

Users of Open Source Software – How do they get help?

Vandana Singh
University of Tennessee
vandana@utk.edu

Michael B. Twidale
University of Illinois at Urbana-
Champaign
twidale@illinois.edu

David M. Nichols
University of Waikato
dmn@cs.waikato.ac.nz

Abstract

A study was conducted across multiple open source software online technical help communities. This paper presents the types of discussions that occur, the types of questions asked and the type of responses that are given. The implications for socio-technical design are considered, exploring how help requests and discussions can be used to improve future help-giving, documentation and interface and functionality redesign.

1. Introduction

A criticism of open source software (OSS) as well as a major hindrance in its adoption is the apparent lack of technical support. Since many of these products come into being through a “programmers itch” [16], OSS developers are not necessarily inclined to provide support to the less technically adept end-users of software. Nor do they necessarily have expertise in developing documentation and manuals. This leads to products being produced but not being consistently supported. It has even led to a secondary market in selling a supportive infrastructure wrapped around free products – most notably in the case of various Linux products. However, although the OSS developers may not be providing support, fellow users do. Typically, open source software product websites have a support section that consists of multiple resources and channels of communication among the product users for the purposes of technical support. Our study focuses on the discussion forums present on these websites, meant for the users of the product to discuss among themselves any problems / issues with the software. As part of a larger body of work on online OSS technical support communities, this paper examines the nature of the questions asked and the responses given. We believe this can be useful both to consider how such analyses can be used to improve help-giving, and indeed the overall usability of OSS, but also may be of use in comparing these discussions with other somewhat similar discussions, such as technical help with proprietary software, in-house technical help forums,

other (non-technical) help forums, and highly technical forums not explicitly oriented to help-giving to end users. This paper situates our work drawing on similar contexts in multiple research areas such as online communities, open source software and discussion forums. We describe the data collection and analysis method, noting the ways that time and budget constrained out investigation. We highlight the main results regarding the nature of the discussions, the types of questions and the types of responses present in online technical support for open source software. Finally, we present a discussion along with design recommendations and future work.

2. Related Work

The OSS technical support bulletin boards that we studied are asynchronous, remote, mostly text-based, and provided by volunteers rather than paid experts. As such they have similarities and differences from the bulletin boards used in the OSS development process itself and from other kinds of help giving, each of which has its own research literature.

2.1. Research in Open Source Software

Research in OSS has examined many issues including the evolution of the developer community, developers’ motivations, adoption in commercial businesses, the culture of developers, policy issues, and problem solving. There have been only a limited number of multiple-project studies, reflecting the early stage of OSS research, the complexity of the phenomenon and the difficulty in obtaining comparable data across projects. Missing from almost all the OSS literature is a consideration of end users and their problem solving approaches.

The community of OSS developers has been studied, mostly large quantitative studies undertaken to see who the people are who contribute most to the code development, what the social network of the community members looks like and how projects revolve around some people. By contrast, there are

certain aspects of OSS development that have received less attention including: usability [13, 14], adoption, the users of OSS [19] and user driven innovation [24]. As noted by [21] “The least explored areas of OSS success included in the model are user impact and the role of user factors...users may vary in their ability to use and benefit from OSS based on factors such as their technical expertise.” Maass [9] sums up the present status of OSS field research as “most empirical studies on OSS communities concentrate on secondary logging information such as that provided by mailing lists, IRC chat logs and code repositories.”

The only work [6] we are aware of that looks at technical help for OSS, is mostly concerned with motivation, trying to answer *why* people bother to help others, and not really examining *how* the help unfolds, which is our main interest, and the focus of this paper.

2.2. Research on Online Discussion Forums

The study by Constant et al. [2] although not about OSS has similar data set to that of the present study: a set of technical help request emails being broadcast to an entire organization. The research showed that problems were solved by the weak tie relationship [4]; there was no need to know a person to answer their question and also that the usefulness of the information provided by the information givers was not based on the fact that the problem was solved or not. In some cases even though some information providers gave useful advice (as rated by help-seekers) they were not necessarily the ones that solved the problem and hence it re-affirms the hypotheses that even though the problem is not being solved but still the contributions are useful. This was validated in the study by the experts who said that despite the knowledge of the inconsequentiality of their response they were posting the answers. They knew that there were other people who could answer this question and also that they might not essentially be contributing to the problem solving process, they would contribute.

