How Open Source software works: "Free" user-to-user assistance

Karim Lakhani* and Eric von Hippel**
May, 2000

MIT Sloan School of Management Working Paper #4117

* Consultant, The Boston Consulting Group (lakhani.karim@bcg.com)
** Professor, MIT Sloan School of Management (evhippel@mit.edu)

Acknowledgements

The response and cooperation of the Open Source software community towards our study was tremendous. We would like to thank the participants on the Usenet newsgroup comp.infosystems.www.servers.unix who responded to our online survey. We would also like to thank the following for providing us with key insights into the Apache community; Brian Behlendorf, Ben Hyde, Robert Thau and Matthew Gray. We also received tremendous help from Ben Ho and Starling Hunter during the technical and analytical phases of our study. Access to Usenet archives was provided by Deja.com.

How Open Source software works:
"Free" user-to-user assistance

ABSTRACT

Open source software products represent the leading edge of innovation development and diffusion systems conducted for and by users themselves – no manufacturer required. Research into this phenomenon has so far focused on how the major tasks of software development are organized and motivated. But a complete user system requires the execution of "mundane but necessary" tasks as well.

In this paper, we explore how the mundane but necessary task of field support for open source Apache server software is organized, and how and why users are motivated to participate in providing it. We find that the present system works well and that information providers are largely rewarded by benefits directly received from a related task. We also find, however, that the present help system is by and for only a few – and that it changes would be needed if and as volume increases. General lessons for user-based innovation systems includes the clear willingness of users to openly reveal their proprietary information. This bodes well for the efficiency of user-only innovation systems, and is rational behavior if the information has low competitive value and/or if information providers think that other users know the same thing they do, and would reveal the information if they did not.

How Open Source software works: "Free" user-to-user assistance

Karim Lakhani and Eric von Hippel

1.0: Overview and Problem Statement

Some very successful "open source" software products have been and are being developed, distributed, and supported in the field by and for users themselves – no supplier required. The motives that induce participation and the mechanisms by which the various aspects of this feat are being achieved are also being evolved by users as they participate in such software development projects.

We propose that study of these evolving mechanisms for complete user-to-user innovation development and consumption systems is of major and general importance. It has been shown that users are the developers of important innovations in many fields. It has also been argued and to some extent shown that the locus of innovation is shifting increasingly towards users over time (von Hippel 1998). If this is so, the organization of user-only innovation systems, observable today in economic niches such as open source software, is potentially a matter of general interest and relevance in the future.

To this point, explorations of the mechanics of and the incentives to participate in open source software projects has barely begun – and what has been done to date has focused on the core tasks of developing and debugging and improving the open source software itself. Motives used to explain why users would voluntarily work on these basic tasks include: (1) a user's direct need for the software and software improvements worked upon; (2) enjoyment of the work itself; (3) the enhanced reputation that may flow from making high-quality contributions to an open source project. But a complete open source software development and diffusion system contains mundane but essential tasks as well – and the three motivations just described seem to apply relatively poorly to these. We, therefore, devote this empirical exploration to understanding why and how a

task at the mundane but necessary end of the scale gets done.

The "mundane but necessary task" we have elected to examine is the delivery of high-quality "field support" to users of open source software. Field support involves provision of assistance to users having difficulties with a product – in this case an open-source software product - because of defects in the product itself or because of the state of the user's own understanding. Commercial software vendors charge users for field support either directly or indirectly. Open-source software does not generally involve a charge for field support. Instead, some product users voluntarily provide answers to the questions of other users – apparently for free.

A number of possible explanations have been put forward as to how and why such a system might work – with the primary puzzle being why information-providers expend the effort needed to help others who ask questions. Proposed motives include altruism; incentives to support one's community; reputation-enhancement benefits received by information providers; and expectations of benefits from reciprocal helping behavior by others ("I help today because I have been helped in the past and/or I expect to be helped in the future.")

Our empirical study shows that, at least in the case of Apache, the need to call upon any of these explanations exists only for a small portion of information-providers participation in the field support system. We find that most of the effort information-providers expend can be understood in terms of direct rewards they immediately receive. Thus, we find that the major cost in providing help, matching a question with a willing and able information provider, is carried out by providers primarily to gain learning for themselves and not to provide help to questioners. The cost of actually delivering help is generally very low – providers only transfer information they already know to questioners, and typically expend only 1-5 minutes on that task per message.

Our examination of help system performance also characterizes those who

deliver and receive help, determines the frequency with which questioners get their questions successfully answered and so forth. We find that the Apache Usenet help system currently performs well but has a design that probably would not scale well to high volume processing of questions and answers. Currently relatively few post to the system – strikingly few given the millions of extant Apache sites. The detailed understanding we gain of the Apache online help system allows us to think very specifically about possible modifications and improvements to such systems. Also and more generally, it helps us to deepen our understanding of complete user-to-user innovation development, diffusion and consumption systems.

In this paper our next step is to describe the context of our empirical research (section 2). Then we review extant literature bearing on our topic (section 3). Next, we describe our research methods (section 4). Then we report our findings under three headings: participation in the Apache help forum (section 5); effectiveness of the Apache help forum (section 6); cost and benefits to help forum participants (section 7). Finally, we discuss the implications of these findings for open source help line design in particular, and user-based innovation systems in general (section 8).

2.0: Apache, an "Open Source" software program

Apache is web server software used on "web server" computers connected to the Internet. A web server's function is to "service" requests from Internet browsers for particular documents or content. A typical server waits for client requests, locates the requested resource, applies the requested method to the resource, and sends the response back to the client. Web server software began by offering relatively simple functionality. Over time, however, Apache and other web server software programs have evolved into the complicated "front end" for many of the technically demanding applications that now run on the Internet. For

example, web server software is now used to handle security and authentication of users, provide e-commerce shopping carts and gateways to databases.

