Free Software Developers as an Occupational Community: Resolving Conflicts and Fostering Collaboration

Margaret S. Elliott
Institute for Software Research
University of California, Irvine
Irvine, CA 92697
949 824-7202
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
In this paper, we present results from the study of a free software development virtual organization, the GNU Enterprise (GNUe) project, and how they develop software in a globally distributed free software development project. In particular, examples of how they mitigate and resolve conflict are presented. Conflict arises over the use of a non-free tool to create GNUe graphic, and over the use of a non-free tool for GNUe documentation. The GNUe developers resolve the conflict using internet relay chat (IRC), threaded email discussions, and community digests. We characterize the GNUe developers as an occupational subculture within the occupational community of free/open source software (F/OSS) developers and show how the beliefs in free software and freedom of choice, and values in cooperative work and community assist GNUe contributors in mitigating and resolving conflict. In addition, we show how, despite fluctuating boundaries of membership in a virtual organization, daily discussions on the GNUe IRC serve to build and perpetuate the global community of GNUe contributors as well as F/OSS developers in general.

Categories and Subject Descriptors

K.4.3 [Computers and Society]: Organizational Impact --- Computer-Supported Collaborative Work; K.7.1 [The Computing Profession]: Occupations.

General Terms
Human Factors, Design

Keywords
Computer-supported cooperative work, conflict, occupational community, organizational culture, free/open source software development, virtual organization

1. INTRODUCTION

Free/open source software (F/OSS) development projects are growing at a rapid rate. The SourceForge Web site estimates 600,000+ users with 700 new ones joining every day and a total of 60,000+ projects with 60 new ones added each day. Thousands of F/OSS development projects have emerged within the past few years [3, 28] leading to the formation of globally dispersed virtual communities [19]. Studies are needed to understand how people in these projects work together to coordinate software development at a distance, and what social worlds arise to assist in this collaboration. Research has focused on the quantitative side of F/OSS development projects, such as aspects of developer defect density, core team size, and other variables [17, 23, 24]. Few researchers have gone beyond the quantitative approach to focus on open software projects as social phenomena [2, 8, 9, 10, 21, 31, 32, 33]. More research is needed to understand the social nature of F/OSS development and what social processes enable this rapidly growing phenomenon.

This paper presents the results of a qualitative study of a free software development community. We show how they mitigate and resolve conflict, and build a community via instant message (IM) streams using internet relay chat (IRC) logs, text-based records of IRC logs, mailing lists, and summary digests (called "kernel cousins" by participants in our study). We characterize the F/OSS developers as an occupational community [39, 40] with beliefs, values, and norms that influence their software development practices including tool choices and conflict resolution. In order to study the culture of an occupational community, we used an organizational culture perspective [22, 34, 39].

Much like typical organizations that exist in buildings, virtual organizations have beliefs and values manifested in norms that form behavioral expectations and that give members guidelines for "the way to do things around here." Using an organizational culture perspective enables researchers to study occupations with globally disbursed members as organizational subcultures [39] each with their own unique sets of beliefs, values, and norms in addition to those shared by the occupational community. Within typical and virtual organizations, occupational subcultures exist and adhere to their own code of ethics [39]. For purposes of this paper, we present F/OSS developers as an occupational community with occupational subcultures forming in various F/OSS projects. These occupational subcultures promote the formation of shared ideologies and cultural forms. Over time, occupational members share a similar view of their work and of

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GROUP'03 Conference, November 9-12, 2003, Sanibel Island, FL.
COPYRIGHT 2003 ACM 1-58113-693-5/03/0011…$5.00.

the environment in which they perform that work [39]. In this paper, we characterize the occupational community of F/OSS developers as consisting of two occupational subcultures – the free software developers as to understand the social nature of F/OSS development and what social processes enable this rapidly growing phenomenon.

This study is part of an ongoing comparative study of various types of open software communities [8, 9, 10 31, 32, 33] including both free and open source software development projects. It is important to distinguish between the philosophical implications of the terms free software [37] and open source software [3]. In the free software community, free software refers to software that is open to anyone to copy, study, modify, and redistribute [37]. The Free Software Foundation (FSF), founded by Richard M. Stallman (known as RMS in F/OSS communities) [37] in the 1980s, advocates the use of its GNU General Public License (GPL) as a copyright license which creates, promotes, and protects software freedom. The GPL in a sense serves as the constitution for the FSF community.

The FSF is at the forefront of the free software movement, and advocates that free and open source code is fundamental to the furthering of computer science and that free source code is necessary for innovation to flourish in computer science [3]. The FSF takes the position that "non-free software is a social problem and free software is the solution (http://www.fsf.org/philosophy/free-software-for-freedom.html)."

