Mobilization of Software Developers: The Free Software Movement

Margaret S. Elliott
Institute for Software Research
University of California, Irvine
Irvine, CA 92697
949 824-8756
melliott@ics.uci.edu

Walt Scacchi
Institute for Software Research
University of California, Irvine
Irvine, CA 92697
949 824-4130
wscacchi@ics.uci.edu

ABSTRACT

Free/open source software (F/OSS) development projects are growing at a rapid rate. Globally dispersed virtual communities with large groups of software developers contribute time and effort often without pay. One force behind this phenomenon is the Free Software Movement (FSM), a 20 year-old social movement whose purpose is to promote the use of free software instead of proprietary software. We show how the ideology of the FSM influences software development work practices in F/OSS communities and how an occupational community of F/OSS developers has emerged from this movement. We present results from an empirical study of a F/OSS development community, GNUenterprise (GNUe) whose purpose is to build an Enterprise Resource Planning system. We show how the beliefs in freedom and freedom of choice, and the values of cooperative work and community building are manifested in the GNUe norms of informal self-management, immediate acceptance of fellow contributors, and open disclosure.

Keywords

Free software movement, computerization movements, occupational community, organizational culture, free/open source software development, virtual community

INTRODUCTION

Free/open source software (F/OSS) development projects are growing at a rapid rate. The SourceForge Web site reports 750,000+ users with 700 new ones joining every day and a total of 75,000+ projects with 70 new ones added each day. Globally dispersed virtual communities (Kollock and Smith, 1999) have formed with large groups of software developers contributing time and effort often without being remunerated. One force behind this emergent phenomenon is the Free Software Movement (FSM), a 20 year-old social movement whose purpose is to promote the use of free software instead of proprietary software.[1]

Kling and Iacono (1988) characterized a computerization movement (CM) as a mobilization force for widespread computerization, such as the artificial intelligence movement as well as the movements supporting ubiquitous internetworking and distant work (Iacono and Kling, 2001). Research by Kling and Iacono (1988) has shown that the ideology of these CMs paint a utopian picture of what companies and society will gain by widespread computerization. Computerization activists disregard the potential social problems accompanying computerization such as protection of personal privacy, quality of jobs, and others. The FSM is a different genre of CM in that it assumes widespread use of computerization, and promotes the production and use of only "free" software, not proprietary software. This paper explores how the FSM is attempting to revolutionalize software development practices by advocating that all software be "free" for access, study, modification, and (re)distribution. The FSM also admonishes the use of non-free software as immoral because it prevents its users from learning (about programming,

etc.) and prevents developers and users from helping their fellow man (Stallman, 2002; Williams, 2002).

In this paper, we show how the ideology of the FSM influences software development work practices in F/OSS communities and how an occupational community (Trice and Beyer, 1993; Van Maanen and Barley, 1984) of F/OSS developers has emerged as part of this movement. In previous papers, we presented the results of a qualitative study of the methods and social processes used in GNUenterprise[2] (GNUe), a free software development community with the goal of developing a free resource planning system (Elliott and Scacchi, 2003a; b). We showed how they jointly build community and a web of software system artifacts via instant message (IM) streams using internet relay chat (IRC), text-based records of IRC logs, mailing lists, and summary digests (Kling and Scacchi, 1982; Scacchi, 2002a; b). We captured the beliefs, values, and norms of the GNUe virtual organizational culture by using a grounded theory approach and by utilizing an organizational culture perspective (Martin, 2002; Schein, 1992; Trice and Beyer, 1993). In this paper, we show how beliefs and values of the FSM are manifested in the norms of GNUe software development practices – informal self-management, immediate acceptance of fellow contributors, and open disclosure of all documentation and work transcripts (IRC logs, mailing list archives, summary digests). In addition, we discuss how the norms reflect the ideology of the occupational community of F/OSS software developers.

In the remainder of this paper, we focus on the evolution of the occupational community of F/OSS software developers based on the ideology of the FSM. The paper is organized

in the following way. We discuss CMs and the FSM, and then we give a description of the GNUe research site. Following that, we present occupational communities and research methods. Next we outline FSM and GNUe beliefs, values, and norms and how the FSM has influenced the formation of GNUe norms. Finally, we end with conclusions.

COMPUTERIZATION MOVEMENTS

CMs are social movements whose activists promote mass computerization of specific computing technologies to bring about a new social order. Social movements can be defined as "collective enterprises to establish a new order of life" (Blumer, 1969:8). CMs can be characterized as both a general CM and specific CMs that can be considered sub-movements of the broader, general movement to computerize society (Kling and Iacono, 1988).

Kling and Iacono (1988) identified five specific computing technologies: urban information systems, artificial intelligence, computer-based education, office automation, and personal computing. They suggested that CMs communicate key ideological beliefs about the links between computerization and a preferred social order which help to legitimize computerization for potential adopters. The groups that form around a computer technology form a social movement with mobilizing ideologies that promote an improved social order or oppose a bad one; form organizations with a variety of membership; and promote the social movement via various communication modes and patterns of computing resource deployment.

Kling and Iacono (1988) analyzed these five CMs promoting specific computing technologies and found the following commonalities among their ideologies:

1) Computer-based technologies are central for a reformed world.
2) The improvement of computer-based technologies will help reform society.
3) No one loses from computerization.
4) More computing is better than less, and there are no conceptual limits to the scope of appropriate computerization.
5) Perverse or undisciplined people are the main barriers to social reform through computing.