Apache, like most early web server software programs, was developed by a user - Rob McCool, who developed it for and while working at the National Center for Supercomputing Applications (NCSA) at the University of Illinois (It was developed in conjunction with Mosaic, the first web browser and predecessor to Netscape, which was also developed at the University of Illinois.) When McCool left NCSA in the middle of 1994, a small group of web masters who had adopted NCSA server software for their own web sites decided to take on the task of continued development for themselves. A core group of eight individuals began the work by gathering all documentation and bug fixes that had been made for NCSA server software up to that point. They put this material together in the form of a consolidated patch. Over time, the name of this *patchy* web server software evolved into Apache. After extensive feedback and modification by users, Apache 1.0 was released on December 1, 1995. In the space of four years and in the face of strong competition from commercial competitors like Microsoft and Netscape, the Apache web server has become the most popular web server software on the Internet, used by more than 60% of the 8 million World Wide Web sites extant in early 2000. It has also received many industry awards for excellence.

Apache is Open-Source software: anyone interested can download and have free access to program source code.[1] Given access to source code,

[1] Other well-known examples of open-source software: the Linux computer operating system, the Perl programming language and the Internet e-mail engine called SendMail. Open source software has its roots in the "free software' movement started by Richard Stallman in the early 1980s. Stallman founded the Free Software Foundation (FSF) as a means to counter the trend towards proprietary development of software packages, and the release of software without the underlying source code. The purpose of the foundation was to encourage development of software that would come with source code and be available to users for their own modification. A key feature of FSF based development is a licensing scheme called 'Copyleft.' Under Copyleft, the author of the program has the traditional and legal entitlements of copyright protection along with a license for users to redistribute and change software. The Copyleft license provides unique distribution terms that gives all users the rights to use, modify and

technically-skilled users of a program can easily make changes and improvements to it. In the case of Apache, this freedom has been exercised by many users and also by programmers working for companies such as IBM and C2Net, that 'package' and sell Apache software for particular applications. Although additions and improvements to Apache code can be made by anyone, additions to the "approved" version of Apache that can be downloaded from the official Apache website must be passed upon by the Apache Development Group, a committee of volunteers (currently 22 in number) who guide the further development and extension of Apache software.

2.1: The Apache field support system

Apache is a relatively complex software program. One of the functions that somehow must be provided for users of such a complex product is "field support" - provision of assistance to users having difficulties with the program because of defects in the program itself or because of the state of their own understanding. Although such a system is needed, the Apache Development Group has made it very clear that *they* do not want to provide it:

> *"There is no official support for Apache. None of the developers want to be swamped by a flood of trivial questions that can be resolved elsewhere. Bug reports and suggestions should be sent via the bug report page. Other questions should be directed to the comp.infosystems.www.servers.unix or comp.infosystems.www.servers.ms-windows newsgroup (as appropriate for the platform you use), where some of the Apache team lurk, in the company*

---

redistribute the programs code or any program derived from it but only if the distribution terms are unchanged. Thus the code and the freedoms become legally inseparable. The Copyleft concept prevents private hoarding of free software if it was just released under a public domain release (Morin 1993). All users are compelled to leave copies behind for others to benefit. The philosophy of the FSF movement has been recently extended by a number of individuals who are promoting the 'Open Source' concept. These individuals are less concerned about the freeness of "free software" and are instead interested in encouraging software companies to release source code for their products. These individuals believe that companies that release source code, under any type licensing, are inherently preferential to closed and proprietary firms (Raymond 1999).

> *of many other HTTPd gurus who should be able to help."* (Apache Group 1999).

Despite or because of this lack of "official support," a very effective on-line Apache field support system has evolved, operated by and for users themselves. The system takes the form of publicly-accessible "newsgroup" discussion forums carried on the Usenet. An Apache user with a question "posts" it on the appropriate discussion forum. Any interested user can read both the questions and answers that have been posted, and can provide answers or add to the discussion if he or she wishes to do so. Both questions and answers are typically signed and identified by the e-mail address of the person posting.

A question posted on the Usenet initiates a new forum "thread" consisting of a question and associated answer(s). A typical example of such a thread (in this case with one answer only) is as follows:

> Subject: Apache-1.3.1 and FrontPage 98 Extensions. A small problem....
> Information Seeker:
> Hi,
>
> I've compiled and installed Apache-1.3.1 with mod_frontpage.c. That section seems to be working. I have installed the FrontPage98 extensions, and that seems to **almost** be working, but I can't find any relevant information anywhere about how to solve this problem. I can look at a home page for a user, but I can't publish to it. Whenever FrontPage tries to connect to the server, this message appears in the error logs:
> [Thu Oct 8 10: 13:31 1998] [error] (104)Connection reset by peer: Incorrect permissions on webroot "/usr/local/httpd/htdocs/_vti_pvt" and webroot's _vti_pvt directory in FrontPageAlias().
> [Thu Oct 8 10: 13:31 1998] [error] File does not exist:/usr/local/httpd/htdocs/vti_bin/shtml.exe/_vti_rpc
>
> I haven't a clue how to fix it. Any help will be very appreciated, and a reply by e-mail will be noticed more quickly (I'm terrible at remembering to check the newsgroups)
> Thanks!
> Information Provider 1:
> Hi there,
> There are two possible causes of your problem:

> 1: Make sure owner and group are the same and that the directories have the same set of permissions. /home/user/public_html user group /home/user/public_html/_vit_bin www group1 should be: /home/user/public_html user group /public_html/_vit_bin user group
> 2: Apache-fp utilizes fpexe and mod_frontpage to provide a higher level of security. Part of the mod_frontpage code sets LOWEST_VALID_UID and LOWEST_VALID_GID. Users with UID's and GID's less than these values will not be able to run the server extensions. These values are configurable. For more information please check the SERK documentation and the apache-fp page.
> Greetings

Multiple sources of technical help for Apache users exist in addition to the Usenet help forum, ranging from books to online journals to an online collection of answers to frequently asked questions. In order to reduce the volume of questions posted on the Usenet help forum, the Apache Development group urges users who encounter problems with Apache software to perform two tasks before posting a question: (1) read the Apache FAQ (Frequently Asked Questions) and known bugs data bases; (2) Search the Apache Usenet archives for related questions and answers that might solve the user's problem without need for a new Usenet posting. (Although there is no official Apache archive, all questions and answers have been and are being automatically indexed and preserved in Usenet archives by companies like Deja.com and Reference.com. These firms offer access to anyone at no charge.)

3.0: Literature Review: Motivations to contribute to Open Source

Academic exploration of open source software is just beginning. However, the interest and importance of the open source phenomenon has been clear to participants for a number of years, and several have been writing about it for fellow "members of the tribe" and for general observers as well. Notable among these authors is Eric Raymond, architect/manager of the open source software project and program called fetchmail - a widely used email transport program.

Raymond (1999) has written an excellent description and analysis of mechanisms used and incentives at work in open source, and a number of noted open source participants and researchers have attested to the accuracy and insightfulness of his work.

Raymond reports that participants in open source programming efforts have at least three basic motives for writing or contributing to the writing of open source software. First, they may directly benefit from the software and software improvements they develop, because they have a use for them. Second, they may enjoy the work itself. Third, they may gain an enhanced reputation in the eyes of peers from making high-quality contributions to an open source project. Each of these motivations has some support in the general literature. Thus, it has been shown that users do develop important innovations in a number of fields, and that this course of action can "pay" (von Hippel 1988). With respect to enjoyment of the work, the characteristics of tasks that individuals often carry out because they are intrinsically rewarding, such as rock climbing, have been explored by Csikszentmihalyi (1975,1990, 1996). Tasks carried out by participants in open source software projects – writing or debugging software, for example, do fit a number of the characteristics identified by Csikszentmihalyi as associated with intrinsically-rewarding tasks – a level of challenge somewhere between boredom and fear, for example. Finally, the fact that "reputation matters" and that seeking to maintain or enhance it can affect behavior has been explored by many.

Lerner and Tirole (2000) propose possible links between the three incentives just described and extant economic language and ideas. They consider the net benefit that participants may obtain as consisting of immediate payoff (current benefit minus current cost) plus a delayed payoff. Immediate payoffs consist of the programmer's own use of the program improvement developed. Immediate cost consists of the opportunity cost of the time invested by the programmer, with the actual cost of this time depending upon how enjoyable the

programmer finds the task. The delayed payoff consists of a career concern incentive (future job offers etc.) and an ego gratification incentive stemming from a desire for peer recognition. Lerner and Tirole argue that both of these delayed payoff elements can usefully be seen as instances of what the economic literature calls signaling incentives (e.g. Holmstrom 1999). They also observe that "…tasks aiming at helping the much-less-sophisticated end user – e.g., …technical support – usually provide lower signaling incentives." (ibid p. 19).

Kollock (1999) discusses four possible motivations to contribute public goods on-line. Given that his focus is incentives to put on line something that has already been created, his list does not include any direct benefit from developing the thing itself – either the use value or the joy of creating the work product. His list of motives to contribute does include the beneficial effect of enhancements to one's reputation. A second potential motivator he sees is expectations of reciprocity. Both specific and generalized reciprocity can reward providing something of value to another. When information providers do not know each other, as is often the case for participants in open-source software projects, the kind of reciprocity that is relevant is called "generalized" exchange (Ekeh 1974).[2] The third motivator posited by Kollock is that the act of contributing can have a positive effect on contributors' sense of "efficacy" - a sense that they have some effect on the environment (Bandura 1995). Fourth and finally, he notes that contributors may be motivated by their attachment or commitment to a particular open source project or group: In other words the good of the group enters into the utility equation of the individual contributor (ibid p. 228-9).

[2] In "generalized" exchanges, help given to a person is reciprocated by someone else in the group and not by the particular recipient of the original help. Generalized exchange is used to explain why, for example, stranded motorists get helped by strangers: the person helping is expecting that when they are stranded, someone will help them in turn. Kollock notes that "…indeed some observers (Wellman and Gulia 1999, Rheingold 1993) have reported that individuals who regularly offer advice and information seem to receive help more quickly when they ask for something." (ibid p. 227).

Kollock also points out that the kinds and quantities of contributions made on-line will be sensitive to the costs and benefits involved – and he notes that on-line costs for distributing a piece of information can be near zero. "While it may be the case that many people spend time and effort producing goods they intend to contribute to the group, another path to the production of public goods is as a simple side-effect of private behavior. People may need to write a particular computer program for their own use with no thought to anything other than solving their particular problem at hand. Having written the program, the costs of now sharing and distributing it with others may be near zero: they can simply post it in an appropriate discussion group or other online community." (ibid p. 229). More generally, Thorn and Connoly (1987) argue on the basis of theories of the economics of public goods that the rates and effectiveness of discretionary information sharing amongst employees in an organization will tend to decrease as: (1) participation costs increase, (2) the size of the overall group increases, (3) lower value of information to participants and (4) greater asymmetries in information values and benefits across participants.