2.3. Research on Online Communities

Preece et al. [16] developed technology-oriented definitions of online communities by the software that supports them. They talk about the chat, bulletin board, listserv, UseNet News, MUDs (Multi-user dungeons), MOOs (Object-oriented MUDs), and web-based communities and emphasize that such descriptions are concise and meaningful to those who know about software. While they indicate what conversation protocols are like, they say little about social interaction in the community.

Butler et al. [1] studied a variety of work-related and non-work related online communities, excluding those that had medical or psychological support as their objective. They explored the question of why people contribute to these online communities, what do they get out of it and who are the people who contribute to the different types of activities in them. They found that it was not that owners or moderators spent much more time in the activities but that they spent more time in doing active work in the form of infrastructure maintenance; they did not differ in the time that they spent on reading and writing messages compared to the active or the silent participants of the community. They found out that the owners were involved in different types of activities than the other users and also perceived different types of benefits. The owners perceived the social benefits as being very important and that led them to spend more time in community building and maintenance activities.

2.4. Research in Online Customer Support

Online executives and Internet marketing academics alike agree that the need to develop a comprehensive understanding of consumer behavior in commercial online environments is urgent. Researchers in [15] assert that there has been a lack of genuine knowledge about what contributes to effective interactions with online customers, although previous research suggests that creating a compelling online environment for Web consumers will have numerous positive consequences for commercial Web providers. Web-based customer support systems can be used for internal and external customer support [8, 22]. The Web environment allows consumers to recover from their mistakes [7] and to overcome some difficulties associated with traditional media; e.g. it may ameliorate problems of accessibility, bottlenecks, interaction, and identification.

Nambisan and Baron [11] found positive benefits to customers of participation in interactive product-oriented environments in four main areas: cognitive/learning, community membership, personal status and affect. Although cognitive benefits (learning how to fix a problem with their product) are important participants reported that the process of interaction itself was valuable (even without a positive outcome). Wiertz and de Ruyter [27] suggest that in firm-supported online forums that user's contributions are most strongly influenced by the learning benefits, social effects and the user's propensity to interact online.

3. Studying Help

The aim of the study is to provide insight into the nature of communication of these online communities. The general research question is “What is being discussed in these user-based online support communities?” Specifically the study was designed to answer the following three questions.

- 1- What are the different types of things being discussed in these communities?
- 2- What are the specific types of questions that are being asked in these communities, which are common across websites?
- 3- What are the specific types of responses being provided to the questions of help-seekers in these communities?

3.1. Data Collection: Source and Criteria

Under the constraints of time and budget, we wanted to get a reasonably complete list of the different types of discussions. Consequently we collected twenty threads each from eight different websites making a total of 160 threads. These threads were collected by using stratified sampling approach. Only threads that were completed were sampled, where completion was defined as no new posting having been added in the six months before the collection date. Starting with the most recent thread to meet this criterion and working backwards, every 10th thread was selected to make up the 20 in the set for that website. Table 1 presents the list of websites used for data collection. These eight websites were chosen because they represent different types of open source software products – the categorization was adopted from the categorization of open source software products on the sourceforge.net website. This was done to ensure that different types of products were being used such as browsers, programmers’ tools, networking, collaborative software, application software, etc.

3.2. Data Analysis

Grounded Theory [3] was used as the basis for data analysis, an approach that is “well suited for phenomena that are emergent or poorly understood” [26]. In the process of grounded theory development, theory generation and development is done inductively by studying the phenomenon it represents. Concepts are discovered, developed, and provisionally verified through systematic data collection and analysis. One does not begin with a theory, and then prove it. Rather, one begins with an area of study and what is relevant to that area is allowed to emerge.

All the problem interactions were coded iteratively to develop some basic concepts, producing a list of categories for types of discussions, types of questions and types of responses. There are three stages of coding (open coding, axial coding & selective coding). In each of these stages, at least three iterations of coding were done to ensure all the categories are present and also to ensure that every time the same categories were identified. So, overall in data analysis each thread was analyzed at least nine times.

For reasons of economy, the coding of the data was done manually by one researcher. This has the advantage of eliminating concerns of inter-coder reliability in developing the categories, but raises concerns of intra-coder reliability and bias. These are at least partially addressed by the iterative nature of method itself, but bias remains a concern to be addressed in future work. Also, part of the aim of this work is that by disseminating the results we solicit independent validation / refutation / assessment of usefulness of the emergent categories by other researchers independently testing our categories on other data sets.

Although 160 threads is a small sample, particularly when compared to the sizes typically used in more quantitative methods, it needs to be noted that coding and recoding is an extremely lengthy and laborious process. The sample size was constrained by resources, but was we believe sufficient for the purposes of the study and for the particular results reported. As a low cost test of relative completeness, 5 more threads from each website were briefly examined (not formally coding the elements) and no new categories were found.