In contrast, much of the world of open source software overlaps the world of free software, but "open source software" is presented, identified, and licensed as something that is more akin to business interests, compared to free software. While definitions and alternative licenses for open source software are available (see www.opensource.org), it should be noted that numerous surveys of open source software projects reveals that the majority use the GPL from the free software world, but otherwise identify themselves as open source software projects. Thus, projects that identify themselves as "free software" development projects are more likely to be closely aligned to FSF, the free software movement, and even to RMS. 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 Free Software Movement.

Conflict is an integral part of cooperative work in many work settings [5, 6]. It is inevitable in software development, especially in virtual organizations where assignments are loosely made, projects are managed informally, and where users are communicating from across the world in mainly text-based venues. As conflicts arise between people engaging in collaborative work activities, CSCW should include an understanding of how collaboration may break down and how it can continue in the presence of conflict [5, 6]. Understanding how conflict is mitigated and resolved in F/OSS development communities is beneficial to researchers interested in developing F/OSS CSCW systems, and for managers considering the introduction of F/OSS development into their organization.

In the occupational community of F/OSS developers, there is a new business model being promoted called the *Free Software Business model* (http://www.gnu.org/projects/dotgnu/win.html) in which companies contribute to the development and improvement of Free Software and uphold the principles of the Free Software Movement. 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. Any business contemplating such an investment would benefit from an understanding of how F/OSS developers mitigate and resolve conflicts and how they build a community that can produce a workable business software package. In this paper, we present a case study of the GNU Enterprise (GNUe) (http://www.gnuenterprise.org) free software project whose purpose is to build business applications for a full Enterprise Resource Planning (ERP) system. We show how they mitigate and resolve conflict using instant messaging, summary digests, and mailing lists. In [9], we give more details regarding this case study.

Section 2 presents background material, Section 3 discusses research methods, and Section 4 describes the research site. Section 5 and Section 6 present case one and two respectively, followed by a discussion in Section 7. Finally, Section 8 ends with the conclusion.

2. BACKGROUND

In this section, we further discuss research regarding occupational communities, conflict in virtual organizations, and related research in CMC.

2.1 OCCUPATIONAL COMMUNITIES

One way of viewing groups with shared goals in organizations is to characterize them as occupational communities [40] or as

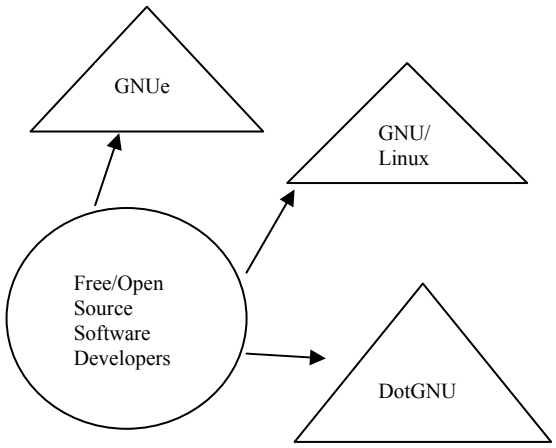

Figure 1. Occupational community conceptualization and examples

organizational subcultures [22, 34, 39]. 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 [40] 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.

Occupational communities are not necessarily collocated [40]. In this paper we characterize the free/open source development community as an occupational community with occupational subcommunities or subcultures [34, 39] forming within each free/open source project. Occupational subcultures then share beliefs, values, and norms from the free/open source occupational community and develop some that are unique to a particular subculture.

Using the nexus approach [22], we view the occupational community of F/OSS developers as crossing organizational borders in various virtual organizations. Figure 1 shows the nexus concept of F/OSS developers joining various F/OSS projects. 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: "Nonunique 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 [22, p. 164]."

Researchers have previously characterized professionals as members of an occupational community [7, 12, 34]. Elliott (7) characterized judges, district attorneys, and public defenders in the criminal courts as occupational communities, and showed how the use of IT changed their organizational culture and, simultaneously, how the organizational culture of the courts altered the usage of the technology. Gregory [12] characterized software developers working in Silicon Valley as sharing the same occupational subculture, no matter which organization they worked in, or what kind of work they did. Schein [34] described IT developers as an occupational community with a common base of knowledge, a common jargon, similar background and training, and a sense of identifying with each other. Occupational communities typically grow up around new technologies and the characteristics of the IT itself, computer hardware and software design require certain kinds of people with certain kinds of skills and cognitive styles [26]. He suggested that as IT becomes differentiated, more variations in the styles of IT community members would occur with subcommunities springing up with their own subcultures. More recently, Bechky [1] presented results of a study of the misunderstandings of the occupational communities' situated understandings of their work including engineers, technicians, and assemblers on a production floor. In her study, members of each occupational community developed shared understandings by developing a common ground on the production floor.