2.1: Motivations to contribute to Open Source help lines

User participation in the major tasks of open-source software projects – software writing and debugging – may in fact be motivated by personal benefit from the work product, fun of the work and reputation. However, some "necessary but mundane" tasks do not seem to fit this set of motivations very well – at least on the face of it. Thus, providing answers to users on a help line does not obviously involve a work product of immediate value to the information provider. The work of answering questions may be fun to some. Also, some reputational advantage may accrue from answering questions - although Apache interviewees suggest that any such benefit is likely to be slim. (The major reputational benefits to be gained in open source projects go to those who contribute in other ways, such as writing

useful and technically elegant code.) So the question we started with remains: why do some users willingly carry out necessary but mundane tasks such as providing free help to others who pose questions on open-source help lines?

Constant, Sproull and Kiesler (1996) have carried out the only empirical study we are aware of that has some empirical data the motivations reported by participants in a computer "help line" system. The particular system they explored was the Tandem Computers Inc. internal corporate help line implemented upon that firm's internal computer network. Their sample was 55 information seekers and 295 information providers (most questions received several replies). Overall, they found that the system was effective: information seekers did get technical advice that they found useful, with 49% saying that replies received had solved their problem.

Table 1: Information providers reasons for answering questions on a corporate online help line*

Reasons for participating	*Points assigned (mean)*
Personal benefits	
I enjoy helping others	16
I enjoy solving problems	9.5
I enjoy earning respect	4.8
The company rewards information sharing	0.9
Total	**31.2**
Organizational Motivation	
Being a good company citizen	17.8
The problem is important to the company	14.0
It is part of my job to answer questions like this one	12.6
I expect others to help me, so it is only fair to help them	11.8
Total	**56.2**

*Source of data: Constant et al (1996) table 5 p. 129.

To measure information providers motivations, the researchers asked each information provider in their sample to allocated 100 points among eight reasons they might have had for replying to the information seeker, with the results shown

in table 1 above. Of course, participating in an open source software help line is not the same as participating in a corporate one. However, on the face of it, these findings suggest that "being a good company (open source project?) citizen" and executing tasks "important to the company (project?)" may be important motives for participation. Enjoyment of the task of answering a question, "part of my job" and reputational gains ("I enjoy earning respect") also appear, but less strongly.

4.0: Research Methods

The empirical exploration of the Apache help system we report upon here was preceded by a pilot study of Apache help system behavior (Lakhani 1999) and by several interviews held with several individuals who had very good first-hand knowledge of the Apache field support system[3]. The empirical data we collected for study was related to postings to the Apache Usenet help forum, CIWS-U (comp.infosystems.www.servers.unix). CIWS-U is one of two Usenet newsgroups that address questions related to Apache web server software. It was chosen for study because the questions posted to it are predominantly Apache- related – only a few postings deal with questions about other varieties of Unix-based server software.

Two basic types of empirical data were collected regarding postings to this Apache Usenet help site:

- For data regarding long-term participation in CIWS-U – who participated, long-term trends, etc. - we examined website log data from 1996 through 1999. This four-year period spans essentially the entire history of online Apache help (recall that Apache 1.0 was released only in December 1995). Website log data was obtained from a World Wide Web service called Deja.com. This service archives all of the discussion groups on the world wide Usenet and makes available advanced search and parsing capability through

[3] These individuals were: two current members and one emeritus member of the core Apache Group; one significant contributor to Apache, and two individuals who had participated frequently in the Usenet portion of the Apache field support system.

their website (www.deja.com).

- We collected questionnaire data from people who posted either questions or answers to CIWS-U during the 4 ½ months from October 1, 1999 to February 15, 2000. During this time period, we monitored activity on CIWS-U near-continuously via computer. Each time a question or an answer was posted on CIWS-U, our computer automatically detected whether the individual was posting a question (e.g., was starting a new "thread") or was providing information related to a previously-posted question (e.g., was referring to an existing thread in his or her posting). It then sent the proper version of our questionnaire (one appropriate to information seeking or one appropriate to information providing) to the email address of that individual. This "automatic" data collection method had the advantage of allowing us to obtain information from posters on a near real-time basis – while recollections regarding what they did and why they did it was still fresh.

While designing our data collection methods, we sought advice from some Apache Group members regarding presentation and procedure. As finally implemented, each questionnaire was accompanied by a brief letter explaining who we were and what we were trying to do – that is, we were trying to learn about the Apache help system. To minimize intrusion on potential respondents, we did not follow up our initial request with any repeated requests to respond, and we only sent a questionnaire out to any individual once – in response to the first time that individual either posted a question or an answer during our period of data collection. We also provided an email address for anyone who wanted to contact us to complain or comment. (In the end, we received only 6 comments, half favorable and half not.)

The sample size and response rates for this sample are as shown below. The data collection period for this sample included Christmas and New Year's vacations, and response rates during these times was about half of the average level shown.