Multiple types of categories and concepts were discovered during this iterative coding process. In this paper we focus specifically on the content of the online discussion forums for technical support: the nature of the discussions, the types of questions and the types of responses available.

4. Results

Grounded Theory typically focuses on qualitative data. However we begin with a few simple (low cost to obtain) quantitative measures to give a rough sense of the scale of the issues being analyzed. As mentioned in Section 3.1, 160 threads were selected from 8 different websites. These 8 forums involve a large number of people as evidenced by the number of registered users. Excluding the most active forum studied (Firefox: exceptionally large), the 7 remaining websites had in total approximately 4 million registered users at the time of the data collection. Of course this is an over-count: some people could well be in more than one

forum and some may have multiple registered identities. However, one does not have to be registered to read these forums - only to post to them. So these numbers may well be an undercount of total forum usage. The view indicates heavy use: 50% of interactions are viewed 100-200 times and 35% are viewed 200-500 times, although of course not all views are productive for users. Nevertheless these crude numbers give a sense of the overall size and significance of the users of open source software engaged in technical support.

Each thread ranged from 1-8 messages in length with a median of 4. A thread involved from 1-7 people with an average of 3-4. The time from posting a question in a new thread to receiving some kind of response is good overall. It is a very impressive one minute for a few messages (3%) and almost half (48%) messages had a response in less than an hour. Within 12 hours, 82% of the questions had received a response. There were a few (3.3%) messages that did not receive any response after one week.

From the above we can see that despite being a service provided by volunteers independently choosing to answer postings, help forums in general seem to be responsive. But are they effective? Deciding from the messages alone whether a help interaction was effective can be difficult – just because the help given is correct is no guarantee that the help seeker can understand it or use it. A very conservative criterion was used to categorize a thread as ‘solved’: whether the help seeker posted a message saying so. If the help seeker (or anyone else) tried the proposed solution and it worked but they never came back to post a message telling the forum that it worked, then the interaction was categorized as unsolved. So, an explicit “thank you, that worked” message had to be present for a thread to be categorized as solved.

This strict criterion gives us a safe undercount measure for success. Out of the 160 threads, about 30% met this strict definition of solved, 60% were unsolved, and the remaining 10% were split roughly evenly between questions that did not get responses and posts that were not questions.

On average the threads were active (time from first request to last response) for 14 days, and the average time for receiving the first response was 3 days 8 hours. Given the very rapid response to many requests, this indicates a long tail response effect. Almost twenty percent (18.5%) of problems received their last response with an hour. This does not necessarily mean that they were solved in less than an hour but it means that these threads were not getting any more messages after one hour. A majority of threads (60%) were over in less than a day, 81% of threads were over in less

than a week and only 3% of threads were active for more than a month.

The amount of time taken to solve a problem is impressive. Just over half of the solved threads were solved in less than one day! Almost 75% of the solved problems were solved in less than a week and only 8.5% problems took over a month to be solved. All the solved problems got their first response within a week. Over 90% of the solved interactions got their first response within a day.

For one of these eight communities (Firefox), we identified the core members of that community – those who posted the largest number of messages.

Table 1. List of Projects for Data Collection

	Websites
1	NVU
2	Opera
3	Filezilla
4	phpMyAdmin
5	phpBB
6	Dropline
7	MozillaFirefox
8	Moodle

As Table 2 shows, these members had an extremely high level of contribution both in the forum as a whole and in our sample. The most active contributor has nearly 36,000 posts on the forum in their 22 months on the forum. The number of months on the forum since first registering does not seem to have a high correlation to the number of messages posted.

Table 2. Core Member Statistics

Username	Months on Forum	Total Messages in Dataset	Messages on Forum
CoreMem1	22	124	35719
CoreMem2	21	40	24432
CoreMem3	51	14	20913
CoreMem4	54	24	14342
CoreMem5	28	15	4592

This disproportionate concentration of effort parallels that found in open-source development communities: 4% of members account for 50% of answers on a user-to-user help site [5] and 4% of developers contribute 88% of new code and 66% of code fixes [10].

4.1. Types of Discussion

Considering all threads, discussions can be categorized into five categories, as shown in Table 3 along with their relative frequency, and elaborated below. As expected, a large majority (76%) of

interactions are problem solving. However, in this data set across eight websites that still leaves almost a quarter of interactions in other categories. Most of the remaining threads are information seeking discussions like “Does this exist?”, “Where can I find ‘x?’”, etc. Six percent of threads have no questions, no information seeking or dissemination but are social discussions that are not necessarily related to the product.