Kling and Iacono (1988) found that these ideological themes help shape public images of computers and computerization as technological progress while deflecting competing social values. They claim that this vision includes computer users acquiring the best technology regardless of the cost and that those who do so will be considered the most virtuous. In another related article (Iacono and Kling, 1996), they discuss CMs in terms of a revolution in which the utopian genre of writing is used to portray all changes resulting from advanced computerization as "good". More recently, Iacono and Kling (2001) characterize the rapid growth of the Internet as a CM in which socially constructed processes of societal mobilization give rise to internetworking technologies. They contend that participants in these CMs advocate positive links between internetworking and a new, preferred social order. Iacono and Kling (2001) conclude that CMs play a role in persuading organizations to accept an ideology which promotes widespread adoption of internetworking technologies. However, in actuality, they predict that the CMs of the late

1990s will most likely result in a reinforcement of existing social orders, rather than widespread transformations as intended.

In this paper, we discuss the FSM, a social movement that has gained momentum in the last ten years in support for its vision of a society where all software is "free". Unlike CMs that proffer visions of widespread utopian computer usage, the FSM promotes the widespread use and development of free software, arguing that non-free software is immoral because it prevents its users from learning and from helping their fellow man. We discuss the beliefs and values of the FSM in the next section. We characterize F/OSS developers as an occupational community with an ideology based on the transformation of how people view, use and develop software.

FREE SOFTWARE MOVEMENT

The FSM refers to a social movement of activists who believe in and promote five principles of free software [3]:

1) Freedom to run a software program for any purpose;
2) Freedom to study how the program works and adapt it to their needs;
3) Freedom to redistribute copies of the software at will;
4) Freedom to improve the F/OSS program and to distribute the altered version;
5) Required distribution of the originating license that specifies the freedoms and rights concerning the preceding properties.

The FSM was envisioned in 1984 by Richard M. Stallman (widely known as RMS in F/OSS communities) when he began work on GNU software with the intention of sharing it with others as free software. RMS is a key figure in the free software movement (Williams, 2002; Stallman, 2002). In 1984, he started the GNU (GNU's Not Unix) project by developing and distributing a UNIX-like operating system as free software. This system has evolved into the GNU/Linux system using the Linux kernel combined with GNU utilities. The GNU project led to the formation of the Free Software Foundation (FSF) in 1985. The FSF is a tax-exempt charity whose purpose is to promote computer users' right to use, copy, modify, and redistribute computer programs. The FSF is dedicated to furthering the principles of free software with the goal of eliminating altogether the need to use proprietary systems and programs. The FSM has evolved from the beliefs of the FSF and is based on the principles of RMS who advocates the sole use of free software as a moral obligation (Williams, 2002).[4]

F/OSS development represents a relatively new approach to the development of complex software systems (Feller and Fitzgerald, 2002). Software development techniques used in F/OSS projects are informal and self-managed with decisions generally made by meritocracy (Scacchi, 2004; Mockus et al., 2002). In most situations, the resulting software system and its associated Web-based documents or development artifacts are globally accessible at little or no direct cost. The FSM and the OSI however differ philosophically on such issues as the use of proprietary software. One could almost consider the OSI to be a counter-FSM.

It is important to distinguish between the terms free software (Stallman, 1999) and open source software (DiBona et al., 1999) and their implications. For purposes of this paper, we reference both free and open as F/OSS. However, we focus the analysis on the free software community and the influence of the FSM on their daily work practices. In the free software community, free software refers to software that is open to anyone to copy, study, modify, and redistribute (DiBona et al., 1999). The FSF advocates the use of its GNU General Public License (GPL) as a copyright license which creates, promotes, and protects software freedom. The FSF is at the forefront of the FSM, serving as the movement's key activist. The FSF takes the position that "non-free software is a social problem and free software is the solution[5]."

In contrast, much of the world of open source software believes that the use of non-free software (i.e., mainly, proprietary software) overlaps the world of free software, but "open source software" is presented, identified, and licensed as something that is more friendly to business interests, compared to free software. In 1999, at a meeting during an open source conference, several developers from the free and open source software development teams suggested that the terms related to free software connoted the exception of widespread acceptance by businesses. After that meeting, they agreed to use the new term "open source" and began the Open Source Initiative (OSI)[6]. One way to characterize the OSI is as a sub-movement of the FSM or alternatively, as a counter-FSM because it allows for the inclusion of proprietary software in conjunction with free software. While definitions and alternative licenses for open source software are available, it should be noted that numerous surveys of open source software projects reveal that the majority use the GPL. Thus, projects that identify themselves as "free

software" development projects are more likely to be closely aligned to FSF, the free software movement's ideology, and even to RMS[7].

In the occupational community of F/OSS developers, there is a new business model being promoted called the *Free Software Business model*[8] in which companies contribute to the development and improvement of Free Software and uphold the principles of the FSM. These businesses invest in Free Software development and make money from it by reselling the software or offering consulting services to companies who use Free Software. In this paper, we show how GNUe works as a virtual community within this *Free Software Business model* and operates within the confines of the GPL and the principles of the FSM.