Table 2: Sample of individuals posting questions or answers on Apache Usenet help site from October 1, 1999 to February 15, 2000

	Total Participants	*Information Seekers*	*Information Providers*
Sample queried	1709	1288	421
Usable responses	336	214	122
Usable responses	20%	16.6	29

An examination of posting histories on CIWS-U during the period 1996-9 showed that some of our information seekers had sought information many more times than the mean for all seekers and that, similarly, some of our providers had provided many more times than the mean for all providers. Preliminary data analyses showed it would be useful to contrast these individuals with more average seekers and providers on a number of variables. Accordingly, we divided our sample of information seekers into two subsamples. "Frequent seekers" were all information seekers who posted 4 or more questions during the period 1996-9 (about the top 10% of our seeker respondents) and who had a ratio of seek to provide posts greater than one. All other seeker respondents were placed into the subsample of "other seeker." Similarly, "frequent providers" were all information providers who posted 10 or more questions during the period 1996-9 (about the top 10% of our provider respondents) and who had a ratio of provide to seek posts greater than one. All other provider respondents were placed into the subsample of "other providers." [4]

5.0: Findings: Nature of participation in the Apache Usenet help forum

[4] The reason for the ratio test was that respondents were sorted into seeker or provider categories according to their role in the *first* (and sometimes only) posting they made in our sampling window of 4.5 months. If analysis of CIWS-U logs showed that they more typically were posting posting messages in the opposite role (e.g., seeker instead of provider) we did not want to include their data in our assessment of "characteristics of seekers vs providers." (We could have gone the next step and shifted them into the category which was their typical role, but elected not to do this. Trial data analyses showed that such category shiftings would affect only a few individuals and would not materially affect our findings. On the negative side, category shiftings would make the analysis more difficult to follow.

Apache version 1.0 was released in December, 1995. The number of websites using Apache has increased dramatically since then, to over 60% of the web server software "market" and over 8 million sites active at the start of 2000 (Netcraft, April, 2000).

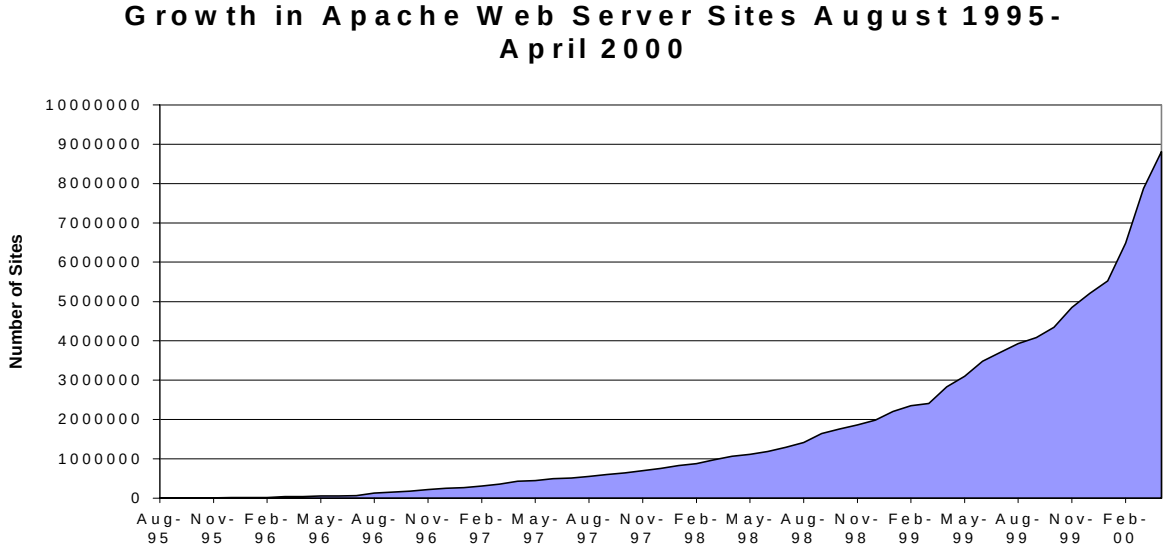

Figure 1: Number of websites using Apache 1995-2000

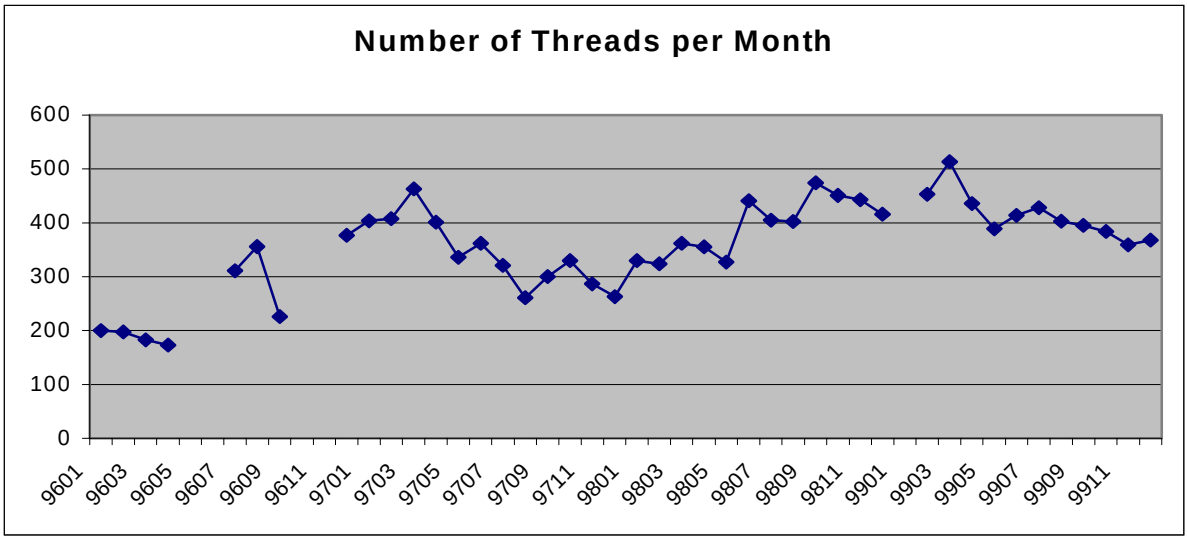

Figure 2: Number of new questions asked per month from 1996-9

The number of new "threads" initiated each month on the Apache help forum (a thread consists of a question plus one or more answers) has also been growing, but not nearly so rapidly. In fact, participation in the Apache Usenet help forum is strikingly small relative to the number of sites (8 million in early 2000 – run by perhaps 800K webmasters) using Apache.