2.2 CONFLICT IN VIRTUAL ORGANIZATIONS

Researchers have studied conflict in closed source software development environments [30], but not in F/OSS development. Only a few researchers have attempted to understand conflict management in virtual communities [18, 35], and none have delved into F/OSS projects. For example, Smith [35] studied conflict management in MicroMUSE, a game world dedicated to the simulation and learning about a space station orbiting the earth. There were two basic classes of participants: users and administrators. Disputes arose in each group and between the two groups regarding issues like harassment, sexual harassment, assault, spying, theft, and spamming. These problems occurred due to the different meanings attributed to MicroMUSE by its players and administrators and due to the diverse values, goals, interests, and norms of the group. Smith concluded that virtual organizations have the same kinds of problems and opportunities brought by diversity as real organizations do, and that conflict is more likely, and more difficult to manage than in real communities. Factors contributing to this difficulty are: wide cultural diversity; disparate interests, needs and expectations; nature of electronic participation (anonymity, multiple avenues of entry, poor reliability of connections and so forth); text-based communications; and power asymmetry among users.

Kollock and Smith [18] studied conflict in Usenet groups emphasizing the importance of recognizing the free-rider problem. In a group situation where one person can benefit from the product or resource offered by others, each person need not contribute to the joint effort, but instead may free-ride on others' work. The authors do a detailed analysis of this free-rider problem and give suggestions for how to avoid it in Usenet groups. For example, it is suggested that bandwidth be used judiciously posting useful information and refraining from posting inappropriate use of bandwidth. Success on a Usenet group also depends on its members following cultural rules of decorum.

2.3 COMPUTER MEDIATED COMMUNICATION

Previous researchers have addressed the issue of the effects of CMC on social influence in groups, the pressure exerted on group members to restrain behavior to fit group norms. Virtual communities have unique circumstances in terms of social control since many times the group members remain anonymous.

Easterbrook, *et al*. [6] discuss several studies focusing on the effects of anonymity on interactions between people in computer-mediated communication (CMC). Other studies indicate that de-individuation occurs when social cues that distinguish individuals are missing. The effect on people is less of a sense of individuality among group participants, detaching individuals from his or her comments that can lead to a reduction in normal restraints on behavior [16].

Another study showed that e-mail reduces social context cues encouraging people to behave irresponsibly more often with a focus on themselves not others [36]. Lea and Spears [20] criticized the Sproull and Kiesler study suggesting an alternative explanation for the de-individuation effect of CMC. They suggest that the social context influences the occurrence of de-individuation. If de-individuation occurs with immersion in a group, then this enhances the salience of the group and strengthens its norms. However, if the group identify is not already established, then de-individuation only serves to strengthen one's sense of individuality, and weakens group norms.

Later, researchers [29] established empirical evidence to refute the de-individuation theory calling it the social identity model of de-individuation effects (SIDE) model. Their work challenged the utopian image of CMC creating a new and more liberated

form of communication. Instead they showed that a group can be seen also as a source of (social) identity and self-expression, and that anonymous CMC can strengthen conformity to group norms. We show in this paper that our research supports this assertion since the GNUe occupational subculture uses the IRC and other forms of CMC to reinforce and perpetuate cultural beliefs and group norms.

3. RESEARCH METHODS

This ongoing ethnography of a virtual organization [15, 27] is being conducted using the grounded theory approach [38] with participant-observer techniques. The sources of data include books and articles on F/OSS development, instant messaging [14, 25] 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 – see http://kt.zork.net). 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 well over 100 hours studying and perusing IRC archives and mailing list samples during open and axial coding phases of the grounded theory. During open coding the first case study presented here was selected as representative of the strong influence of cultural beliefs on GNUe software development practices. The selection of cases was aided by the indexing of each Kernel Cousin into sections labeled with a topic. For example, we read through all Kernel Cousins and found the following title "Using Non-Free Tools for Documentation" in (http://kt.zork.net/GNUe/gnue20011124_4.html). Hyperlinks from this cousin pointed us to a similar case where non-free tools were being used for documentation of code. The third case was found during the coding of the selection for case two (case three is described in [9].

The initial research questions that formed the core of the grounded theory are:

1) How do people working in virtual organizations organize themselves such that work is completed?
2) What social processes facilitate open source software development?
3) What techniques are used in open source software development that differ from typical software development?

We began this research with the characterization of open source software communities as communities of practice. A community of practice (COP) is a group of people who share similar goals, interests, beliefs, and value systems in a common domain of recurring activity or work [41]. An alternative way of viewing groups with shared goals in organizations is to characterize them as organizational cultures [22, 34, 39]. As the grounded theory evolved, we discovered rich cultural beliefs and norms influencing "geek" behavior [28]. This led to us to the characterization of the COPs as virtual organizations having organizational cultures. Furthermore, we characterized the field of F/OSS developers as occupational communities [1, 39, 40], also known as occupational subcultures [39]. An occupation typically consists of a line of work or set of tasks. Occupations are also cultural since they usually come into existence via social processes internal to them. They emerge spontaneously within groups of people who perform like tasks. 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. The global nature of the F/OSS occupational community with people working on many virtual projects simultaneously makes the use of CMC very important for community building and resolving conflicts.