1. *Problem Solving* – This is the most common category where the help-seeker does not know what is wrong or how to do a certain thing. The interaction typically consists of one person posting a question and others posting back responses to either ask more detailed clarificatory questions or to provide solutions. These problem solving interactions are discussed in detail in sections 4.3 and 4.4 under the types of questions and types of responses.

Table 3. Types of Discussions

Types of Discussion	Occurrence (%)
Problem Solving	76
Information Seeking	14
Social Discussion	6
Feature Request	2
Information Dissemination	2

2. *Social Discussion (about the software, about the forum, about the help process, thanks interactions)* These interactions are not about any specific feature or problem or solution but are general discussions about the software and the discussion forums.

3. *Information Seeking Discussion* - Unlike problem solving, here the help seeker has a clear idea of what they want, but just need information on locating it.

a) *Where to find x?* – This was the second most prevalent category, where one person would ask where to find a site to upgrade, a piece of documentation, or an extension.

b) *Need Information / Recommendation* –Interactions that are not really about solving a problem or necessarily about the software in question, but just general help with computer problems and asking for recommendations from the people on the forum. An example is a request at Mozilla Firefox Community for recommendations on a registry scan.

“I keep on getting these annoying messages about my registry being corrupt and needs to be scanned and cleaned. I don't understand this considering this is a brand new hard drive that I just had installed 2 days ago.

If anyone has any suggestions on a free registry scan please post.”

c) *Does this Exist* – These were the messages exploring the functionalities of the software in question and would generally take the form “Can I do this?”, “Is there a way to do this?”Etc. Sometimes, there was a comparison with competitive software or the previous version. Some questions were of the form: “this feature was present in the previous version / software I used; does it exist in this one?”

4. *Information Dissemination* These are not problem solving but tips for the software use. They are also presented as work-arounds to some existing known problems with the software.

“I installed a fresh copy of Dropline Gnome 2.14.1 on a brand new Dell and Slackware install and found that Gaim wouldn't stay running. It turns out that it crashes with the following:
Code: Creating link
/home/[i]account[/i]/.kde/socket-freyja. can't create mcop directory where freyja is the hostname for my system and account is my login account home. The solution is to do the following as the appropriate login (modify it accordingly for your situation):

Code: mkdir -p
/home/[i]account[/i]/.kde/socket-freyja I couldn't find anyone else in the forum with the problem, so thought I'd just post it here.”

5. *Feature Request Discussion* As the name indicates this are the messages that are requesting a particular feature in the software and show the process of feedback to developers in the responses that followed these requests.

“.... I gave opera 8.5 a shot and found it faster and better behaved. I would forusre go and run opera 8.5 if the "block content" was there. There are many a smart person and "computer geeks" that post here. Maybe a modified version of 8.5 could be released. Not sure if it would work. Not sure it can be done. Its just my humble opinion that a 8.5 with a good ad blocker would be a stripped down version of 9.0. ...”

From these five categories of discussion types, we can say that these forums, though essentially used for solving problems, are also important in other aspects. Specifically, we find the categories about information seeking, information dissemination and feature request to be crucial. These threads are intermingled among all the threads, but we propose that these are also very

special contributions of these users and should be harvested for important bits of information. For instance, the information dissemination category has tips and work-arounds to successfully perform a particular task using the software. This is an important category where the software does not do this work, but the users have appropriated a smart use of it, to get their work done. Also, the tips and work-arounds are suggested are the points of user initiated innovation and must be utilized for that. Just as a starting point in the forums, these tips / work around can be presented as sticky threads (which remain at the top of the list of threads in the forum, rather than moving down the list as new threads are created) This would preserve them for re-use and avoid their loss in all the other messages. Also, these information dissemination messages should be made more visible than the average question and answers because they will potentially serve a larger number of users.

We found the *potential* of explicit and implicit feedback to the product developers. The category of Feature Request contains messages where users are specifically asking the developers to add certain functionality that will be useful in getting their work done. In our study, we did not explore evidence of this feedback to the developers being utilized but we strongly believe that these messages should have a way to make it to the developers. These can just be forwarded to another list or can be collected and send to the developers of the software as direct feedback from the users. We also found instances of indirect feedback to developers through the information seeking messages categorized as “Does this exists?” These messages once again, are looking for a feature that the users think should be in the software and would be useful but are not aware of its presence. This means one of two things; either that this feature is not present and hence this can also be used as a feature request message or that the feature is present but is not visible to the user and hence the usability of this feature needs to be re-evaluated. In both the types of feedback to the developers, once a threshold of messages is reached, the feature should move to the priority list of developers, thereby providing direct feedback to the developers.

