcome the digital divide and this quote of a developer from India expresses this approach to FS/OSS. However, in their political motivation for FS/OSS, developers from LMICs do not differ very much from their colleagues from HICs. The political goals of the community are assigned equal importance (mean LMICs: 2.94, mean HICs: 2.98) and the findings are similar for the share of political motivations (mean LMICs: 2.89, mean HICs: 2.51).

Differences can be observed only for traditional political participation that is lower for LMICs as was previously discussed, and some community-related motivations. Developers from LMICs have a stronger motivation to help others which can be interpreted as a stronger social approach of programmers from LMICs. Similarly, the higher appreciation of the independence of FS/OSS has its source in the specifics of LMICs as it was already mentioned that GNU/Linux is often seen as an alternative to US-based companies that are perceived as taking over the countries software industry. Of the three motives that were expected to be of special relevance to LMICs only the software affordability is assessed differently. In line with this is a general stronger approval of creators' rights to get paid (mean LMICs: 3.56, HICs: 3.08, mean difference significant*). I would interpret both findings as reflecting the more serious economic situation in these countries. However, the share of paid/non-paid developers in LMICs was comparable to HICs.



Graph 47: Motives by Country Income Status.

So overall, despite the different backgrounds of developers from LMICs and HICs, the reasons for contribution and the political motivations are quite similar. This might indicate that computer programmers in general constitute a rather homogeneous group and country-dependent differences could be moderated by the fact that developers especially in LMICs are a privileged group of society.

4.3.4 Paid Developers no Less Motivated than Volunteers

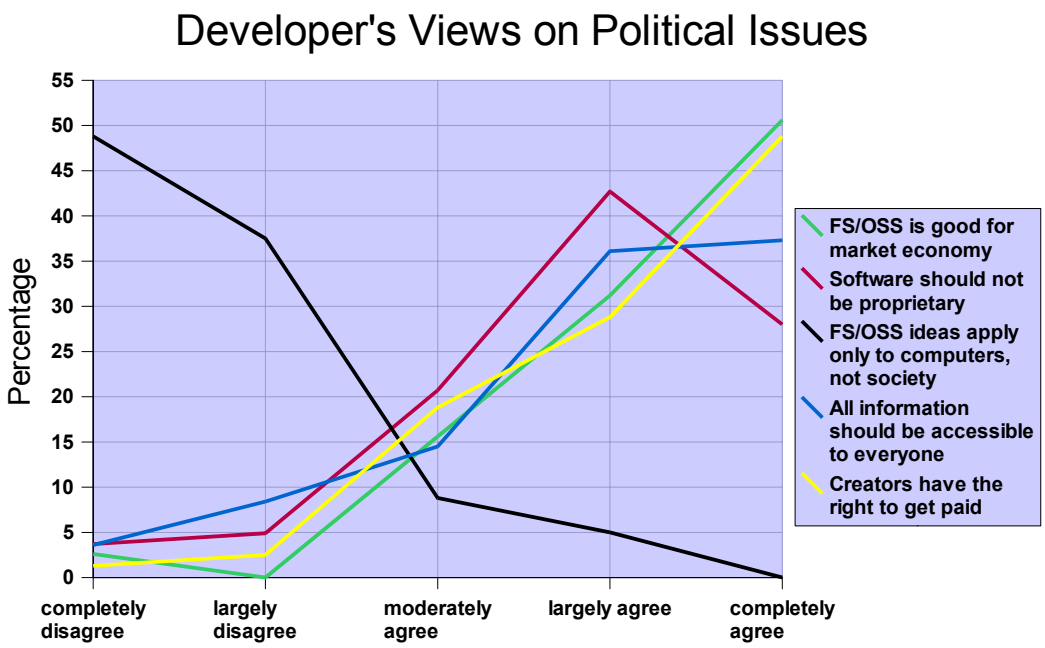
> *"My main criteria when I chose my current job was that I would only have to do development of GNU/Linux-software."*

This quote from the survey gives us a hint why it was not possible to find significant differences in the FS/OSS politicisation of developers who are paid for their work on FS/OSS (development, administration or support) and the ones who are not. The assessment of the importance of political goals and the share of political motivations are very much alike. Also the general political interest corresponds. These results make sense if we acknowledge that the majority of the paid developers works with FS/OSS not because they are paid to do so but out of passion for FS/OSS – see the quote earlier. This is nicely illustrated by the finding that even for the paid developers the request of the employer is only of little (but not significant) higher importance for their motivation compared to the volunteers (mean paid developers: 1.16, mean

voluntary developers: 0.73). In fact, there are no great differences for these two groups of developers overall, except that for volunteers gaining new knowledge (mean 2.66) and helping other people (mean 2.53) are slightly more important than for paid developers (mean 2.32 and 2.15 respectively) which reflects at least a bit the work-nature of paid developers' involvement in FS/OSS.

4.4 Political Views of FS/OSS Developers

This research has shown that for the majority of the developers in this sample political motivations play a role for their involvement in FS/OSS. Yet it is not clear what kind of ideas the programmers connect to it. Are their ideas in line with some of the more radical proponents of FS/OSS like the Indymedia activist who was cited earlier with *"free software is a revolutionary tool"*? Can we think of FS/OSS developers as revolutionaries that see the FS/OSS ideas of sharing and collaboration as alternative concepts for a capitalist system, or should we rather see them as being generally content with the established political system but with some proposals for improvements? The graphic below illustrates the developers' assessment of five statements that bring up some first clues.



Graph 48: Political Views.