During the four year period (1996-1999), there were 11511 distinct users. Of these, 4902 only posted answers on CIWS-U (information providers), 8981 only posted questions, and only 2372 did both. Information providing was relatively concentrated: approximately 50% of the answers on the system were provided by the 100 most prolific providers (2% of all providers). In contrast, 50% of the questions were provided by the 2152 most prolific posters of questions (24% of all information seekers).

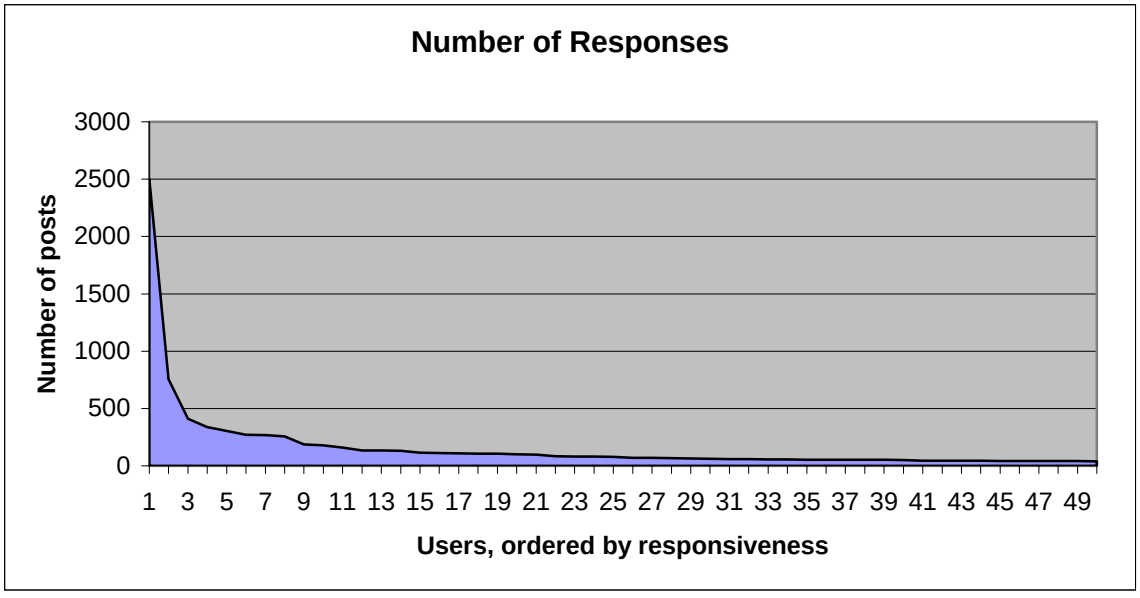

Figure 3: Number of answers provided by the top 50 Apache Usenet help participants from January, 1996 through September, 1999

The 100 most active information seekers posted an average of 10.43 questions and the 100 most active information providers posted an average of

83.63 answers during the four year period 1996-9. Frequent participants also turned out to be long-term participants. We found that mean elapsed time between first and last posts during the 1996-9 period was 674 days for frequent information providers; 168 days for other providers; 661 days for frequent information seekers and 107 days for other information seekers. (These periods of participation should be taken as "equal to or greater than" statements about length of participation, since it is likely that many will continue to post during year 2000 and beyond.)

Table 3: Attributes of respondent information seekers and information providers

Attribute	*Frequent Providers*	*Other Providers*	*Other Seekers*	*Frequent Seekers*
Mean Usenet reading time/session (mins)	12.48	18.09	18.52	17.69
Mean time using web servers (months)	47.71	43.94	29.70	50.31*
Mean Apache experience (months)	33.86	31.99	21.13	41.54*
Mean work time dedicated to web server operations	51.19	36.85	29.38	24.39
Scale of my web site – (millions of hits/day)	4.89***	4.08	3.20	4.14***
I have modified Apache source code	81%	46%	22%	31%
My website is for professional purposes	48%	60%	47%	69%
Mean total posts as information seeker over 4 years (1996-9)	3.81***	1.44	1.71	4.77*
Mean total posts as information provider over 4 years (1996-9)	169.29	2.53	1.98	2.08**

Information seekers differed from information providers on a number of attributes. In general, frequent information providers and frequent seekers as well appear to be more expert than "other" information seekers or providers, having on

average have more months of experience with Apache, and with web servers in general. Frequent providers are much more likely to modify the Apache source code (81% have done this) than are other posters to Apache Usenet help (table 3).

6.0: Findings: Effectiveness of the Apache help forum

Users rank Apache technical support overall as somewhat better than that of its major commercial rivals in the server software field. Thus, participants in the Serverwatch internet poll rank Apache 4.5 out of 5 with respect to technical support, Netscape 4, and Microsoft IIS 4 (Serverwatch.internet.com, 1999). This general endorsement may or may not apply to Apache on-line help specifically however: Apache technical support has a number of elements and, as our information-seekers attest, many are used.

Table 4: Additional Apache help resources used by individuals posting questions on CIWS-U

Apache resource used	*Frequent Seekers*		*Other Seekers*	
	% Using	Mean Time (min)	% Using	Mean Time (min)
APACHE FAQ	69	13.3	79	39.3
Usenet Archives	77	23.5	78	30.2
Other online resources*	69	18.8	40	38.4
Books on Apache	54	65.8	39	140
Known bug data base	69	2.5	32	13.6

* For example, online “journals” such as Apache week.com and RTF.com.