RESEARCH SITE: THE GNUE ENTERPRISE

We selected the GNU Enterprise (GNUe)[9] virtual community as our research site to study processes of free software development and the work practices of free software developers. GNUe is a meta-project of the GNU[10] Project that exists on the Web, but otherwise has no physical place of operation or business. GNUe is nonetheless a going concern that collects enterprise software for developing electronic business and enterprise resource planning (ERP) applications in one location on the Web (Scacchi, 2002b). GNUe seeks to develop:

1) A set of tools that provide a development framework for enterprise information technology professionals to create or customize applications and share them across organizations;

2) A set of packages written using the set of tools to implement a full ERP system; and

3) A general community of support and resources for developers writing applications using GNUe Tools

GNUe operates as an international virtual organization for software development based in the U.S. and Europe. It is centered about the GNUe Web portal and global Internet infrastructure that enables remote access and collaboration. Developing and accessing the GNUe software occurs through the portal, which serves as a global information sharing workplace and collaborative software development environment. As many as twelve companies located across the U.S. and Europe sponsor paid participants. These companies provide salaried personnel, computing resources, and infrastructure that support this organization. However, many project participants support their participation through other means. In addition, there are also dozens of unpaid volunteers who make occasional contributions to the development, review, deployment, and ongoing support of this organization, and its software products and services. Finally, there are untold numbers of "free riders" who will simply download, browse, use, evaluate, deploy, or modify the GNUe software with little/no effort to contribute back to the GNUe community.

F/OSS DEVELOPERS AS AN OCCUPATIONAL COMMUNITY

One way of viewing groups with shared goals in organizations is to characterize them as occupational communities (Bechky, 2003; Elliott, 2000; Gregory, 1983; Van Maanen and Barley, 1984) or as organizational subcultures (Martin, 2002; Schein, 1992; Trice and Beyer, 1993). Occupational communities share similar goals, work practices, beliefs, interests, and value systems. They are bound by socially constructed rules and ethics that promote formation of shared ideologies and cultural forms. Van Maanen and Barley (1984) suggested the use of occupational communities as an alternative to an organizational frame of reference for understanding why it is that people act as they do in the workplace.

The impetus for the evolution of an occupational community comes from a desire for occupational self-control. Members derive valued identities or self-images from their occupational role. Work is a source of pride and meaning. Occupational self-control is an important facet of occupational communities since it enables members to dictate who will and who will not become a member. It also provides them with empowerment against management directives depending on circumstances. For example, attorneys in the United States are regulated by State Bar Associations that are made up of attorneys themselves.

F/OSS communities are not necessarily collocated (Van Maanen and Barley, 1984). In this paper we characterize the F/OSS development community as an occupational community with occupational subcommunities or subcultures (Schein, 1992; Trice and Beyer, 1993) forming within each F/OSS project. F/OSS occupational subcultures share

beliefs, values, and norms from the overall F/OSS occupational community, while developing some that are unique to their particular free software project. For example, although the GNUe community adheres to the beliefs in free software and freedom of choice promoted by the FSF, they also elect to use daily IRC, IRC archives, and summary digests as their key communication medium, something not as widespread in all free software projects.

Using the nexus approach (Martin, 2002), we view the occupational community of F/OSS developers as crossing organizational boundaries in various virtual organizations. In the nexus approach to the study of culture, researchers acknowledge that an organization is unlikely to be isolated and unaffected by the society at large. This is especially true in virtual organizations that have fluctuating boundaries: "Non-unique manifestations reflect influences external to the focal organization. What is unique and organizational, then, will be the particular content and mix of these influences as they come together within the permeable, fluctuating boundary of a collectivity, such as an organization" (Martin, 2002 p. 164).

RESEARCH METHODS IN BRIEF

The sources of data for this qualitative study include books and articles on F/OSS development, instant messaging (Nardi et al., 2000; Herbsleb and Grinter, 1999) transcripts captured through IRC logs, threaded email discussion messages, and other Web-based artifacts associated with GNUe such as Kernel Cousins (summary digests of the IRC and mailing lists)[11]. This research also includes data from email and face-to-face interviews with GNUe contributors, and observations at Open Source conferences. The

first author spent over 200 hours studying and perusing IRC archives and mailing list samples during open, axial coding and analysis phases of the grounded theory. During the open coding phase, the first case study presented in (Elliott and Scacchi, 2002a) was selected as representative of the strong influence of cultural beliefs on GNUe software development practices. We interpreted books and documents as well as Web site descriptions of F/OSS processes (Scacchi, 2002a; b). We discovered strong cultural overtones in the readings and began searching for a site to apply an analysis of how motivations and cultural beliefs influenced the social process of F/OSS. The GNUe Web site offered public access to downloadable IRC archives and mailing lists as well as lengthy documentation - all facilitating a virtual ethnography (Hine, 2000). We took each IRC and kernel cousin related to the three cases and applied codes derived from the data (Strauss and Corbin, 1990). In this way, we discovered relationships between the codes derived in the open coding phase. During the axial coding phase of several IRC chat logs, mailing lists and other documentation, we discovered relationships between beliefs and values of the work culture and manifestations of the culture. For a detailed presentation of the variables and relationships, see (Elliott and Scacchi, 2003a). We discovered that for some GNUe participants, the strong belief in the development and use of free software was an idealistic motivation for joining and perpetuating the community. In addition, we explored the influence of the FSM on the ideology of GNUe and its work practices. We then began characterizing F/OSS developers as an occupational community with a quest for self-control of affairs and with varying strengths in beliefs ad values. Figure 1 shows how we can view the F/OSS developers as split into two occupational subcultures – those who believe in the free software philosophy and those who believe in the open source software philosophy. In this research, we are focusing on

the free software occupational subculture. The Figure also shows some overlap between the two subcultures indicating that some F/OSS developers do not believe in the "free" or "open source" philosophy exclusively.