During the open coding, we interpreted books and documents as well as Web site descriptions of the F/OSS process. 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. We selected GNUe as a research site because it exemplified the essence of free software development providing a rich picture of a virtual work community with a rapidly growing piece of downloadable free software. The GNUe Web site offered access to downloadable IRC archives and mailing lists as well as lengthy documentation - all facilitating a virtual ethnography. We took each IRC and kernel cousin related to the three cases and applied codes derived from the data [38]. We used a text editor to add the codes to the IRC text logs using [**Begin** and **End**] blocks around concepts we identified such as "***belief in free software***". 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 [8]. Table 1 lists the basic variables related to the cultural analysis.

Table 1. Research Variables

Beliefs	Values	Norms
Free software - distributing copies of free software **Cooperative work** - resolving conflicts through cooperation	**Freedom of choice** – freedom to select assignments, and tools for development **Building community** – working as a team	**Open disclosure** **Immediate acceptance of outsiders** **Informal management**

These variables are also shown in Figure 3 as the causal conditions of the grounded theory [38]. Figure 3 shows the conceptual diagram that identifies the trajectory of socially constructed events and their relationships to the beliefs and values of the GNUe culture. This diagram will be explained in more depth later in the presentation of the case studies.

As the research evolved, the variables and their relationships were continuously analyzed. We discovered that for some GNUe participants, the strong belief in the use of free software was an idealistic motivation for joining and perpetuating the community. We then began characterizing F/OSS developers as an occupational community with varying strengths in beliefs and values. Figure 2 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.

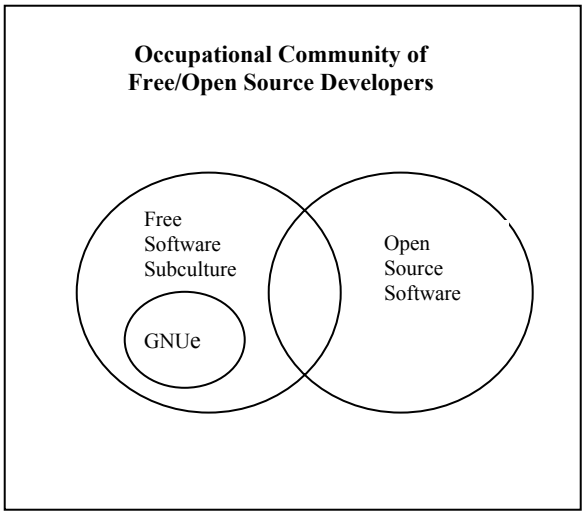

Figure 2. Occupational Community of F/OSS Developers

As the research evolved, the variables and their relationships were continuously analyzed. We discovered that for some GNUe participants, the strong belief in the use of free software was an idealistic motivation for joining and perpetuating the community. We then began characterizing F/OSS developers as an occupational community with varying strengths in beliefs and values. Figure 2 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

4. RESEARCH SITE: GNUE ENTERPRISE

The research site is a free software development community, the GNU Enterprise (GNUe) (http://www.gnuenterprise.org). GNUe is a meta-project of the GNU (http://www.gnu.org) Project. GNUe is designed to collect Enterprise software in one location on the Web. 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 Enterprise Resource Planning (ERP) system; and

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

GNUe is 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 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 [26].

5. CASE ONE: CONFLICT AND DEBATE OVER USE OF NON-FREE GRAPHICS TOOL

This case study reveals a trajectory of a conflict and 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 (See http://www.gnuenterprise.org/irc-logs/gnue-public.log.25Nov2001 for the full day's log). 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. Table 2 displays the total number of contributors and the number of days of the conflict. Eight of the nine regular GNUe contributors were software developers and one was working on documentation. The infrequent contributors drifted on and off throughout the day – sometimes lurking, and other times involved in the discussion.

Table 2. Contributors and Duration of Conflict in Case One

Total Contr-ibutors	Regular Contributors	Infrequent Contributors	Number of Days
17	9	8	1

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 subculture has influence on how a free software project is maintained. Reinhard, a core maintainer,

[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.

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 (note the **Immediate acceptance of outsider passage)** 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 people do be able to use free softwares.
…
<CyrilB> neilt: you are compromising our freedom by using non-free software: we can't modify and/or redistribute the source vector file.