4.2. Types of Questions

Questions posed in online discussion forums can simply be divided into two broad categories: those asked by the help-seekers and those asked by the help-givers. Subsequent analysis gave a number of sub-categories. The help-seekers posted questions about how to get things done, why am I stuck, what is wrong, etc. while the help-givers asked about background

information of the help seeker’s computer, the task that they were doing, the results of following the steps that they have taken, and so forth. Table 4 and Table 5 show the categories that emerged after in-depth coding. Both questions and responses arise from non-unique coding – a posting can contain multiple questions, responses or both.

Table 4. Questions by Help-givers (Follow-Up Questions)

1	Background Information questions	OS? Browser?
2	Clarification questions	what says phpMyAdmin/ libraries/common.lib. php on line 2979
3	Give a screenshot	I really can't see and find these icons. Can you screenshot it?
4	History Details	Any extensions? And have you cleared your downloads history lately?
5	It works for me, did you try this	Works for me. Did you clear your cache and try it again? If that doesn't work, close Firefox and delete mimeTypes.rdf from your profile: http://kb.mozillazine.org/Profile folder# Where_is_my_profile_folder.3 F
6	Repeat the question	what is your problem then, upload or displaying uploaded table?

4.3. Types of Response

A response typology was developed to categorize the different types of responses posted on these online discussion forums as illustrated in Table 6. The ten types of responses are as follows:

1. *Instructional* – This category represents the responses that were usually given by help-givers and consisted of explicit, detailed, step-by-step instructions, code for achieving the solution.
2. *Explanatory* – These are the messages that were written either by help-givers or by help-seekers. When help-givers write explanatory messages, they give details of the software processes; explain functionality and logic of the software activities. Help-seekers are explaining what they have done on their own before reaching the bottleneck.
3. *Informational* – This category represents the messages that are providing basic information about the software or the forums, these also can come from either the help-seekers or the help-givers. When help-givers write informational messages they are telling from their experience in the forums.

4. *Re-directional* – This is a very important category of responses. It consists of the messages, usually from help-givers, directing help-seekers to solutions. For example, the help giver will point the help – seeker to a previous thread / wiki / manual / documentation with solution to their problems.

5. *Verificational* – The responses that fall in this category are “me too” responses. These are responses from help-seekers and might seem like noise and not useful, but, as we found out in our data, these “me too” responses serve two purposes. Firstly, they verify the

initial help-seekers problem by saying that they are having the same problem. Secondly, and more importantly, they provide contextual information about the problem to the help-givers and help in formulating the complete problem.

6. *Innovative* –These are the responses where users post some innovative solutions like work-around or tips. These are sometimes solutions and sometimes unsolicited responses with the hope of common good. The users post a trick or special appropriation of the software.

Table 5. Types of Questions by Help-seekers

Type of Questions		Examples
1	Any suggestions / ideas	after the last two updates, I get a regular "Server not found" screen. After I click the "try again" button 4 or 5 times, the page comes up. I have checked everything I can think of and can find no fix. VERY annoying and time consuming. Any suggestions?
2	Anyone else has this problem	I just updated my firefox and now, every time I open the browser, the toolbar buttons on my Yahoo! toolbar don't load. When I "click here to retry" they come up fine. Does anyone else have this problem?
3	Can this be done / Is it possible to do this?	"Is it possible force use another charset (Win1251, for example) to make Filezilla Server compatible with older clients (like FAR Manager, for example)?" "I use Moodle as file manager in topic format is it possible to get the topic number larger than 52?"
4	Comparison	Is it just me or FFx 2.0 RC 3 is a bit faster than Opera?
5	Does this exist?	Is there an extension that replicates this Opera feature in Firefox?
6	Feature Request/Exist	Opera seriously needs a better ad-blocker. The ad-blocker that come with Opera 9.02 is horrible IMHO. I have to manually pick what to block! I don't think so; you can't even block some ads.
7	Error message	Anyone any idea why we are getting a 403 forbidden error code?
8	Doesn't work	Media player and Opera not working properly together. When using the next/previous/play etc buttons on my keyboard, they don't work when the Opera window is active
9	How to	Anyone knows how to bypass the Login screen?
10	Information needed / Recommendation	Any recommendations for any extension that can support unlimited note pages (preferably in tab style like foxnote) for firefox?
11	Is it a bug? Is there a workaround?	The paragraph of body text overlaps the heading on my webpage when viewing in Opera. Is this a bug in Opera? Is there a workaround?
12	What to do?	When I "Print preview" web pages the images dont appear. What setting do I need to Change?
13	What am I doing wrong?	the page I am trying to edit on my site does not go to the webpage that I can see on the Internet. I can see it under one of the lists in the site manager and whenI click on it it appears in the window next to the site manager. Does anyone have any suggestions on what I am doing wrong.
14	What is happening?	This error should happen once or twice, but it keeps happening, whats going on?
15	What is wrong?	These are my settings can anyone see what is wrong?
16	Where	Where to find NVU compatible pre-built template?
17	Why does this happen? / Why is this happening and how can I fix it?	Opera keeps freezing up and making my computer go slow and i have to push shift alt and delete and kill it. Why does this keep happening? I hyperlinked text in my instruction to the location of the forums. all those links appear to be removed/defaulted to something else/ written in gibberish in the last few days today they are not the same. What can cause this and how can I make sure the links retain their original address?