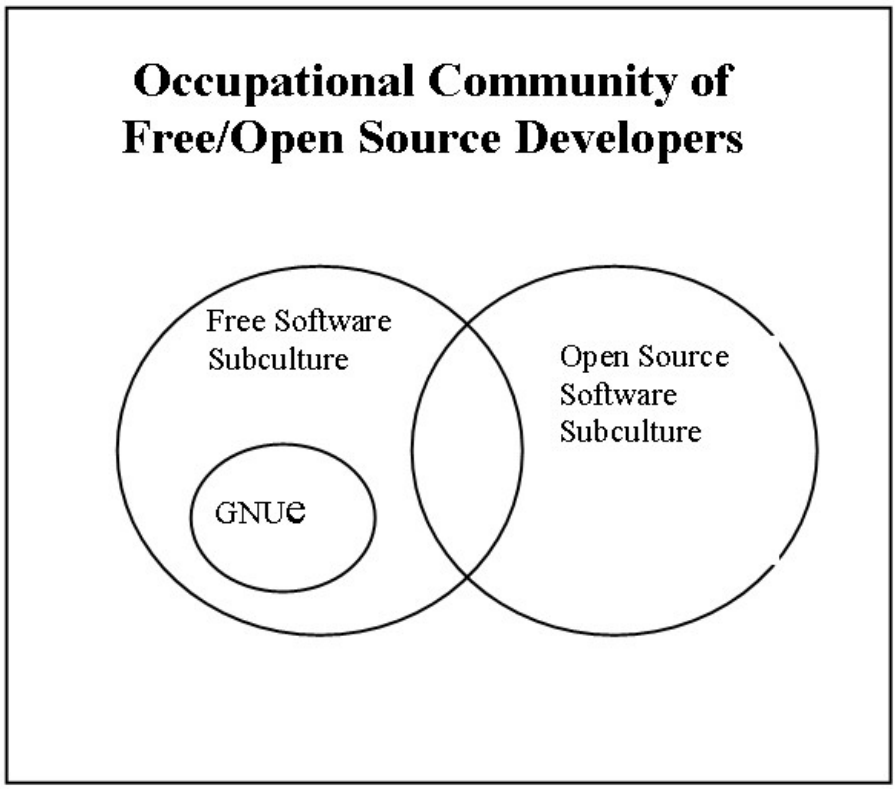

Figure 1. Occupational Community of F/OSS Developers

In the next section we present empirical results from the GNUe case study using three norms that reflect the manifestation of the FSM ideology. They include: informal self-management of task assignment and completion; immediate acceptance of new contributors; and open disclosure of IRC discussions, email, discussion lists, kernel cousins, and documentation.

BELIEFS

Beliefs form the core of ideologies, and as such, are an important component of any cultural study, as well as a driving force that empowers a social movement like the FSM. In this study, they include the beliefs in free software and freedom of choice. We describe these beliefs in terms of how they are reflected in the GNUe community.

Belief in Free Software

The belief in free software is a core motivator of free software developers. GNUe developers show a strong belief in free software extolling its virtues on its Web site and in daily activity on the IRC logs. This belief is manifested in electronic artifacts such as the Web pages, source code, software design diagrams, and accompanying articles. The GNUe Web site advertises that it is "a free software project with a corps of volunteer developers around the world working on GNUe projects". The GNUe Web site home page [12] advertises itself as a world-wide enterprise. The Web site provides a link to an article by RMS, as well as declaring that its purpose is “putting the free back into free enterprise.” The GNUe software is licensed under the GNU General Public License (GPL)[13]. The preamble to this license states the philosophy behind the free software approach:

> "The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to *guarantee your freedom to share and change free software*--to make sure the software is free for all its users ... When we speak of free software, we are

> referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that *you know you can do these things*." (Emphasis added).

The belief in free software is manifested formally, through the rights and imperatives afforded in the GPL that one realizes if employing free software, and informally in the moral imperatives (emphasized in the quoted excerpts) that contextualize the software development work practices of F/OSS contributors. Throughout the GNUe summary digests and IRC logs, there are numerous references to the importance of adhering to the principles of free software.

Belief in Freedom of Choice

Free software developers are attracted to the occupation for its freedom of choice in assignments. Both paid and unpaid GNUe participants to some degree can select the work they prefer. This belief is manifested in the informal methods used to assign and select work in an open source project. Pavilcek (2000) describes this motivation as:

> "Another cherished priority in geek culture is the ability of the geek to pursue her passions and ideas. Their bosses assign most people working in the software industry to projects. In geek culture as well, people are often willing to take on tasks that need

> to be done, even if it is a task they do not relish the thought of pursuing. But geek culture recognizes that there are also tasks that need to be done not because a project requires it, but because the task is burning in the heart and mind of the geek" (Pavlicek, 2000, p. 56).

During an interview with one of the core contributors at a LinuxWorld conference in August 2002, we asked how assignments were made and monitored. He answered with:

> "The number one rule in free software is 'never do timelines or roadmaps'. This is a problem in open source projects. We could use a better roadmap, not having one hinders us. The features we add come about by need during consulting implementations. We may need some kind of roadmap in the future as we expand with more people." (Derek, face-to-face interview, August 2002).