CyrilB and neilt get into a heated discussion over the philosophical reasons for using free versus non-free software. In Neilt shows the tension between his **belief in free software and belief in freedom of choice** by stating that he feels that there is no reason to avoid non-free software if developing free software is about freedom. He continues to argue with CyrilB with statements such as:

<neilt> otoh i see no reason to avoid non-free software either
if this is really a freedom thing then we should be free to use whatever we want

He believes in freedom of choice yet feels hampered by the strict adherence to using free software to develop an open source system.

After several interchanges with CyrilB, Reinhard recalls from CyrilB's name that he is a regular particpant in the fsfeurope (Free Software Foundation in Europe). As such, he asks CyrilB for any other comments that he may have on GNUe. Again, this global belief system in the fsfeurope extends to the GNUe family as well. In fact, CyrilB reflects this attitude of the F/OSS development occupational community by suggesting that "you are compromising our freedom by using non-free software". People continue for the rest of the day to debate the issue of whether or not the use of non-free software to create the graphic is acceptable. During the debate, the **beliefs in free software and freedom of choice** are reinforced by the persistent recordation of the arguments. Through real-time testing and team discussion of free tool alternatives, the conflict is resolved.

The issue ends with a clarification by derek (a core developer) that contributors can use any tools necessary to create free software even if some are non-free tools. In addition, he indicates to CyrilB that if the graphical object were part of the free software product, GNUe (not for a graphical diagram intended for documentation), then he would be more concerned. The discussion ends with reinhard reminding everyone that the issue is resolved (with neilt agreeing to redo it at a future date with free software).

In the next case, conflict arises among frequent contributors when one contributor objects to the use of a non-free tool to complete documentation.

6. CASE TWO: CONFLICT OVER NON-FREE TOOL FOR DOCUMENTATION

The second case study explores project insider review of the procedures and practices for developing GNUe documentation (See http://www.gnuenterprise.org/irc-logs/gnue-public.log.15Nov2001 for the full three day logs). 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. Table 3 displays the number of contributors and their classification for participation in case two. This case illuminates the emphasis placed on strict adherence to the use of free software in the F/OSS occupational community.

Table 3. Contributors and Duration of Conflict over Documentation

Total Contributors	Regular Contributors	Infrequent Contributors	Number of Days
24	9	15	3

In order to understand this example, some background information is needed. The GNUe core maintainers selected a free tool to use for all documentation called *docbook* (http://www.docbook.org). DocBook is based on an SGML document type definition, which provides a system for writing structured documents using SGML or XML. However, several GNUe developers as of November 15, 2001 were having trouble with its installation. Consequently, they resorted to using lyx tool to create documentation (http://www.lyx.org). However, lyx requires the use of a non-free graphics package.

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
…

Maniac, a fellow contributor, questions chillywilly's incessant reminders about using non-free software. Chillywilly continues his debate showing his **strong view of free software**. Reinhard agrees with chillywilly as do others, but in order to complete the documentation, they agree to use an interim solution. Chillywilly is so adamantly opposed to the use of non-free software that he references Richard Stallman 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 emblamatic of the global belief system that drives the FSF and occupational community membership. Later Chillwilly is mollified by colleagues who persuade him with arguments like appealing to his **freedom of choice**. After many contributors explain to chillywilly that he can continue creating documentation in HTML or text format until a free GUI becomes available, chillywilly starts to back down and tires of talking about it anymore. For example, **mdean** suggests:

<mdean> chillywilly: you have a choice – which is what is "really" important
<**chillywilly**> mdean: I choose GNU whenever I can, that is **my** choice.
<**Mr_You>** Sorry your choice is a frustrating one
...
<**chillywilly**> whatever man I am burnt to a crisp.

Finally, chillywilly accepts the fact that no fellow contributors are willing to change the documentation tool to one that requires free software only. In the next section,w e discuss the beliefs and values shown in both cases.

7. DISCUSSION

Our analysis of this case study and related data reveals how the beliefs and values of the GNUe occupational subculture are manifested in conflict mitigation and resolution. F/OSS developers have unique and esoteric skills and ways of doing software development that differ from a proprietary in-house arrangement [28]. 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. High-speed networks enable the worldwide discussion of software development on live chat-lines, email, or mailing lists. For example, the GNU*e* project includes core software developers from the United States, New Zealand, Estonia, and the United Kingdom. "Meetings" occur over the Internet spontaneously or as planned. Often these developers view themselves as different from the rest of society and identify themselves as geeks [28]. This global occupational community of F/OSS developers shares beliefs and values that motivate them to contribute 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 in the efforts to resolve conflict.

In both cases, conflict arose over the use of non-free software for documentation. During data analysis phase of the research, we found that there is an interesting tension between the **belief in free software** and the **belief in freedom of** choice. In case one, the strict adherence to the free software principle (i.e. using all free tools for F/OSS software development also caused some developers to lose their freedom of choice when selecting graphical tools. In case two, chillywilly's strong beliefs in free software prevented him from having a freedom of choice in the tools used for documentation, even if the choice to use a non-free tool was temporary. He was not willing to compromise his principles.