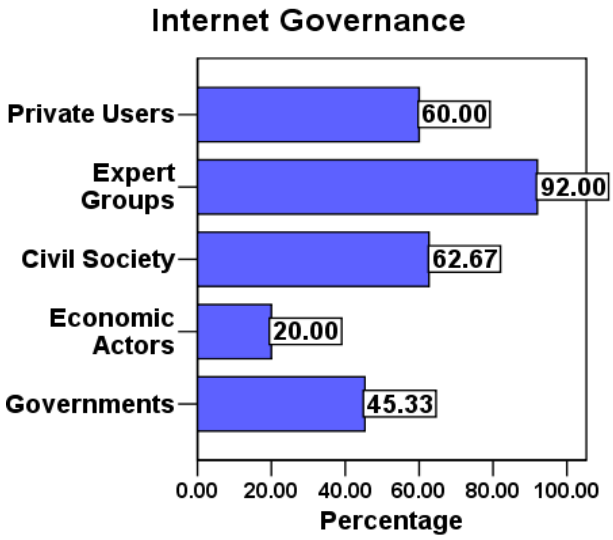
First of all, as we would expect from developers that see FS/OSS as politically significant, most of them do not think that the ideas of FS/OSS would be limited to

computers: 85% of the developers state the concepts can also be applied to society. Yet this does not signal that developers are meant to change the economic system and share everything freely, as their assessments of the two market related statements tell us: "FS/OSS is a healthy component of a free market economy" and "Creators have the right to get paid" got the highest average approval of all the statements (means 3.27 and 3.21 respectively). What is more, this rating does not change with the degree of politicisation. Therefore I would conclude that 80% of the developers (who largely or completely agree to the first statement) see FS/OSS and free market capitalism as compatible. Of course, to think FS/OSS is a healthy component of a free market economy does not express one is entirely satisfied with capitalism and the way it works but it is scarcely the opinion we would expect from someone who is very critical about contemporary society. Furthermore, the two economic statements obtain their significance out of the fact that the two other statements which express values of the hacker ethic and promote a view more in contradiction to established economic and politic structures, were approved less (mean 2.95 for "Information should be accessible to everyone" and 2.87 for "Software should not be proprietary"). These findings add to the impression that while the FS/OSS community might have some new ideas for society, the developers in general are not largely critical of the established structures and do not fit into the category "counter-cultural" as it is traditionally defined.

Also revealing is a look at the aims stated for the protest actions developers took part in: While there is a variety of different aims like gender equality, gay rights and opposition to the war in Iraq, the majority of the actions (offline, 58%; online 67%) was related to FS/OSS, in particular software patents. So as political actors the developers are very much focused on their own community. However, I think it would be wrong to assume that these political efforts are undertaken solely out of selfish motives and for the sake of the technology. Rather I would see it at least partly as an attempt to secure an idea and a technology that is considered to be beneficial for society as the findings concerning the applicability of FS/OSS confirm.

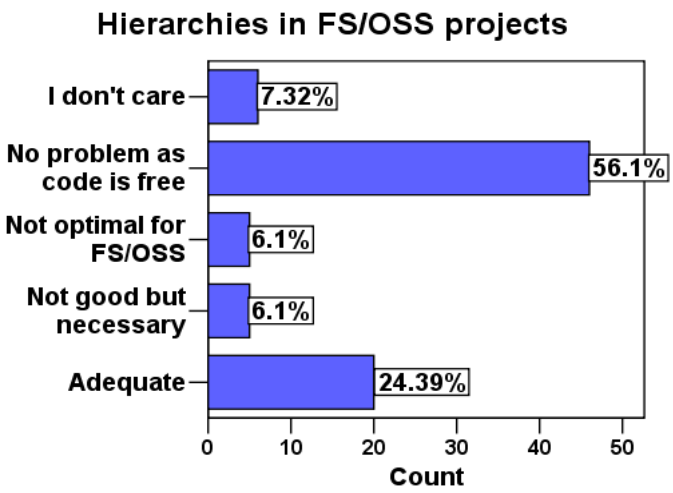
The question about who should govern the Internet was introduced in order to obtain information on whether developers favour more democratic or more elitist approaches. The overall comparison shows that the programmers do very much approve

the authority of expert groups. However, only 12% want to have expert groups in charge exclusively. The majority wants to see the governance of the Internet in the hands of private users, civil society and expert groups but with a strong emphasis on the latter. The views do not really differ with the degree of politicisation, except that highly politicised developers prefer a stronger role of civil society institutions. While the less important role of governments is in line with libertarian opinions promoted by prominent hackers such as Stallman or Raymond, the very low rating for a responsibility of economic actors is somehow in contradiction to the previously expressed general acceptance of the market. If there is nothing wrong with the free market economy, why should economic actors not have an influence on the Internet?



Graph 49: Internet Governance. (75)

The general appreciation of expert knowledge corresponds with an uncritical assessment of the hierarchies in FS/OSS projects. Although for example Bezroukov (1999) is highly critical of the "benign dictatorship" of project maintainers, the majority of respondents does not see the hierarchies as a problem. The critics of this type of organization are clearly in the minority (12%).



Graph 50: FS/OSS Hierarchies. (82)

This survey could only tap a bit into the pool of political opinions of FS/OSS developers and so one cannot infer too many conclusion from it. Despite of this I think it became obvious that FS/OSS developers cannot be considered as politically radical or as serious critics of the present capitalistic society, however FS/OSS and GNU/Linux in particular might be perceived by non-developers. After all we also have to acknowledge that computer programmers do benefit a lot from the established structures as their skills are in high demand. Yet, as was discussed earlier, the concepts of FS/OSS are clearly alternative and this survey shows that the respondents are convinced of their applicability to society. Just it is difficult to answer from the data how a society that incorporates the principles of FS/OSS should look like. What is more, I believe that probably the developers themselves do not have an exact idea of this alternative society as the observed contradiction in the developers' statements indicates. I would put that down to what I earlier called a missing theoretical framework. The ideas that do not come out of the developers' practical experience might be incoherent at times, given the programmers' preference for practice instead of theory.

Chapter 5: Conclusion

"We have a great OS here, and even greater ideals behind it."