VALUES

The core values identified for the GNUe culture are building community and cooperative work. These values were evident in our study of F/OSS websites, documentation, and informative articles and books (DiBono et al., 1999; Pavlicek, 2000; Raymond, 2001). Here we present how these values are manifested in the GNUe community.

Building Community

The GNUe online community exists for the purpose of developing a free ERP system. The beliefs in free software and freedom of choice foster a value in community building

as a prerequisite for completing work. In some cases, this value is espoused as stated below:

> "Many free software folks think IRC is a waste of time as there is 'goofing off', but honestly I can say its what builds a community. I think a community is necessary to survive. For example GNUe has been around for more than 3 years. I can not tell you how many projects have come and gone that were supposed be competition or such. I put our longevity solely to the fact that we have a community." (Derek, email interview (2002))

In other cases, it is inferred from the research and is evident in the IRC archives when newcomers join GNUe offering contributions and existing contributors quickly accept them as part of the community.

Cooperative Work

The GNUe community's belief in freedom and value in community fosters a value in cooperative work. The analysis of IRC archives gives evidence of this value when conflict occurs and many contributors try to resolve the problem. (See (Elliott and Scacchi, 2003a; b) for a detailed account).

GNUE NORMS

The three norms presented below were prevalent throughout the GNUe data. While other norms prevailed as well, we selected these as the main themes related to the

manifestation of the beliefs and values of the FSM and as emblematic of the occupational community of F/OSS developers.

Informal self-management

One of the key issues of importance to the F/OSS occupational community is its ability to manage software development without a top manager monitoring activity and passing judgement on the quality and timeliness of the work (Scacchi 2004). The entire GNUe virtual organization is informal. There is no lead organization or prime contractor that has brought together the alliance of individuals and sponsoring firms as a network virtual organization. It is more of an emergent organizational form where participants have in a sense discovered each other, and have brought together their individual competencies and contributions in a way whereby they can be integrated or made to interoperate (Crowston and Scozzi, 2002). In GNUe no company has administrative authority or resource control to determine: (a) what work will be done; (b) what the schedule will be; (c) who will be assigned to perform specified tasks; (d) whether available resources for the project are adequate, viable, or extraneous; nor (e) who will be fired or reassigned for inadequate job performance (Scacchi 2002b). The project appears to be monitored by a small group of core developers (known as core maintainers in their culture) who are responsible for a large bulk of the code, test, and release of software. There is also a list of frequent contributors who work on software development, documentation, and other administrative duties. GNUe also welcomes a crew of casual volunteers (those who contribute once or twice, or who work sporadically). The participants come from different small companies or act as individuals that collectively act to move the GNUe

software and the GNUe community forward. Thus, the participants self-organize in a manner more like a meritocracy (Fielding, 1999; Scacchi, 2004).

A core maintainer explains the typical method of managing the GNUe software assignments as:

> "The number one rule in free software is 'never do timelines or roadmaps'. This is a problem in open source projects. We could use a better roadmap, not having one hinders us. The features we add come about by need during consulting implementations. We may need some kind of roadmap in the future as we expand with more people."
>
> (Derek, face-to-face interview, August 2002)

In two of the GNUe cases studies, we found evidence of debates over the use of non-free versus free tools. In both cases, the group resolved the conflict without a top manager needing to intervene. Both examples illustrate how the strong belief in free software exhibited by two contributors can result in a heated discussion on the IRC and mailing list archives. Each case will be discussed briefly here. Details may be found in (Elliott and Scacchi, 2003a).

The first case study reveals a debate over the use of a non-free tool to create a graphic diagram that is posted on the GNUe Web site. This exchange takes place one day on the IRC channel and ends the next morning.[14] This example illustrates the ease with which a

newcomer comes onboard this F/OSS project and then criticizes the methods used to produce a graphical representation of the system architecture on the GNUe Web site.

The strong belief in free software of the outsider leads to conflict among those insiders who have a moderate view of the use of free software for GNUe software development. A daylong debate ensues among Neilt, creator of the graphic; CyrilB, the outsider; and other GNUe contributors regarding the use of a non-free software tool to create a graphic for a GNUe screenshot for Website documentation. This first excerpt shows how **CyrilB** gets on the IRC and expresses his concern for the "shocking" use of a non-free tool on a free software project[1]:

> <CyrilB> Hello
>
> <CyrilB> Several images on the GNUe website seems to be made with non-free Adobe softwares, I hope I'm wrong: it is quite shocking. Does anybody know more on the subject ?
>
> <CyrilB> lynx -source http://www.GNUe/modules/NS-My_eGallery/gallery/GNUe/GNUePkgArchitecture.png | strings | head
>
> <CyrilB> We should avoid using non-free software at all cost, am I wrong ?
>
> <CyrilB> Anyone awake in here ?