Figure 3 illustrates the outside influence of the free software movement on the creation, routine work practices, and sustenance of the GNUe free software community. Some F/OSS contributors are motivated by the desire to establish a professional reputation as an F/OSS developer by his or her record of performance on an F/OSS project's archives (e.g. GNUe IRC archives or CVS), others by a desire to give something back to society or the "community at large" of F/OSS. Still others are interested in contributing for selfish desires for personal use of the software in businesses. These motivations are discussed in [28] and in results from recent surveys of F/OSS developers [13,

http://www.osdn.com/bcg/]. Many F/OSS developers contribute to multiple F/OSS projects simultaneously and hence, the global nature of the beliefs and values of the FSF are continuously reinforced and established via participation in various forms of CMC in free software communities.

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. In addition, other forms of CMC used by F/OSS developers need to be studied to determine the relationship between Internet communication forms and reinforcement of beliefs and values. For example, the DotGNU community lists a DotGNU Wiki on their website (http://wiki.dotgnu.org) which serves as an additional communications tool with comments being entered by contributors in an asynchronous manner (as opposed to the synchronous nature of IRC). Here we find free software beliefs such as:

> "DotGNU is about cooperation between developers, users, languages, platforms, and communications, without the cost of freedom." - Adam Ballai

Overall, the goal of this research is to develop a theory of how F/OSS communities develop software and how the cultural beliefs of the occupational community influence their work practices. Such theory is needed to address outstanding research questions such as: Is there something esoteric and special about virtual occupational communities and subcultures that differ from typical occupational communities? What is the free software business model and does it sustain employment over an extended period of time? Will virtual consulting (i.e. where consultants work from a distance to prepare a software change package for GNUe) be a norm of the future in these communities? Is there a relationship between strong cultural beliefs among F/OSS developers and a more successful, long term F/OSS project?

8. CONCLUSIONS

We have shown how the community spirit of the occupational subculture in the form of collective beliefs, values, and norms of the GNUe project fosters collaboration and resolution of conflict that emerge within the group of GNUe contributors during software development in a virtual community. Results have implications for software developers and managers planning to initiate F/OSS projects with similar global temporal collocation and virtual communication characteristics. The two main conclusions from the research follow, though others may be found elsewhere [9]:

1) Text-based CSCW systems and related artifacts in the form of IRC instant messaging streams, persistent IRC logs, digests (kernel cousins), and mailing lists help to build community and resolve conflicts.
Results from this study indicate that when someone like a newcomer or frequent contributor generates a debate on the use of free versus non-free software for GNUe development, the articulation of strong (somewhat polarized) views of free software strengthens the community. Anyone can come onboard in the middle of a debate, review the previous day(s) worth of IRC messages and/or mail list postings, and fairly quickly come up to speed on the issues. This fast access to archived information perpetuates the cultural beliefs that have been articulated and assists in resolution of conflicts. In addition, we concur with some of the empirical results of the SIDE model researchers [29] that the absence of individual cues enhances awareness of the group dimensions of identity and interaction, and can strengthen these social influence effects, in this case, cultural beliefs, values, and norms.

2) Strong organizational cultural beliefs in a virtual community tie a group together so that conflict is more easily mitigated and resolved.
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 [34] 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 engrained in the everyday work practices in their pursuit of an ERP system implemented as free software. These beliefs and values enhance and motivate mitigation and resolution of conflict despite the distance separation and amorphous state of the contributor population.

This research indicates the importance of recorded logs of instant messages for resolving conflicts in virtual work communities. The use of persistent online IRC logs and mailing list archives serves to tie the GNUe virtual work group together, contributing to conflict resolution and, at the same time, reinforcing the beliefs in free software and freedom of choice, and the more tacit values of cooperative work and community building. Managers of F/OSS development projects who are cognizant of the beliefs, values, and norms of the F/OSS development occupational community are better equipped to understand the "geek" culture among developers.

9. ACKNOWLEDGMENTS

The research described in this report is supported by grants from the National Science Foundation #IIS-0083075 and #ITR-0205679. 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.

10. REFERENCES

[1] Bechky, B. (2003). Sharing Meaning Across Occupational Communities: The Transformation of Understanding on a Production Floor. *Organization Science, 14*(3), 312-330.

[2] Bergquist, M., & Ljungberg, J. (2001). The Power of Gifts: Organizing Social Relationships in Open Source Communities. *Information Systems Journal, 11*(4), 305-320.

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

[4] Dube, L., & Robey, D. (1999). Software Stories: Three Cultural Perspectives on the Organizational Practices of

Software Development. *Accounting, Management and Information Technologies, 9*(4), 223-259.