This quote of one respondent wraps up many of the results of this research in one short sentence. While there is on the one hand the appreciation of the technology of "*a great OS*" along with all the practical benefits that come together with its collaborative production such as learning and fun in programming, for the majority of the developers in this sample there is something *even greater* beyond the technology: *ideals*. The results of this survey reject the thesis of a vastly apolitical community of FS/OSS programmers. On the contrary, the majority of the developers in this sample obtain a considerable motivation for their work out of the fact that they attribute FS/OSS a political significance. Underlining this general political understanding of FS/OSS is the finding that the majority of the respondents believes that the ideas of the community are applicable to society in general, not only to the realm of computers. What is more, their convictions also induce them to take action both online and in "the real world", especially if they see their community and their ideas threatened as was (and still is) the case with software patents for instance. The survey could also observe a high general interest in traditional forms of politics that are usually accompanied by a high FS/OSS politicisation, but as was shown, a highly political motivation for FS/OSS is not dependent on a general political interest. In this sense the online and offline world are separated with the politicisation in and for the technological realm being higher than for traditional forms of politics.

One might be surprised to learn that it is neither simply the volunteers in contrast to the paid developers who attribute special importance to the political goals of the community, nor the developers from low/medium income countries with special incentives for the use of FS/OSS. Instead the most politicised in their work with FS/OSS are the developers who can be considered as hackers in the definition according to the hacker ethic. If any this survey showed that the self-assessment of the hackers is extremely different from how they are typically thought of by the public: as asocial computer geeks or even worse – criminals. But together with the programmers who see

themselves as part of the Free Software movement, which is by the majority still perceived as the political centre of the community, the hackers form the political conscience of the community.

An understanding of software as more than just a technical entity is considered to be important because in this work I made the case for the profound importance of software for society and the important implications that arise out of the use and distribution it allows. I argued that developers play an important role in defining this implications both as actual creators of software as well as informed citizens of an information-centred society. Furthermore, I attributed FS/OSS particular political relevance as it challenges established assumptions about production and property, for which it has already been embraced by various actors with political connotations. In contrast to some authors who see the politicisation of this kind of software solely originating in the objectives of its users, I would argue that the survey showed that the developers themselves politicise FS/OSS. Despite of this, those political activists who see FS/OSS as a means for a post-capitalist society that emphasises collaboration and sharing, will be disappointed though: the little data on political opinions this survey could gather does not point in the direction of revolutionary or counter-cultural opinions among the developers, as expressed in the finding that most respondents see FS/OSS as a component of a free market economy rather than as an alternative to it.

On the other hand, I also argued that hackers have a very special approach to politics that mediates political ideas and participation through technology. This approach itself and the results it might yield could be much more radical than the developers themselves are actually aware of. In this issue I see the connecting point for continuing research: What I think is crucial for future studies is to further examine the political views developers connect to FS/OSS. As was argued, computer programmers lack a coherent theoretical background and so often there might not be sufficient self-awareness for the ideas one promotes and their incoherence, illustrated by the contradictory statements of the respondents in this sample. Therefore I would argue it will be necessary for further research to rely more on what developers actually do, for example in the form of content analysis of developers' online discussions as well as what kind of information they access and which news are important on community

relevant websites and why. Additionally, as the success of other researchers suggests (see for example Coleman and Hill), participant observation, for instance at developers conferences, also seems to yield useful material.

Reviewing my own research process, my general assessment is that the project succeeded in reaching its objectives. Especially the online questionnaire met my expectations: It reached a great number of people from diverse backgrounds, also from low/middle income countries and the response rates were impressive. The questionnaire itself was well perceived as the respondents' positive comments about the study underline. Furthermore, while much work was involved in setting up the online form, subsequently a lot of effort could be saved as it provided the answers already coded and in digital format. Furthermore the qualitative material was very important. I would argue that every research misses a great opportunity by relying solely on closed questions. Even if there might be problems to analyse this kind of data statistically, it provides the researcher with a useful context of the respondents' quantitative answers.

However, I do see room for improvement in the extent of the questionnaire. A stronger focus on the main topic with less general questions would have been advantageous as the complexity of the data interpretation and with it the amount of work grows exponentially with the number of questions asked. Besides, I would have wished to get more insights into the nature of the developers' political ideas, although this information is difficult to gather, especially by self-assessment. I already outlined what steps could be taken by continuing researches to overcome these problems. Despite of this I think a survey was the right choice as it produced the basic stock of information on which future research can built up.

The question that remains is in how far the results of this survey can be applied to the whole community. Clearly the findings cannot be said to be valid for the overall population as the selection of the respondents did not happen randomly and the sample size is too small, especially if we consider the discussed diversity of motivations for contribution. Furthermore, this survey focused on a certain group of developers who collaborate on GNU/Linux distributions. On the other hand, the demographic characteristics of the sample correspond mainly to the ones found for much bigger studies. However, even if we allow for a biased sample as the survey might have

attracted more politically interested developers, these political developers are part of the community and with them are their *even greater ideals*. At least these people prove everyone wrong who would state that FS/OSS developers are completely apolitical. Yet, I do not think that even the big majority is apolitical. One major finding of this survey is that the hacker values play an important role in the politicisation of the community. Who adheres to them is likely to see his or her work as more than just a technical activity. These values have formed the roots of the Free Software movement and they have endured in the community despite a growing popularity of its ideas that attracted the most contrasting contributors (think of IBM and Indymedia). I can think of nothing that would prevent the hacker ethic from its further existence, so that it might live on in the community and with it a socially responsible approach to computer programming. Nevertheless, a research project that wants to produce considerably reliable findings for the whole community must include more participants. A first step has already been made by applying principles of FS/OSS to this research and by putting the whole data of the sample with its 200 participants to the free use of others, suggesting that the FS/OSS principles are indeed useful and can be so even beyond their application for technology.

Bibliography

Aldermann, John (2002): *Sonic Boom. Napster, MP3, and the new Pioneers of Music.* London: Fourth Estate.

Alvestrand, Harald (2004): *The Linux Counter*. Available at: http://counter.li.org [09.08.2004]

Berners-Lee, Tim with Mark Fischetti (2000): *Weaving the Web : the past, present and future of the World Wide Web by its inventor.* London : Texere.