This is an example of how the global belief system of the free software occupational community has influence on how a free software project is maintained. Reinhard, a core

[1] The IRC excerpts are presented verbatim with extraneous text eliminated for clarity. They are in a different font than regular text. The codes are shown in parentheses in italics bold type.

maintainer, arrives and points out to CyrilB that the main goal of the project is to produce good free software and *how* it is produced is not a main concern. In this next passage, Reinhard explains his moderate view of the belief in free software and surprisingly, he accepts the criticism of CyrilB engaging him in conversation to explain the reason for allowing such work on a GNU free software project:

> <reinhard> CyrilB: our main goal is to produce good free software
> <reinhard> we accept contributions without regarding what tools were used to do the work
> <reinhard> especially we accept documentation in nearly any form we can get because we are desperate for documentation just like any other gnu project. just as long as the format itself isn't proprietary, and it can be viewed without proprietary programs, anything is ok for us.
> <reinhard> at least that is my understanding

The discussion continues with a technical discussion of what it would take to redo the graphic in free software. CyrilB emphasizes the need for free software again. Reinhard agrees in principle but wants an interim solution on a practical level.

> <CyrilB> We need to be able to modify the code and we can't modify Adobe files with free software...
> We need people do be able to use free softwares.

Later he admonishes neilt, the original creator of the graphic:

> <CyrilB> neilt: you are compromising our freedom by using non-free software: we can't modify and/or redistribute the source vector file.

After several interchanges with CyrilB and after many suggestions from other contributors and lurkers, Neilt agrees to change the graphic. However, he has a heated discussion with CyrilB saying that if he is developing free software, he should also have a freedom of choice. Even though he does not agree with CyrilB, they part as colleagues and worked out the difference of opinion without a manager.

In case two, we explored project insider procedures and practices for developing GNUe documentation.[15] Once again the debate revolves around polarized views of the use of non-free tools to develop GNUe documentation. In this case, Chillywilly, a frequent contributor, balks at the need to implement a non-free tool on his computer in order to edit the documentation associated with a current release. Even though his colleagues attempt to dissuade him from his concerns by suggesting that he can use any editor--free or non-free--to read the documentation in HTML or other formats, Chillywilly refuses to back down from his stance based on a strong belief in free software. This debate lasts three days.

The strength in the belief in free software drives the three-day long discussion. The debate and its resolution also illustrate the global effort by GNUe developers to collaborate and work cooperatively through the use of the IRC channel. Chillywilly begins his IRC with an observation that a fellow collaborator, jamest, has made documents with lyx:

Action: chillywilly trout whips jamest for making lyx docs

Action: jcater troutslaps chillywilly for troutslapping jamest for making easy to do docs

<chillywilly> lyx requires non-free software

<Maniac> lyx rules

<chillywilly> should that be acceptable for a GNU project?

<jcater> chillywilly: basically, given the time frame we are in, it's either LyX documentation with this release, or no documentation for a while (until we can get some other stinking system in place)

<jcater> pick one :)

<chillywilly> use docbook then

…

The following interchange illustrates the importance to the occupational culture of RMS and the belief system of the FSF. Chillywilly is so adamantly opposed to the use of non-free software that he references RMS as part of his reasoning – "I will NOT install lyx and make rms unhappy". This passage shows how RMS is considered the "guru" of the free software movement. Eventually chillywilly sends an email to the mailing list:

"OK, I saw on the commit list that you guys made some LyX documents. I think it is extremely ***that a GNU project would require me to install non-free software in order to read and modify the documentation. I mean if I cannot make rms happy on my debian system them what good am I as a Free Software developer?… I really shouldn't have to be harping on this issue for a GNU project, but some ppl like to take convenience over freedom and this should not be tolerated… Is it really that unreasonable to request that we

not use something that requires ppl to install non-free software? Please let me know." (Chillywilly, mailing list).

Jcater returns later with the following email rejoinder to chillywilly's "flaming" email:

"I would like to personally apologize to the discussion list for the childish email you recently received. It stemmed from a conversation in IRC that quickly got out of hand. It was never our intention to alienate users by using a non-standard documentation format such as LyX. … LyX was chosen because it is usable and, more importantly, installable. After many failed attempts at installing the requirements for docbook, James and I made the decision that LyX-based documentation with the upcoming 0.1.0 releases was better than no documentation at all…

PPS, By the way, Daniel, using/writing Free software is NOT about making RMS happy or unhappy. He's a great guy and all, but not the center of the free universe, nor the motivating factor in many (most?) of our lives. For me, my motivation to be here is a free future for my son." (Jcater, mailing list).

Jcater responded to chillywilly's email with the idea that the overall goal of GNUe is to cooperatively create documentation as easily and quickly as possible. His parting remark of being motivated for the free future of his son is an exemplar of the global belief system that drives the FSF and occupational community membership. Later Chillywilly is mollified by colleagues who persuade him with arguments like appealing to his freedom of choice.

Immediate acceptance of new contributors

In case one, we showed how a new contributor's criticism of the non-free graphic was accepted by core maintainers and how the original creator of the graphic eventually agreed to change it to one created using a free graphics tool. In case three (Elliott and Scacchi, 2003a), a new member joins the IRC discussion inquiring about what work needs to be done since he is available for a month (i.e. he is a consultant who is unemployed for a month). In one day, he downloads the latest GNUe version, discovers a bug, and offers a valid bug-fix. His performance is lauded by core maintainers and he is encouraged to continue his contributions.

Open disclosure

Open disclosure refers to the open content of the GNUe Website including the software source code, documentation, and archived records of IRC, kernel cousins, and mailing list interchanges. The GNUe contributors join others online via IRC on a daily basis and record the conversations for future reference. All documentation and source code are easily downloaded from the GNUe website and user criticism is welcomed by frequent GNUe maintainers. In the "geek" culture, truth is a core priority in developing open source software: "It should not be too surprising, then, that one of the key values for the community is truth. In a world where people are constantly exchanging ideas, evaluating concepts, and suggesting enhancements, it is vitally important that everyone speak the truth as he sees it. If someone fails to speak the truth, the process of creating software will be greatly impaired (Pavlicek, 2000, p. 53)."