Questions posted by information seekers varied in nature (table 5), and only 8.9% said that the problem they posted online was extremely critical and that they needed an answer right away.

Table 5: Nature of questions posted on Usenet by information seekers

Type of problem asked about	*Frequent Seekers*		*Other Seekers*	
	n	%	n	%
Complete Down	-	-	5	2.6
Functional - missing important features	4	33.3	48	24.7
Functional - missing optional features	5	41.6	109	56.2
Installations problems	2	16.6	26	13.4
Upgrade problems	1	8.3	6	3.1
Total n	12	100	206	100

Data collected on response times from 1996-99 website logs and also from our "real-time" sample showed that initial answers to publicly-posted questions generally came quite quickly - at least 50% were answered on the day of or on the day after posting (table 6).

Table 6: Response to questions posted on Usenet

Sample	*Got public reply Same day*	*Got public reply next day*	*Got public reply 2 days +*	*Got private email reply only*	*No reply received*	*(n)*
1996-9 Usenet log data	32%	17%	12%	NA	39% no *public* reply	12964
4.5 month real-time sample	34%	18%	9%	16%	23% no public *or* private reply	1288

As can be seen from table 6, 39% of information seekers received no public reply (that is, a reply posted for all to read on Usenet) to their Usenet posting (true for both samples in table 6). However, 40% of the respondents to our 4.5 month real time sample who received no public reply to their query reported receiving one or more replies that were sent privately via email instead. If this ratio holds for the

historical data as well, then only about ¼ of questions posted on Usenet do not receive an answer. (Lakhani (1999) compared the content of a sample of messages that did receive replies with a sample that did not, and found no obvious differences with respect to clarity, completeness or technical difficulty.)

7.0: Findings: costs and benefits of participating on Apache help forum

To successfully complete an information transaction on the Apache Usenet help forum, three tasks must be completed: (1) a question must be posted; (2) the information sought must be matched to an appropriate and willing provider of information; (3) an answer must be provided. Obviously, the burden of question-asking must be placed upon the information seeker, and the burden of information provision (both the time associated with providing it and any losses associated with sharing proprietary information) on the information provider. However, the burden of seeker and provider match-up varies according to the design of the information system. For example, in the case of an encyclopedia or a FAQ data base (a list of answers to Frequently Asked Questions), the burden of match-up is placed upon the information seeker.

7.1: Costs and benefits of question and answer matching

In the case of the Apache help Usenet forum, the burden of matching up an information seeker and an information provider is placed on the information provider. Potential information providers find questions that they can and are willing to answer by simply reading or scanning the questions posted on the Apache help forum. In order to understand the extent of the match-up burden placed upon information providers, we asked our respondents about the time they spent reading CIWS-U. We and found that annual time spent, especially by information providers, is typically quite substantial (table 7).

Table 7: Respondents' Usenet reading pattern

*How frequently do you read Apache Usenet? Time expended?**	*Frequent Provider (n=21)*	*Other Providers (n=68)*	*Other Seekers (n=195)*	*Frequent Seekers (n=13)*
Daily	76%	32.4%	11.3%	23%
Weekly	24%	42.6%	22.1%	30%
Monthly	-	7.4%	8.2%	-
Only when problem	-	17.6%	58.5%	47%
Mean annual reading volume (mins)*	4774	2774	1838	1816

* annual reading volume was calculated by multiplying number of reading sessions reported times average length of session reported

If information providers incurred the substantial time expenditures devoted to reading CIWS-U only to identify questions they were able and willing to answer, they would indeed be spending heavily to help information seekers. But information providers (and seekers) report that the most important reason they read CIWS-U is to learn: they gain valuable information from reading about problems other users are encountering, and how these might be solved. In contrast, only frequent information providers show some agreement with the statement that they read the Apache help forum in order to answer questions.

Table 8 : Respondents' reasons for reading Usenet

(7-point scale: 1 = strongly disagree, 7 = strongly agree)

Reasons for reading	*Frequent Providers (n=21)*	*Other Providers (n=68)*	*Other Seekers*	*Frequent Seekers (n=13)*
To learn	5.90	5.75	5.29 (n=191)	6.38*
To answer	4.95*	4.00	3.72 (n=189)	4.167
For fun	4.29	3.97	2.90 (n=190)	3.46
For break	4.81**	3.99	2.66 (n=188)	2.69

* Significant difference at p <0.01
** Significant difference at p <0.05

7.2 Costs and benefits of question posting

Members of the Apache community are very familiar with Usenet procedures. As a consequence, cost to information seekers posting a question to the Apache Usenet help site consists only of their time expenditure to prepare and post that question. Seekers report preparation and posting time to be a mean of 11.5 minutes (n = 212).

Benefits to seekers consist of the problem-solving time saved due to answers received to their posted question. As can be seen from table 9, a majority of both frequent seekers and other seekers who received replied to their questions judged the information contained in those replies to be useful. (Respondents who received both public and private replies generally judged both to be of equal value: 14% judged the private replies to be of higher value and 21% judged the public replies to be of higher value (n = 188).)

Table 9: Information seekers' evaluation of the answers that they received to the question they posted on Apache Usenet

What was the value to you of the answers you received?	*Frequent Seekers*	*Other Seekers*
Solved my problem completely	23% (n=3)	17% (n=34)
Gave me information that helped solve my problem	69% (n=9)	44% (n=87)
Did not solve my problem	8% (n=1)	39% (n=77)

Seekers who received answers to their questions estimate the problem-solving time they saved due to answers received to their questions at a mean of 381 minutes (194 seekers responding). Thus, the mean net time benefit information seekers receive from posting a question on CIWS-U is 370 minutes. Or, to put it another way, the benefit cost ratio experienced by information seekers who post a question is about 35 – quite a good return on investment!