Bezroukov, Nikolai (1999): A Second Look at the Cathedral and Bazaar. In: *First Monday* 4(12). Available at: http://firstmonday.org/issues/issue4_12/bezroukov/index.html [01.08.2004]

Bonaccorsi, Andrea and Cristina Rossi (2003): *"Comparing motivations of individual programmers and firms to take part in the Open Source movement. From community to business"*. Available at: http://opensource.mit.edu/papers/bnaccorsirossimotivationlong.pdf [28.06.2004]

Bryman, Alan (2001): *Social Research Methods.* Oxford: University Press.

Castells, Manuel (2000): *The rise of the network society.* 2nd ed. Oxford: Blackwell Publishers.

Castells, Manuel (2001): *The Internet Galaxy.* Oxford: Oxford University Press.

Coleman, E. Gabriella (*forthcoming*): The Political Agnosticism of Free and Open Source Software and the Inadvertent Politics of Contrast. In: *Anthropology Quarterly.* Available at: http://www.healthhacker.com/biella/aaa_paper_final.rtf [06.07.2004]

Coleman, E. Gabriella and Mako Hill (2004): How Free Became Open and Everything Else Under the Sun. In: *M/C: A Journal of Media and Culture*, 7. Available at: http://www.media-culture.org.au/0406/02_Coleman-Hill.html [10.08.2004]

Coleman, E. Gabriella, and Benjamin Hill (*forthcoming*): The Social Production of Ethics in Debian and Free Software Communities: Anthropological Lessons for Vocational Ethics. In: Koch, Stefan (ed.): *Free/Open Source Software Development*, (pp. 271-293). Hershey: Idea Group Publishing.

Diamond, David (2003): The Peacemaker. How Linux Torvalds, the man behind Linux, keeps the revolution from becoming a jihad. In: *Wired* 11(7). Available at: http://www.wired.com/wired/archive/11.07/40torvalds.html [13.07.2004])

DiBona, Chris, Sam Ockman and Mark Stone (1999): *Open Sources. Voices from the Open Source Revolution.* Cambridge: O'Reilly

van Dijk, Jan (2000): Models of Democracy and Concepts of Communication. In: Hacker, Kenneth L. and Jan van Dijk, eds (2000): *Digital democracy: Issues of theory and Practice.* London: Sage.

Doctorow, Cory (2004): *Microsoft Research DRM talk.* Available at: http://www.dashes.com/anil/stuff/doctorow-drm-ms.html [23.06.2004]

Everard, Jery (1999): *Virtual states: the Internet and the boundaries of the nation state.* London: Routledge.

Gay, Joshua (ed) (2002): *Free Software, Free Society: Selected Essays of Richard M.* Stallman. Boston: GNU Press. Available at: http://notabug.com/2002/rms-essays.pdf [20.04.2004]

Gehring, Robert A. and Bernd Lutterbeck eds. (2004): *Open Source Jahrbuch 2004. Zwischen Softwareentwicklung und Gesellschaftsmodell.* Berlin: Lehmanns Media. Available at: http://www.think-ahead.org [08.07.2004]

Ghosh, Rishab Aiyer (1998): "Cooking pot markets: an economic model for the trade in free goods and services on the Internet". In: *First Monday 3(3)*. Available at: http://www.firstmonday.dk/issues/issue3_3/ghosh/ [28.06.2004]

Ghosh, Rishab Aiyer with Ruediger Glott, Bernhard Krieger and Gregorio Robles (2002): *"Free/Libre and Open Source Software: Survey and Study"*. Available at: http://www.infonomics.nl/FLOSS/report/ [27.06.2004]

Ghosh, Rishab Aiyer (2003): Licence fees and GDP per capita: The case for open source in developing countries. In: *First Monday 8(12).* Available at: http://www.firstmonday.org/issues/issue8_12/ghosh/index.html [11.05.2003]

Ghosh, Rishab Aiyer (2004): *Why developing countries need to use and create Free Software.* Presentation at First African Conference on the Digital Commons. Available at: http://www.infonomics.nl/FLOSS/papers/200401/idlelo-GHOSH.ppt [13.07.2004]

Hannemyr, Gisle (1999): *Technology and Pleasure. Considering Hacking Constructive.* In: *First Monday* 4(2). Available at: http://firstmonday.org/issues/issue4_2/gisle/index.html [11.05.2004]

Heywood, Andrew (2002): *Politics.* 2nd ed. Basingstoke: Palgrave.

Holtgrewe, Ursula (2004): Heterogene Ingenieure – Open Source und Freie Software zwischen technischer und sozialer Innovation. In: Gehring & Lutterbeck (2004)

IDC Press Release (08.10.2003): *IDC Says Microsoft Is Moving into Dominant Role in Server Operating Environments, Even as Linux Grows.* Available at: http://www.idc.com/getdoc.jsp?containerId=pr2003_09_29_140158 [26.04.2004])

Lakhani, Karim R., Robert G. Wolf, Jeff Bates and Chris DiBona (2002): *Boston Consulting Group / OSDN Hacker Survey.* Available at: http://www.osdn.com/bcg/ [29.06.2004]

Lakhani, Karim R. and Robert G. Wolf (2003): *Why Hackers Do What They Do: Understanding Effort in Free/Open Source Software Projects.* MIT Sloan School of Management. Available at: http://freesoftware.mit.edu/papers/lakhaniwolf.pdf [01.07.2004]

Lancashire, David (2001): Code, Culture and Cash: the fading altruism of Open Source Development. In: *First Monday* 6(12). Available at: http://firstmonday.org/issues/issue6_12/lancashire/index.html [28.06.2004]

Levesque, Michelle (2004): Fundamental issues with open source software development. In: *First Monday 9(4).* Available at: http://firstmonday.org/issues/issue9_4/levesque/index.html [13.07.2004]

Levy, Steven (1994[1984]): *Hackers: heroes of the computer revolution.* London: Penguin.