In the GNUe culture and in the open source culture in general, the importance of speaking the truth in daily work practices is a key element of their culture. In the GNUe project, truth is apparent in the norm of open disclosure of software development processes. This is accomplished by the recording and public archiving of CMC via various mediums all recorded for archival purposes: IRC logs, email discussions, and digests (i.e. kernel cousins).

Each digest summarizes IRC logs and/or email messages for a period of from one to two weeks, includes direct quotes from participants, and includes hyperlinks to the original message sources. A digest sometimes reads more like a dramatized account with editorial remarks than like a simple summary of facts. These summaries serve as a resource and organizational memory of activities within the GNUe virtual organization (Ackerman and Halverson, 2000).

CONCLUSIONS

We have presented a model of the F/OSS occupational community as having beliefs in free software and freedom of choice and values in cooperative work and community. Our analysis of the GNUe virtual organizational culture revealed a manifestation of these beliefs and values in the norms of informal self-management, immediate acceptance of contributors, and open disclosure. F/OSS developers have unique and esoteric skills and ways of doing software development that differ from a proprietary in-house arrangement. The boundaries of the free/open source community, in general, and GNUe, in particular, fluctuate constantly as new software developers contribute new code, suggest design changes, and fix bugs. Often these developers view themselves as different from the rest

of society and identify themselves as geeks (Pavlicek, 2000). This global occupational community of F/OSS developers shares beliefs and values that help mobilize contributions to free and open source projects, and serve to define the boundaries of their virtual space. They share a sense of accountability to produce free software with the use of free software tools as much as possible. This assumed system of accountability ties the community together while allowing it to move forward in efforts to produce free software.

Overall, the goal of this research is to develop a theory for how F/OSS communities develop software and how the cultural beliefs of the occupational community influence their work practices. Further study is needed to compare the beliefs, values, and norms of GNUe to other F/OSS projects to explore the extent of generalization to the work patterns of F/OSS developers in the F/OSS occupational community. However, we have shown how the FSM has influenced the evolution of the occupational community of F/OSS developers and how the beliefs and values of the FSM ideology are manifested in the norms of GNUe, a typical occupational subculture of the community. The FSM differs from previous CMs characterized by Kling and Iacono (1988; 2001) where mass computerization was the goal of specific CMs. The FSM has a goal of altering the type of software people create and use (from proprietary to free) with an assumption that massive computerization will or has already taken place.

The ideology of the FSM enables and sustains the development of an occupational community to carry on the beliefs and values in occupational subcultures of virtual work communities. The beliefs in freedom, free software, and freedom of choice create a

special bond for the people working on free software projects. These beliefs foster the values of cooperative work and community building. Schein's (1992) theory of organizational culture includes the revelation of underlying assumptions of cultural members that are on a mostly unconscious level. In the GNUe world, the underlying assumptions of cooperative work and community building become routine in the everyday work practices of GNUe contributors in their pursuit of an ERP system implemented as free software. The strong identity of the GNUe contributors to the principles of RMS and the FSF serves to mobilize participation, perpetuate the GNUe community and its work, and memorialize the FSM in general.

ACKNOWLEDGMENTS

The research described in this report is supported by grants from the National Science Foundation #ITR-0083075, #ITR-0205679, #ITR-0350754, and #ITR-0205724. No endorsement implied. Mark Ackerman at University of Michigan, Les Gasser at University of Illinois, and Chris Jensen at the UCI Institute for Software Research are collaborators on this research.

REFERENCES

Ackerman, M. and Halverson, C. (2000), "Reexamining Organizational Memory", *Communications of the ACM,* Vol. 43, No. 1, pp. 59-64.

Bechky, B. (2003), "Sharing Meaning Across Organizational Communities: The Transformations of Understandings on a Production Floor", *Organization Science,* Vol. 14, No. 3, pp. 312-330.

Blumer, H. (1969), "Social Movements", in McLaughlin, B. (Ed.) *Studies in Social Movements: A Social Psychological Perspective*, Free Press, New York.

Crowston, K. and Scozzi, B. (2002), "Exploring Strengths and Limits on Open Source Software Engineering Processes; A Research Agenda", in *2nd Workshop on Open Source Software Engineering*Orlando, Florida.

DiBona, C., Ockman, S. and Stone, M. (1999), *Open Sources: Voices from the Open Source Revolution,* O'Reilly & Associates Inc., Sebastol, CA.

Elliott, M. (2000), *Organizational Culture and Computer-Supported Cooperative Work in a Common Information Space: Case Processing in the Criminal Courts,* University of California, Irvine, Irvine.

Elliott, M. (2003), "The Virtual Organizational Culture of a Free Software Development Community", in *3rd Workshop on Open Source Software*Portland, Oregon.

Elliott, M. and Scacchi, W. (2003a), "Free Software: A Case Study of Software Development in A Virtual Organizational Culture", Institute for Software Research, University of California, Irvine, Irvine, CA.

Elliott, M. and Scacchi, W. (2003b), "**Free Software Developers as an Occupational Community: Resolving Conflicts and Fostering Collaboration**", in *GROUP '03*ACM, Sanibel Island, Florida.