[5] Easterbrook, S. (Ed.). (1993). *CSCW: Cooperation or Conflict*. New York: Springer-Verlag.

[6] Easterbrook, S. M., Beck, E. E., Goodlet, J. S., Plowman, M., Sharples, M., & Wood, C. C. (1993). A Survey of Empirical Studies of Conflict, *CSCW: Cooperation or Conflict?* (pp. 1-68). London: Springer-Verlag.

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

[8] Elliott, M. (2003). *The Virtual Organizational Culture of a Free Software Development Community.* Paper presented at the 3rd Workshop on Open Source Software, Portland, Oregon.

[9] Elliott, M., & Scacchi, W. (2003). *Free Software: A Case Study of Software Development in A Virtual Organizational Culture* (Working Technical Report). Irvine, CA: Institute for Software Research, University of California, Irvine.

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

[11] Glaser, B., & Strauss, A. L. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. New York: Aldine.

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

[13] Hars, H., & Shaosong, U. (2002). Working for Free? Motivations for Participating in Open-Source Projects. *International Journal of Electronic Commerce, 6*(3), 25-39.

[14] Herbsleb, J. D., & Grinter, R. (1999). *Splitting the Organization and Integrating the Code: Conway's Law Revisited.* Paper presented at the 21st International Conference on Software Engineering.

[15] Hine, C. (2000). *Virtual Ethnography*. London: Sage Publications.

[16] Jessup, L. M., Connolly, T., & Tansik, D. (1990). Toward a Theory of Automated Group Work: The De-individuating Effects of Anonymity. *Small Group Research, 21*(3), 333-348.

[17] Koch, S., & Schneider, G. (2000). *Results from Software Engineering Research into Open Source Development Projects Using Public Data.* Paper presented at the Wirtschaftsuniversitat Wien.

[18] Kollack, P., & Smith, M. (1996). Managing the Virtual Commons: Cooperation and Conflict in Computer Communities. In S. Herring (Ed.), *Computer-Mediated Communication: Linguistic, Social, and Cross-Cultural Perspectives* (pp. 109-128). Amsterdam: John Benjamins.

[19] Kollock, P., & Smith, M. (1999). Communities in Cyberspace. In M. Smith & P. Kollock (Eds.), *Communities in Cyberspace* (pp. 19). London: Routledge.

[20] Lea, M., & Spears, R. (1991). Computer-mediated Communication, De-individuation and Group Decision-making. *International Journal of Man-Machine Studies, 34*, 283-301.

[21] Mackenzie, A., Rouchy, P., & Rouncefield, M. (2002, February 25-26, 2002). *Rebel Code? The Open Source 'Code' of Work.* Paper presented at the Open Source Software Development Workshop, Newcastle-upon-tyne, UK.

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

[23] Mockus, A., Fielding, R., & Herbsleb, J. (2002). Two Case Studies on Open Source Software Development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology* (To appear).

[24] Mockus, A., Fielding, R. T., & Herbsleb, J. (2000). *A Case Study of Open Source Software Development: The Apache Server.* Paper presented at the ICSE '2000.

[25] Nard, B., Whittaker, S., & Bradner, E. (2000). *Interaction and Outeraction: Instant Messaging in Action.* Paper presented at the Conference on Computer Supported Cooperative Work, Philadelphia, PA.

[26] Olsen, M. (1971). *The Logic of Collective Action*. Cambridge, MA: Harvard University Press.

[27] Olsson, S. (2000). *Ethnography and Internet: Differences in Doing Ethnography in Real and Virtual Environments.* Paper presented at the IRIS 23, Laboratorium for Interaction Technology, University of Trollhattan Uddevalla.

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

[29] Postmes, T., Spears, R., & Lea, M. (1998). Breaching or Building Social Boundaries? *Communication Research, 25*, 689-715.

[30] Sawyer, S. (2001). Effects of Intra-Group Conflict on Packaged Software Development Team Performance. *Information Systems Journal, 11*, (155-178).

[31] Scacchi, W. (2002a). *Is Open Source Software Development Faster, Better, and Cheaper than Software Engineering?* Paper presented at the 2nd Workshop on Open Source Software Engineering, Orlando, Florida.

[32] Scacchi, W. (2002b). Understanding Requirements for Developing Open Source Software Systems. *IEE Proceedings - Software, 149*(2), 24-39.

[33] Scacchi, W. (2002c). *Understanding the Social, Technological, and Policy Implications of Open Source Software Development.* Paper presented at the NSF Workshop on Open Source Software.

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

[35] Smith, A. D. (1999). Problems of Conflict Management in Virtual Communities. In M. A. Smith & P. Kollock (Eds.), *Communities in Cyberspace* (pp. 134-163). New York, NY: Routledge.