Meretz, Stefan (2000): *LINUX & CO. Freie Software – Ideen für eine andere Gesellschaft.* Wasserburg: AG SPAK Bücher. Available at: http://www.kritische-informatik.de/fsrevol.htm [12.07.2004]

Moody, Glyn (2001): *Rebel Code: Linux and the Open Source Revolution.* London: Penguin

Raymond, Eric S. (1998a): A Brief History of Hackerdom. In: DiBona et al: *Open Sources*, p.19-29

Raymond, Eric S. (1998b): The Revenge of the Hackers. In: DiBona et al: *Open Sources*, p. 207-219

Raymond, Eric S. (1999): *The Magic Cauldron.* Available at: http://www.catb.org/~esr/writings/magic-cauldron/magic-cauldron.html [12.07.2004]

Raymond, Eric S. (2000[1997]): *The Cathedral and the Bazaar.* Available at: http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ [22.04.2004]

Reijswoud, Victor van and Corrado Topi (2003): *Alternative Routes in the Digital World: Open Source Software in Africa*. http://opensource.mit.edu/papers/reijswoudtopi.pdf [13.07.2004]

Register, The (25.09.2003): Linux taking off – and being taken off. In: *The Register.* Available at: http://www.theregister.co.uk/2003/09/25/linux_taking_off_and_being/ [13.07.2004]

Riemens, Patrice (2003): Some thoughts on the idea of 'hacker culture'. In: *Multitudes 8.* Available at: http://multitudes.samizdat.net/article.php3?id_article=1030 [06.07.2004]

Robles, Gregorio with Hendrik Scheider, Ingo Tretkowski and Niels Weber (2001): *Who Is Doing It? A research on Libre Software developers.* Available at: http://ig.cs.tu-berlin.de/s2001/ir2/ergebnisse/OSE-study.pdf [27.06.2004]

Salin, Phil (1991): *Freedom of Speech in Software.* Available at: http://www.philsalin.com/patents.html [02.07.2004]

Schiller, Dan (1999): *Digital capitalism: networking the global market system.* Cambridge, Mass. ; London : MIT Press.

Slashdot.org (01.03.2003): U.S. Army's Future Combat System Will Run Linux. In: *Slashdot.org.* Available at: http://slashdot.org/articles/03/03/02/0216215.shtml [01.07.2004]

Stallman, Richard M. (1996): The Free Software Definition. In: Gay, Joshua: pp. 41-44

Stallman, Richard M. (2001): Free Software: Freedom and Cooperation. In: Gay, Joshua: pp. 155-186

Stallman, Richard M. (2002a): The GNU Project. In: Gay, Joshua: p. 15-30

Stallman, Richard M. (2002b): Why "Free Software" is Better than "Open Source". In: Gay, Joshua: p. 55-60

Stewart, Katherine J. and Sanjay Gosain (2003): *Impacts of Ideology, Trust, and Communication on Effectiveness in Open Source Software Development Teams*. Available at: http://opensource.mit.edu/papers/stewartgosain.pdf [29.06.2004]

Tuomi, Ilkka (2002): *Networks of innovation : change and meaning in the age of the Internet.* Oxford : Oxford University Press.

United Nations, Department of Economic and Social Affairs, Population Division (2004): *World Urbanization Prospects. The 2003 Revision. Data Tables and Highlights.* Available at: http://www.un.org/esa/population/publications/wup2003/2003WUPHighlights.pdf [09.07.2004]

United Nations Development Programme (2003): *Human Development Report 2003 of the UNDP*. Available at: http://hdr.undp.org/reports/global/2003/ [07.07.2004]

Wagner, Michael G. and J. Randall Jue (2003): *The Open Source Myth. Why Executive Should Think Twice about Making Company Source Code Public.* Available at: http://citeseer.ist.psu.edu/554044.html [12.07.2004]

Weber, Karsten (2004): Philosophische Grundlagen und mögliche Enwicklungen der Open-Source- und Free-Software-Bewegung. In: Gehring & Lutterbeck (2004)

Youngs, Gillian (2001): Information and Communication Technologies. In: Boyd-Barrett, Oliver, Chris Newbold and Hilde van den Bulck (2001): *The Media Book,* p. 382-390. London: Arnold.

Zimmermann, Thomas (2004): Open Source und Freie Software – soziale Bewegung im virtuellen Raum?. In: Gehring & Lutterbeck (2004)

Appendix

A.1 Explanation of Computer Terms

binary	A binary is a computer file that contains code that only makes sense for computers, basically zeros and ones.
bug	An error in software. Basically it is used for anything that differs from the software's expected behaviour.
compiler	A compiler is a special computer program that is usually used to translate *source code,* written in a high-level programming language, into a low-level language, that can be interpreted by a computer.
Digital Rights Management (DRM)	From Wikipedia: Digital Rights Management or Digital Restrictions Management (DRM) is an umbrella term for any of several arrangements by which the usage of a copyrighted digital work can be restricted by the owner of the rights to the work.[xii]
free software	The FSF says: Free software is software that comes with permission for anyone to use, copy, and distribute, either verbatim or with modifications, either gratis or for a fee. In particular, this means that source code must be available.[xiii]
GNU	Project started by Richard M. Stallman in 1984 to create a free software alternative to the operating system UNIX. The GNU (Gnu is not Unix) project tried to develop all the software necessary for an UNIX operating system itself. While the project was very successful, Linus Torvalds was faster in developing the last missing bit, the necessary kernel.
GNU/Linux	Often simply named Linux, it refers to an operating system that consists of the Linux-kernel and free software from the GNU project as well as from numerous other sources. The FSF favours the term GNU/Linux because the Linux-kernel is actually just one component of a whole free operating system that Stallman and his fellows were creating through the GNU project.

GNU/Linux distribution

To make a computer usable it needs more than just the kernel, it needs software that actually allows the user to carry out work with this computer. These programs are often called applications. Examples would be a word processor or software to edit digital photographs. For the Linux-kernel there exist innumerate of these applications, most of them as FS/OSS. As most users do not want to be bothered too much by searching themselves for all kinds of different software and to configure it, different institutions offer GNU/Linux distributions. These software packages include not only the Linux-kernel along with the GNU-software but a lot of other software that is preconfigured and easy to install. Providers of GNU/Linux-distributions can be both commercial (see for example Suse[xiv]) and non-commercial (see for example the Debian project[xv]). While the software contained in the distributions is usually available free of charge, the user is paying the commercial providers for its service of creating preconfigured software packages and providing support.

kernel

A kernel is the core of every operating systems, the basic piece that coordinates the hardware. However, a kernel alone is of limited use. In addition to the kernel there are a lot of other software programs necessary to form a full operating system.