- 7. *Speculative* – These are the messages where the help giver is not sure if what h/she is proposing is a solution or not but they speculate and ask the help seeker to try something and report back to the forum.
- 8. *Negational* – These are the responses that are not adding anything to the conversation and posted by help-givers who volunteer and will say that they don't know how to solve this problem.
- 9. *Solicitational* – The messages in this category are very common as follow-up questions by the help-givers. In these messages the help giver is the details about the software, operating system, work history, etc. These questions form a very large part of the iterative process of problem solving and automating these iterations can lead to much quicker solution.
- 10. *Social* – These messages usually do not contribute to the process of problem solving but are expressions of, or comments on, etiquette. These messages include messages like “thank you”, “you are welcome”, “Search the forum before asking”, etc.

Table 6. Types of responses

Types of Responses	
Instructional	A better way to do this
	Code
	Small direct solution
	Step by step instructions
	Do this, this is the solution,
Informational	Fixed in next version
	Not a problem for this product
Re-directional	Link to external resources
	Link to Internal Resources
	Link to Previous Thread
	Link to Wiki
	Read FAQ / Read the manual
Innovative	No can't do that but workaround
	This is what works for me
	Tips
Speculation	Suggestion for a solution
	Try this
Verificational	Me Too
Explanatory	Detailed Explanation
Social	Don't waste our time
Negational	I don't know
Solicitation	Background Info questions
	Clarification questions
	Give a screenshot
	History Details
	Repeat the question

5. Discussion

It can be seen that although entirely volunteer-based, OSS technical help is on the whole remarkably timely and effective. However, this voluntary nature means that there is a substantial variance in both speed and effectiveness; posting a request is no guarantee that anyone will bother to reply, or reply in a reasonable time. Responsiveness shows a long tail effect. Also, there is inevitably a selection effect – we are only seeing those people who go to the forums and ask for help. We don't see those who are unable, unwilling or intimidated to ask for help in such a public way. We also don't see those who manage to get help from other sources on the website (various documentation including help files, tutorials, blogs or wikis) or from other less 'official' sources. We also do not see those who get the information they need from simply reading pre-existing threads. However, we believe this is a useful start to understanding online technical help in general and OSS technical help in particular. Although a typical help-giving interaction might consist of a request, a reply containing a set of contextual questions to help clarify the problem, some problem-solving, one or more proposed solutions and comments on their efficacy, there are a number of other variant interactions. In other work we look at issues of media use and the use of external resources to support the help-giving, problem-solving and clarificatory processes. In this paper we note the kinds of questions and responses that commonly occur. Although our current focus is on understanding OSS technical support and how it can be improved, we are interesting in future work in comparing the categorizations found here with other help-giving settings. Do the categorizations also apply in those other contexts?

The open source technical help forums obviously have similarities with open source technical development: open to all to participate, reliant on volunteers choosing to work on a particular need, having emergent leadership arising in the form of a few core members who make disproportionately large contributions and a long tail of participants with smaller contributions. However, there are also differences. The barriers to participation are lower. It is typically easier to offer help on how to use an application than to recode that application. Some help (particularly to novice users) is relatively easy to provide and does not require specialist knowledge. Some help is simply a matter of pointing to pre-existing resources that address the problem raised. Other is a matter of explaining or re-contextualizing existing help so that the recipient can understand what they need to do.