Elliott, M. and Scacchi, W. (2004), "Free Software Development: Cooperation and Conflict in a Virtual Organizational Culture", in Koch, S. (Ed.) *Free/Open Source Software Development*, Idea Press.

Feller, J. a. F., B. (2002), *Understanding Open Source Software Development,* Addison-Wesley, New York, NY.

Gregory, K. (1983), "Native-view Paradigms: Multiple Cultures and Culture Conflicts in Organizations", *Administrative Science Quarterly,* Vol. 28, pp. 359-376.

Herbsleb, J. D. and Grinter, R. (1999), "Splitting the Organization and Integrating the COde: Conway's Law Revisited", in *21st International Conference on Software Engineering*ACM Press, Los Angeles, CA.

Hine, C. (2000), *Virtual Ethnography,* Sage Publications, London.

Iacono, S. and Kling, R. (1996), "Computerization Movements and Tales of Technological Utopianism", in Kling, R. (Ed.) *Computerization and Controversy: Value Conflicts and Social Change, 2nd Ed.*, Academic Press, San Diego, CA.

Iacono, S. and Kling, R. (2001), "Computerization Movements: The Rise of the Internet and Distant Forms of Work", in Yates, J. A. and Maanen, J. V. (Eds), *Information Technology and Organizational Transformation: History, Rhetoric and Practice*, Sage Publications, .

Kling, R. and Iacono, S. (1988), "The Mobilization of Support for Computerization: The Role of Computerization Movements", *Social Problems,* Vol. 35, No. 3, pp. 226-242.

Kling, R. and Scacchi, W. (1982), "The Web of Computing: Computer Technology as Social Organization", in Yovits, M. (Ed.) *Advances in Computers*, 21, 3-85, Academic Press.

Kollock, P. and Smith, M. (1996), "Managing the Virtual Commons: Cooperation and Conflict in Computer Communities", in Herring, S. (Ed.) *Computer-Mediated Communication: Linguistic, Social, and Cross-Cultural Perspectives*, John Benjamins, Amsterdam.

Kollock, P. and Smith, M. (1999), "Communities in Cyberspace", in Smith, M. and Kollock, P. (Eds), *Communities in Cyberspace*, Routledge, London.

Martin, J. (2002), *Organizational Culture: Mapping the Terrain,* Sage Publications, Thousand Oaks.

Mockus, A., Fielding, R. T. and Herbsleb, J. (2002), "A Case Study of Open Source Software Development: The Apache Server", *ACM Transactions on Software Engineering and Methodology,* Vol. TBD, .

Nardi, B., Whittaker, S. and Bradner, E. (2000), "Interaction and Outeraction: Instant Messaging in Action", in *Conference on Computer Supported Cooperative Work*ACM Press, Philadelphia, PA.

Pavlicek, R. G. (2000), *Embracing Insanity: Open Source Software Development,* SAMS Publishing, Indianapolis, IN.

Scacchi, W. (2002a), "Open EC/B: A Case Study in Electronic Commerce and Open Source Software Development", Institute for Software Research, UC Irvine, Irvine, CA.

Scacchi, W. (2002b), "Understanding Requirements for Developing Open Source Software Systems", *IEE Proceedings--Software,* Vol. 149, No. 2, pp. 24-39.

Scacchi, W. (2004), "Free and Open Source Development Practices in the Game Community", *IEEE Software,* Vol. 21, No. 1, pp. 59-67.

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

Stallman, R. M. (2002), *Free Software, Free Society: Selected Essays of Richard M. Stallman,* GNU Press.

Strauss, A. and Corbin, J. (1990), *Basics of Qualitative Research: Grounded Theory Procedures and Techniques,* Sage Publications, Newbury Park, California.

Trice, H. M. and Beyer, J. M. (1993), *The Cultures of Work Organizations,* Prentice Hall, Englewood Cliffs, NJ.

Van Maanen, J. V. and Barley, S. R. (1984), "Occupational Communities: Culture and Control in Organizations", *Research in Organizational Behavior,* Vol. 6, pp. (287-365).

Williams, S. (2002), *Free as in Freedom: Richard Stallman's Crusade for Free Software,* O'Reilly & Associates, Sebastopol, CA.

[1] See http://www.fsf.org/philosophy/free-sw.html for a detailed definition of free software.
[2] http://www.gnuenterprise.org
[3] See http://www.fsf.org/philosophy/free-sw.html for a detailed description of the FSM philosophy.
[4] For more information on the FSF, see http://www.fsf.org.
[5] http://www.fsf.org/philosophy/free-software-for-freedom.html
[6] http://www.opensource.org
[7] For example, in the DotGNU project (http://www.fsf.org/projects/dotgnu/), throughout the website, references are made to the FSF and the philosophical foundations of the FSM.
[8] http://www.gnu.org/projects/dotgnu/win.html
[9] http://www.gnuenterprise.org
[10] http://www.gnu.org
[11] See http://kt.zork.net for the complete set of GNUe kernel cousins.
[12] http://www.gnue.org
[13] http://www.gnu.org/copyleft/gpl.html
[14] See http://www.gnuenterprise.org/irc-logs/gnue-public.log.25Nov2001 for the full day's log.
[15] See http://www.gnuenterprise.org/irc-logs/gnue-public.log. 15Nov2001 for the full three day logs.