Linux

Talking about Linux today, in most cases means talking about an operating system with a Linux-kernel and free software from GNU and other sources. The FSF champions the usage of the term GNU/Linux as it emphasises their big contribution to the operating system. See therefore GNU/Linux.

operating system	The basic software that performs the task of coordinating all the hardware in a computer (for example the microprocessor, the hard drive or the keyboard) and provides an interface for all other applications to interact with the computer's hardware. For example the operating system controls the graphics adaptor and makes sure that the output of an application (say a software video player) is displayed correctly on the screen. Nowadays operating systems themselves offer applications by default as for example an Internet Browser (see Microsoft's Internet Explorer) or a word processor.
peer-to-peer network / P2P	Networks in which a number of computers function both as clients and servers in contrast to a client-server-model where there is a number of dedicated servers to which clients connect. The advantage of P2P networks is that they do not rely on a centralized network infrastructure as every computer fulfils the same function.
proprietary software	Software which comes without source code and which the user is not allowed to modify or redistribute.
public domain	If something is in the public domain, it means that the original author does not claim any copyright on his work. Therefore it can be used by everyone for any purpose, even to take the original work and claim copyrights on it. In the case of software, public domain does not necessarily mean that the source code of the program is available.
software	From Wikipedia: Computer software is a generic term for organized collections of code representing instructions executed by a computer.[xvi]
source code	Source code is the text that a programmer is writing. He uses special terms of a computer language to describe the action the program should carry out. This text is then translated by a *compiler* into instructions the computer can understand.

A.2 Why Access to Source Code Is Important

To understand why it is important to have access to the source code in order to be able to modify software, one must have a closer look at the process of programming. The language a computer does understand is called machine code, basically a combination of zeros and ones. It is almost impossible for humans to write a program in machine code or to understand it. Instead programmers write software in a programming language that is much easier to handle. The resulting text is commonly referred to as *source code*. Afterwards this text is translated by a special computer program, called compiler, to assembly language. The result are so called binaries, code that the computer understands but that does not make much sense to humans.

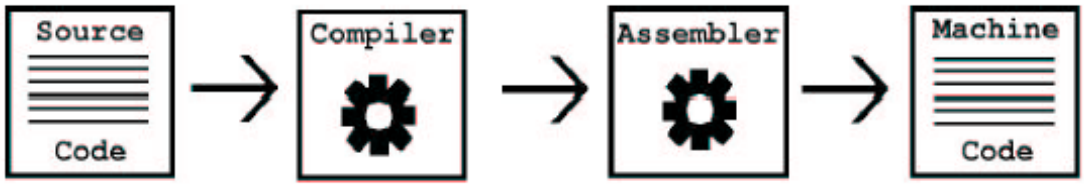


Illustration 3: Compilation Steps.

Nowadays software usually comes in binary format. The binaries are later again translated by the computer itself into the final machine code. What is basically happening in the whole process is that the source code is translated from a high-level language which can be more or less easily understood and manipulated by humans, to a low-level language which can only be understood by computers. Until now there exists no reliable means to reverse the process in order to get source code from the binaries. So to be able to modify a program the source code must be available. The graphic illustrates the process and two examples give an impression of the fundamental differences between source code and machine code (graphic and examples taken from: Gay, 2002: pp. 3-4):

source code	***machine code***
int main(){	11000111101110101001010010010010 1
printf(”Hello World!”);	01101010100110000011110010110101 0
return 0;	01001111111111100101101100000000 1
}	01001000011001010110110001101100 0

A.3 Questionnaire

The following pages contain the questionnaire in plain text format. Please refer to the survey web page at http://www.chronovault.net/websites/tobi/researchID/ (user name and password *guest*) to access the online version of this questionnaire that was deployed for the survey.

FS/OSS Involvement

page 1 of 9

The beginning of this survey is interested in the nature of your engagement with Free Software / Open Source Software development.
In this survey, the term "FS" stands for "Free Software" according to the definition of the Free Software Foundation. "OSS" is used as abbreviation for "Open Source software" according to the definition of the Open Source Initiative.
The term "FS/OSS" is used as an abbreviation to refer to all software that is either Free Software or Open Source software according to the respective definition.

devyears

How many years have you been developing FS/OSS?

allproj

In how many FS/OSS development projects have you been involved?

N/A

projects

Would you like to give an idea about the projects you have contributed to? You could state simply the name of the project(s) or what kind of software you were contributing to (ie. Desktop, hardware drivers, word processing etc.). As all other information in this survey, to answer this question is completely voluntary!

translat

Did you ever contribute translations of interfaces or documentations of FS/OSS software projects?

Yes No N/A

paidFSOS

Are you paid for developing, supporting or administrating Free Software / Open Source Software?

Yes No N/A

devwork

If you develop either proprietary software and/or FS/OSS on behalf of your employer or on behalf of your clients:
What kind of software do you develop at work?

N/A

initiate

Did you ever initiate yourself a FS/OSS project (for example by putting software you have written under a FS/OSS-license)?

Yes No N/A

FS/OSS Assessment

page 2 of 9

Two terms exist to describe software whose source code is open to everyone: *Free Software* and *Open Source Software*. There is some discussion about whether these terms describe different approaches and whether the differences are important. This section wants to learn how you assess this issue.

diffFSOS +

What do you think, are FS-developers different from OSS-developers?

- Yes, the ideas and the work are different
- Yes, but only in principles. The work is the same
- No, there are no differences
- I don't know what the difference is between FS and OSS
- N/A

commpart

Do you feel you belong to the Free Software movement or to the Open Source Software movement?