Other requests are much more difficult, rather like the 'stumpers' that skilled reference librarians relish as

worthy problem-solving challenges to their expertise (see [19] for a comparison of technical help with the reference interview).

Online technical help predates the current growth of open-source software, and can be seen in various Usenet news forums for a range of technical areas aimed at both novices and expert users. Over time an etiquette of question asking and responding has evolved and continues to evolve. For example it is considered good manners to try and search for relevant help before asking, and to indicate in one's request that one has indeed done so.

Modern bulletin board functionalities and interfaces have been adopted by technical help websites to good effect. However it is worth considering how help-giving could be made more effective. Many possible changes are non-technical, involving recruiting more help-givers and providing better education and orientation to both help-seekers and help-givers. But there are also technological interventions that might help, either by lowering the costs and effort of help-seeking or help-giving, improving their effectiveness, supporting the development, refinement and re-use of help resources, or by feeding user-centered results back into the OSS development process. Here we list some examples, inspired by our analyses.

- (Semi) Automation of contextual information to be supplied by the help-seeker. A good help request needs to provide information such as the hardware and operating system being used, the version of the application, what the user is trying to do, what has been tried (and presumably failed) and which help resources have been completed. Much of this information could be obtained automatically if it was an option built into the application, and the rest might be supported, (and the user encouraged to provide it) by a suitably structured form.
- Support for refactoring a help request into a FAQ or online help documentation wiki
- System support for volunteer updating of help: taking a help explanation and updating it for a new version of the software.
- Lightweight measures of re-use. We know people are consulting 'dead' threads - would a simple 'this worked for me too' button get used?

The observations made in this study lay a path for improvement by highlighting the multiple uses that can be made of these interactions. Here we present some of the uses that we believe have strong potential of improving the current process. Help requests can be used as feedback for the redesign of both the system and its help. Some help requests are pointers to usability problems with the software and hence can be highlighted for analysis by the OSS User Interface team. They will have documented evidence of the users

having problems with the application and can provide some quick semi-fixes to the interface and eventually produce longer more complex full solutions or design to obliterate the problem.

Creating a better visualization of the entire discussion database will facilitate the assessment of patterns of help requests, such as which parts of the system seem to cause disproportionate confusion. It could also allow organizational improvement through supporting queries such as:

- Show me all threads that point to a particular piece of documentation (help, FAQ, etc.)
- Bundling of threads: show all threads with a similar topic. Are they repeats? Can they be connected, rejoined, or better tagged to avoid duplication and aid re-use?

6. Future Work and Conclusion

Future work in this area could include triangulating these results with studies using alternative methods such as interviewing forum participants [26]. If these categories are found to be significant then there may be implications for design of support environments and their integration with both formal and static help sources (such as manuals) and with the software application itself. A wider issue is whether the findings are transferable to proprietary software or beyond software to physical products and services [11]. Extending the work in this way would help determine whether the nature of the OSS software and its community have an identifiable influence on the pattern of help-related activities.

We would also like to compare this process with other settings including other online technical help, in-house technical forums, non technical help, technical discussions not purely help related and tech forums in Usenet, around mashups, etc. West & O'Mahony [26] compare autonomous and sponsored OSS communities in terms of software production, governance and intellectual property; it would be interesting to explore whether this dimension also influences user-support activities. A further extension could be to consider product-related forums to investigate how the nature of the topic (software, physical product or service) influences user contributions [11]. We will also be conducting analysis on this data with respect to information re-use and tools to aid the process of re-use. We view this paper as an exploratory precursor to more detailed qualitative studies. In particular our methodology complements other approaches such as those based on questionnaire surveys [27], interviews [26] or data mined from public mailing lists [20]. We suggest that these methods on their own are unlikely to

be sufficient for exploring in detail how online help actually unfolds.

In this paper we present some results about the content of online discussion support forums from a larger study. We have shown that the forums do not just contain straightforward help requests. We have presented a detailed list of categories of questions that are asked. We highlight the “how” of the process of problem solving by showing the different types of responses for problem questions and present a typology for these responses. This paper presents multiple pragmatic approaches to improve the process of collaborative problem solving in open source software by some quick fixes and by highlighting multiple points of inquiry for research.