[36] Sproull, L., & Kiesler, S. (1986). Reducing Social Context Cues: Electronic Mail in Organizational Communication. *Management Science, 32*, 1492-1512.

[37] Stallman, R. (1999). The GNU Operating System and the Free Software Movement. In C. DiBona, S. Ockman, & M. Stone (Eds.), *Open Sources: Voices from the Open Source Revolution* (pp. 53-70). Sebastopol, CA: O'Reilly & Associates, Inc.

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

[39] Trice, H. M., & Beyer, J. M. (1993). *The Cultures of Work Organizations*. Englewood Cliffs, NJ: Prentice Hall.

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

[41] Wenger, E. (1998). *Communities of Practice: Learning, Meaning, and Identity*. Cambridge, Massachusetts: Cambridge University Press.

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

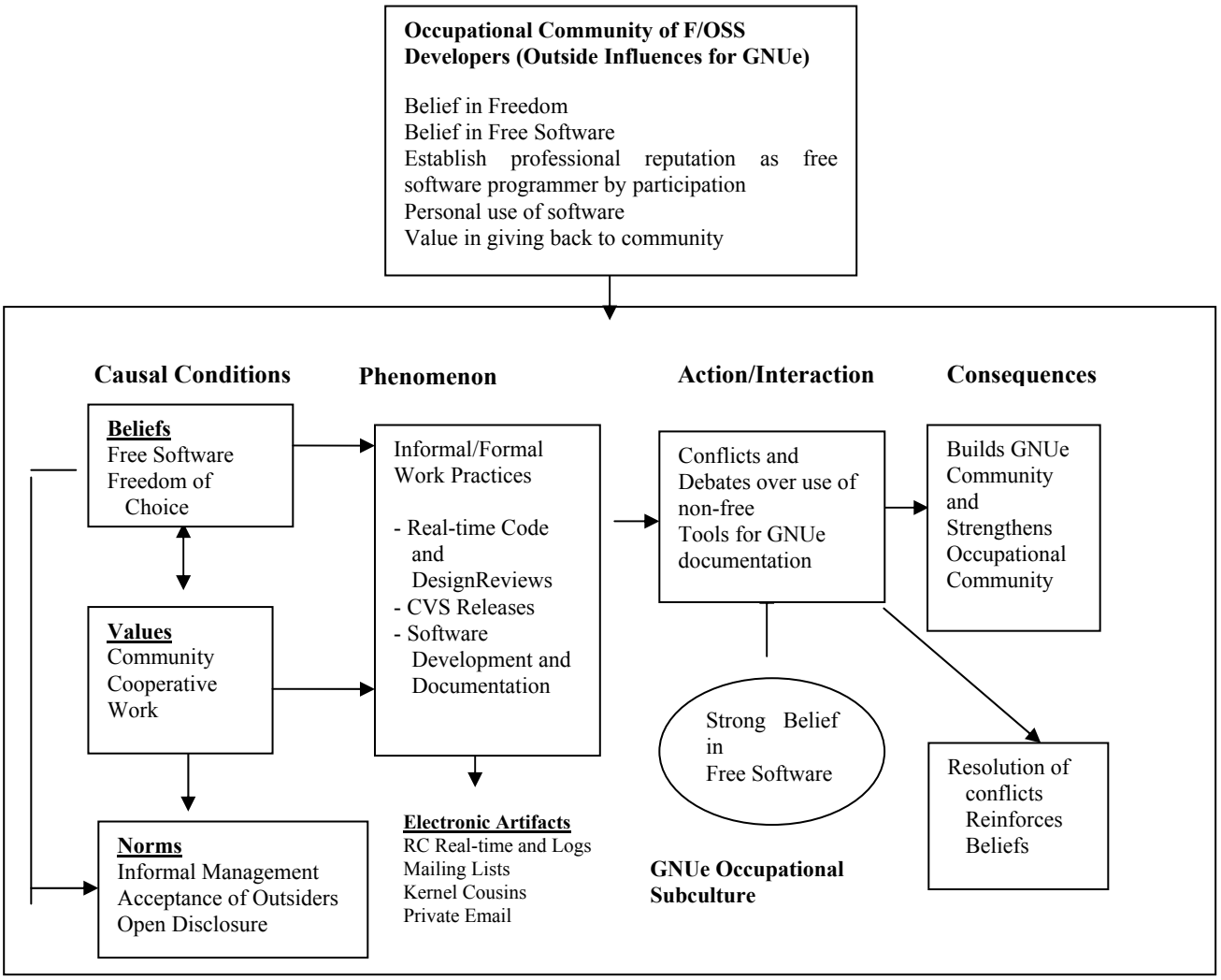

Figure 3. Conceptual Diagram of Relationships between GNUe Variable