- Free Software movement
- Open Source Software movement
- I feel I belong to both
- I don't care
- I don't think there exists a community at all
- N/A

Would you like to give some reasons for the point of view you expressed by answering the previous questions?

reasons

What kind of contacts do you have to other FS/OSS developers?

friends

- ☐ Some FS/OSS developers are my friends
- ☐ I sometimes have general discussions about FS/OSS with other developers
- ☐ My communication with other developers concerns only the actual project
- ☐ I don't have contacts to other FS/OSS developers
- ☐ N/A

FS/OSS Motivation I

Your answers to the following questions shall provide a basis for assessing your motivation to contribute to FS/OSS projects. For each of the following statements would you please indicate how important it is for your developing-engagement in FS/OSS. This self-evaluation might not be easy. Though, this survey is interested in real motivations rather than in what seems to be desireable or appropriate. There are no *wrong* answers!
Please think of each statement as starting with: **I contribute to Free Software / Open Source Software because ...**

... I want to learn new skills and improve my knowledge.

knowledg

☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... FS/OSS enables you to be independent of large software companies such as Microsoft.

resistms

☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... FS/OSS is software that is affordable for everyone.

cost ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... I need a particular feature in an existing software or I need a special program.

persneed ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... I want to support the FS/OSS community and its ideas.

support ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... FS/OSS is adaptable to specific cultural needs, for example languages.

cultadap ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... I want to help others with my work and knowledge.

help ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

FS/OSS Motivation II

page 4 of 9

This page contains still some more questions about your motivations to develop FS/OSS.

I contribute to Free Software / Open Source Software because ...

... I want to gain reputation among peers.

reput ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... FS/OSS can be used with old hardware.

oldHW ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... my employer / my client requested to develop FS/OSS.

request ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... with FS/OSS no one is in ultimate control over the software and its functionality.

control ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... thereby I can improve my job opportunities.

jobgain ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... I receive so much from the FS/OSS movement (ie. software, knowledge etc.) that I want to give something back.

reciproc ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

... I simply enjoy writing software.

funprog ☐ completely unimportant ☐ not very important ☐ quite important ☐ very important ☐ N/A

If there are any other motivations you would like to mention, please feel free to state them in the box below.

motivtxt

Politics and Motivation

How important are the political goals of the FS/OSS community for you?

polimp

- ☐ completely unimportant
- ☐ not very important
- ☐ quite important
- ☐ very important
- ☐ there are no political goals
- ☐ N/A

polmot

If you think about your reasons to contribute to FS/OSS-projects: What would you say, how much of your motivation is of a political nature? (for example: freedom of information, equal access for everyone, against huge corporations etc.)

N/A

propagan

Do you actively try to convince other people to use FS/OSS?

☐ Yes ☐ No ☐ N/A

In FS/OSS-projects the maintainer has the ultimate control about which changes are incorporated into the code and which are not. What do you feel about these hierarchies?

hierarch

- ☐ The hierarchies are adequate
- ☐ In general hierarchies are not good but they are necessary for FS/OSS-development
- ☐ The hierarchies are not the best way to organize FS/OSS projects
- ☐ There is no problem because everyone can take the code and set up a new project
- ☐ I don't care
- ☐ N/A

Political Views

The following section is interested in learning about some of your political views and attitudes concerning FS/OSS and the Internet. For each of the following statements you are asked to indicate how much you agree with it by selecting one of these options:

- I completely disagree with it
- I largely disagree with it
- I moderately agree with it
- I largely agree with it
- I completely agree with it

FS/OSS is a healthy component in a free market economy (for example because it stimulates competition).

economy

☐ completely disagree ☐ largely disagree ☐ moderately agree
☐ largely agree ☐ completely agree ☐ N/A

Generally, software should not be proprietary.

resistps

☐ completely disagree ☐ largely disagree ☐ moderately agree
☐ largely agree ☐ absolutely agree ☐ N/A

The ideas of FS/OSS are only applicable to the realm of computer technology and not to other aspects of society.

FSOSlife

- ☐ completely disagree
- ☐ largely disagree
- ☐ moderately agree
- ☐ largely agree
- ☐ absolutely agree
- ☐ N/A

All information should be accessible to everyone.

shareinf

- ☐ completely disagree
- ☐ largely disagree
- ☐ moderately agree
- ☐ largely agree
- ☐ absolutely agree
- ☐ N/A

Everyone who is creating something (for example computer programmers or artists) has the right to get paid for his/her creation.

payart

- ☐ completely disagree
- ☐ largely disagree
- ☐ moderately agree
- ☐ largely agree
- ☐ absolutely agree
- ☐ N/A

Which parties should play a part in governing the Internet (check as many as you regard necessary)?

intgov

- ☐ Governments or Governmental Institutions (ie. United Nations)
- ☐ Economic Actors
- ☐ Civil Society (ie. non-governmental organisations)
- ☐ Expert Groups (ie. the World Wide Web Consortium)
- ☐ Private Users

Political Interest

page 7 of 9

The following part of the survey is concerned with your interest in politics in general and some of your political viewpoints. Please be ensured that the information you give is completely anonymous and that it is not possible to learn who gave this information.

What would you say, how interested are you in politics in general ? Think for example about voting in elections, discussing political topics with friends etc.

polint

- ☐ very interested
- ☐ quite interested
- ☐ hardly interested
- ☐ not at all interested
- ☐ N/A

newspol

How often do you access news about politics in the media (ie. newspaper, radio, TV, Internet, ...)?

N/A

protestr

In the last half-year: Did you take part in a kind of real-world political action like for example taking part in a demonstration, contacting a politican, signing a petition, boycotting a certain product etc.?

Yes No N/A

protrcat

If you answered the preceeding question with "Yes":
Can you please state the aim of the action? (for example "against war in Iraq", "against software patents", etc.)

protestv

In the last half-year: Did you take part in a political action on the Internet like for example signing an online petition, taking part in a virtual sit-in, setting up a campaigning website, writing an adovacy article for an online medium etc?

Yes No N/A

protvcat

If you answered the preceeding question with "Yes":
Can you please state the aim of the action? (for example "against war in Iraq", "against software patents", etc.)

disscont

How often do you contribute to discussions online (any topic), for example in chats or message boards?