7. References

- [1] Butler, B. Sproull L., Kiesler S. & Kraut R. (2002) “Community Effort in Online Groups: Who Does the Work and Why?” In L. Atwater and S. Weisband, *Leadership at a Distance*.
- [2] Constant, D., Sproull, L. and Keisler, S. (1996) “The Kindness of Strangers: The Usefulness of Electronic Weak Ties for Technical Advice.” *Organization Science*, 7(2), 119-135.
- [3] Glaser, B.G. and Strauss, A.L. (1967) “The discovery of grounded theory: strategies for qualitative research.” Chicago: Aldine.
- [4] Granovetter, M. (1973) “The Strength of Weak Ties.” *American Journal of Sociology*, 78(6), 1360-1380.
- [5] Lakhani, K. R. and von Hippel, E. (2002) “How Open Source Software Works: ‘Free’ User-to-User Assistance.” *Research Policy*, 32, 923-943.
- [6] Lakhani, K.R. and Wolf, R.G. (2003) “Why hackers do what they do: Understanding motivation and effort in Free/Open Source software projects.” Technical report, MIT Sloan School of Management.
- [7] Lenz, V. (1999) “The Saturn Difference: Creating Customer Loyalty in Your Company.” New York: Wiley.
- [8] Longoria, J. (1996) “Focus on: help desk applications and technologies.” *Telemarketing and Call Center Solutions*, 148, 52-55.
- [9] Maass, W. (2004) “Inside an Open Source Software Community: Empirical Analysis on Individual and Group Level.” *Proceedings of the 4th Workshop on Open Source Software Engineering*, Edinburgh, Scotland, 65-71.
- [10] Mockus, A., Fiedling, R.T. and Andersen, J.D. (2002) “Two Case Studies of Open Source Software Development: Apache and Mozilla,” *ACM Transactions on Software Engineering and Methodology*, 11 (3), 309-346.
- [11] Nambisan, S. and Baron, R.A. (2007) “Interactions in virtual customer environments: Implications for product support and customer relationship management,” *Journal of Interactive Marketing*, 21(2) 42-62.
- [12] Neff, G. and David S. (2002) “Permanently Beta: Responsive Organization in the Internet Era.” Center on Organizational Innovation, Working Paper, Columbia University.
- [13] Nichols, D.M. and Twidale, M.B. (2003) “The usability of open source software.” *First Monday* 8, 1
- [14] Nichols, D.M. and Twidale, M.B. (2006) “Usability processes in open source projects”, *Software Process: Improvement and Practice*, 11(2), 149 – 162
- [15] Novak, T., Hoffman, D. and Yung, Y. (2000) “Measuring the Customer Experience in Online Environments: A Structural Modeling Approach.” *Marketing Science, INFORMS*, 19(1), 22-42.
- [16] Preece, J., Maloney-Krichmar, D. and Abras, C. (2003) “History of Emergence of Online Communities.” In B. Wellman (Ed.), *Encyclopedia of Community*. Berkshire Publishing Group, Sage.
- [17] Raymond, E.S. (1998) “The cathedral and the bazaar.” *First Monday*, 3(3).
- [18] Sandred, J. (2001) “Managing Open Source Projects.” Wiley.
- [19] Singh, V., Twidale, M.B., and Rathi, D., “Open Source Technical Support: A look at peer to peer help giving.” HICSS-39, Hawaii, January 2006.
- [20] Sowe, S.K., Stamelos, I., and Angelis, L. (2008) “Understanding knowledge sharing activities in free/open source software projects: An empirical study” *Journal of Systems and Software* 81(3) 431-446.
- [21] Stewart, K.J. (2004) “OSS Project Success: From Internal Dynamics to External Impact.” *Proceedings of the 4th Workshop on Open Source Software Engineering*, Edinburgh, Scotland, 92-97.
- [22] Tourniaire, F. and Farrell, R. (1997) “The Art of Software Support.” Upper Saddle River, Prentice-Hall.
- [23] Vixie, P. (1999) “Software engineering.” In *Open Sources: Voices from the Open Source Revolution*, C. Dibona, S. Ockman, and M. Stone, Eds. Sebastopol, CA: O’Reilly, 91-100.
- [24] Von Hippel E. (1988) “The Sources of Innovation.” New York: Oxford University Press.
- [25] Weber, J. (1999) “The bottom line.” *The Industry Standard*, 5, August 2-9.
- [26] West, J. and O’Mahony, S. (2008) “The Role of Participation Architecture in Growing Sponsored Open Source Communities”. *Industry & Innovation* 15(2) 145-168.
- [27] Wiertz, C. and de Ruyter, K. (2007) “Beyond the Call of Duty: Why Customers Contribute to Firm-hosted Commercial Online Communities”, *Organization Studies*, 28(3) 347-376.