N/A

Demography

page 8 of 9

Before this survey comes to an end, would you please answer some demographic questions? These questions are very general. Be again insured that it is not intended to figure out your identity with the information you provide.

resident

In which country do you live at present?

N/A

area *

What phrase describes best the area you live in?

N/A

How old are you?

age

N/A

Please indicate your gender!

gender

female male N/A

What is the highest qualification you achieved so far?

edu

N/A

What is your profession?

profess #

N/A

Are you currently

workstat

employed self-employed not in paid work N/A

Research Data

page 9 of 9

In order to apply the idea of Free Software/Open Source to this research, I would like to make the generated data publicly available in order to share the data of this survey with other interested people. The information you entered is completely anonymous and it is not possible to connect your identity with the data stored. Furthermore, a special license will be applied to the data that ensures that no one is using it for commercial purposes. (See the *Creative Commons Attribution-NonCommercial License* for more information about the license applied and the FAQ to get more information about the data that is stored.)

Would you agree with that the information you provided is released under the conditions stated above?

opendata

Yes No

Please feel free to comment on the survey in the box below. Is there something you missed, an issue you want to raise? Any criticism and advice is welcome as well as your thoughts concerning the topic of this survey. Thank you!

comments

A.4 Invitation Letter

Sorry, this is off topic:

Hello,
my name is Tobias Escher, I am a student at the University of Leicester in Great Britain and currently I'm researching for my Master-thesis which deals with the political motivations of GNU/Linux-programmers. Therefore I conduct an online-survey that is aimed at learning in detail about the political views and the motivation of software developers concerning their contribution to GNU/Linux. Independent of whether you consider yourself politically interested or not I would like to invite you to take part in this survey. It will take approximately 10 to 20 minutes to answer the questions.
This survey addresses everyone who contributes or has contributed code and/or ideas for components of a GNU/Linux-system (In more detail, as components I understand the Linux-kernel, GNU-software or any other free software / open source software that can be used on a system with a Linux-kernel).

The questionnaire is available online at:

http://www.chronovault.net/websites/tobi/researchID/

In order to take part you have to type in the user name XXXX and the password XXXX . This cannot be used for identification purposes! The survey is completely anonymous and not-for-profit.

If you have any questions please feel free to contact me or consult the FAQ (http://www.chronovault.net/websites/tobi/researchID/faq) at the site of the survey.

I would be very happy if you would consider to take part in this survey to help this small academic project. Please feel free to forward this letter to anyone who might be interested. Thank you very much.

With kind regards, Tobias Escher

PS: The questionnaire is drafted in English language. However, participants of all countries are highly welcome. Due the the small scale of this research a translation was not possible.

background:
My name is Tobias Escher, I am a student of the MA Globalization & Communications (http://www.le.ac.uk/cmcr/teach/pg/mcpgglob.html) at the University of Leicester (http://www.leicester.ac.uk), Great Britain. My supervisor is Dr. Gillian Youngs (http://www.le.ac.uk/politics/staff/gy4.html) at the Centre for Mass Communication Research. If someone wishes to contact me by paper mail, that can be done via the office of the Centre:
CMCR, attn. Tobias Escher
University of Leicester
University Road
LE1 7RH, Leicester, United Kingdom

A.5 Technical Issues Concerning the Online Questionnaire

The online questionnaire was realised with the Zope application server, a free software product[xvii]. The form itself was constructed using the Formulator framework by Martijn Faassen.[xviii] Several Python scripts were used to collect the data out of the questionnaire forms and store them in a plain text file. This text file was later imported into the statistic program SPSS. The complete set up of the questionnaire is available online at the address http://www.chronovault.net/websites/tobi/researchID (user name and password: guest) and can be used to configure online questionnaire to suit own purposes.

A.6 Online Resources for this Survey

The data collected by this survey is available at http://www.chronovault.net/websites/tobi/researchID/. (User name as well as password to access the site is *guest*.) The raw data is published under the Creative Commons Attribution-NonCommercial License 1.0, provided the participants have agreed to do so. That allows every non-commercial use as long as proper credit is given. Furthermore, a completely coded version of the raw data is available in the file format of SPSS.

A.7 Internet Addresses of Mentioned Projects

i Free Software Foundation. Available at: http://www.fsf.org [20.04.2004]
ii Wikipedia – The Free Encyclopedia Main Page. Available at http://www.wikipedia.org [30.07.2004]
iii Apache Web Server – Homepage. Available at: http://www.apache.org [11.08.2004]
iv The Mozilla Project – Homepage. Available at: http://www.mozilla.org [11.08.2004]
v Creative Commons – Homepage. Available at: http://www.creativecommons.org [10.08.2004]
vi MIT Open Course Ware – Homepage. Available at: http://ocw.mit.edu/ [10.08.2004]
vii Free Software and Open Source Foundation for Africa. Available at: http://fossfa.net [13.07.2004]
viii The IndLinux Project. Available at: http://www.indlinux.org [08.08.2004]
ix Open Source Initiative. Available at: http://opensource.org [20.04.2004]
x Mir – Content Management System. Available at: http://mir.indymedia.org [05.07.2004]
xi Association for Progressive Communications – Homepage. Available at http://www.apc.org [10.08.2004]
xii Wikipedia – The Free Encyclopedia: Digital Rights Management. Available at: http://en.wikipedia.org/wiki/Digital_rights_management [10.05.2004]
xiii Free Software Foundation: Categories of Free and Non-Free Software. Available at: http://www.gnu.org/philosophy/categories.html#FreeSoftware [21.04.2004]
xiv Suse Homepage. Available at: http://www.suse.com [28.07.2004]
xv Debian GNU/Linux Homepage. Available at: http://www.debian.org [28.07.2004]
xvi Wikipedia – The Free Encyclopedia (Revision as of 14:28, 23 Apr 2004): Computer Software. Available at: http://en.wikipedia.org/wiki/Software [24.04.2004]
xvii Zope Homepage. Available at: http://www.zope.org [27.07.2004]
xviii Formulator Homepage. Available at: http://www.zope.org/Members/infrae/Formulator [27.07.2004]