7.3: Costs and benefits of information providing

In the Apache system, as we noted earlier, the cost of question and answer match-up falls upon the information provider. However, providers accomplish the match-up task by reading or scanning questions posted on Usenet. And, as responses in table 8 showed, providers do this primarily in order to learn, rather than to answer questions. Given this finding, we reason that the task of question and answer match-up in the Apache Usenet system is effectively achieved as a costless side-effect of an activity undertaken for another reason by potential information providers. We therefore think it is reasonable to leave aside the cost of question and answer match-up in assessing the net benefit of posting to CIWS-U for information providers.

Leaving match-up costs aside, costs incurred by an information provider who answers a question on Usenet involves two elements: (1) value of proprietary information that may be lost when that information is publicly posted on the Apache Usenet forum, and (2) the costs and benefits associated with generating and posting an answer to a posted question. We assess each of these elements in turn.

Information held by information providers loses any proprietary value it might have had (unless it is protected by patent – a very unlikely circumstance) if it is publicly posted to the Apache help forum. However, *if* potential providers think that others know the same information and *if* they think those others will provide it if they do not, providers should assess the loss of intellectual property value associated with their choosing to answer a question at zero. (Indeed, under these conditions, a provider's best strategy may be to strive to be the first to reveal the information sought in order to reap any associated reputational advantages.)

On the basis of this reasoning, we asked the information providers in our sample "how many other readers of CIWS-U do you think also knew a solution?" to the question they had answered on the Apache forum. As can be seen below, all providers reported that they did think that some or many other readers also knew a

solution and so could potentially furnish an answer.[5]

Table 10: How many others do you think know the solution to the question you answered on Usenet?

How many others do you think knew the answer to the question you answered?	*Frequent Providers*	*Other Providers*
Many	38% (n=8)	61% (n=41)
A few with good Apache knowledge	38% (n=8)	18% (n=12)
A few with specific problem experience	24% (n=5)	21% (n=14)
No others	NA (see fn. 5)	NA (see fn. 5)

Information providers potentially concerned about losses of valuable proprietary information incurred by answering a question posted on the Apache help forum have no logical reason to be concerned – if and as they think that others holding the same information would answer if they did not. We did not ask providers whether they in fact held this view. We did, however, ask a related question: "I answered the question because I thought the poster might not get a good answer if I did not." On a scale of 1-7 with 1 being strongly disagree and 4 being neutral, frequent information providers expressed a moderate level of agreement (a mean of 4.52) with this statement (c.f. table 14 reason #12).[6] This suggests that at least these information providers are not viewing answer-provision in terms of potential loss of value of proprietary information – whether or not they "should."

We next consider the costs and benefits associated with generating and posting an answer to a question posted on the Apache help forum. An important finding here is that the cost of carrying out this task is typically quite low. About

[5] NB, the level of this response is to some unknown degree inflated: the authors neglected to include an explicit response option of "no others" for this question on our questionnaire, and so the only way that a respondent could even indicate such a view was by not indicating agreement with any of the options presented - which none did.

[6] The level of agreement with the question (on a scale of 1-7, with 1 = strongly disagree) was: 1 = 10, 2 = 5, 3 = 20, 4 = 40, 5 = 24, 6 = 14, 7 = 9. Total n = 122.

half of frequent information providers spent 1 minute or less answering a question on Usenet, and 87% of other providers spent 5 minutes or less at this task.

Table 11: How long did it take you to answer the posted question?

Time spent to answer	*Frequent Providers*	*Other Providers*
1 minute or less	48% (n=10)	19% (n=13)
1 to 2 minutes	29% (n=6)	21% (n=14)
2 to 5 minutes	19% (n=4)	40%(n=27)
5 to 10 minutes	-	16% n(=11)
More than 10 minutes	4%(n=1)	4% (n=3)

As we can see from table 12, this small time-expenditure was possible because providers generally already knew the answer to the posted question.

Table 12: What was your state of knowledge when you first looked at the question you answered?

State of knowledge	*Frequent Providers*	*Other Providers*
Already knew solution	76% (n=16)	64% (n=44)
Knew where to find the solution	5% (n=1)	2% (=1)
Some useful information but not solution	19% (n=4)	28% (n=19)
No solution but had ability to solve	-	6% (n=4)

Providers were asked whether they knew the answer because of their general knowledge of Apache (32%, n = 38), or because they had experienced the same problem themselves (68%, n = 82). When information providers knew the answer due to their general expertise in Apache, their mean time expenditure was significantly shorter (3.2 minutes) than when they knew the answer because they had experienced the problem themselves (5.5 minutes mean time expenditure) ($p < 0.013$).

Whatever their state of knowledge at the time information providers saw the posted question, they typically *only* provided information they already had in hand

(table 13).

Table 13: What did you do to answer the question

Activity undertaken	*Frequent Providers*	*Other Providers*
Provided information I already had	90% (n=19)	82% (n=56)
Searched for additional information	10% (n=2)	15% (n=10)
Engaged in problem solving	-	3% (n=2)

On average, information providers who only provided information they already had expended 4.0 minutes to provide an answer. Providers who either searched for more information or engaged in problem solving before answering expended 9.33 minutes to respond. This difference is significant at the 0.05 level (p=0.022).

To this point we have found that the costs incurred by information providers to answer a question on Apache Usenet are typically quite small: Frequent providers typically take 2 minutes or less to generate and post an answer, and ordinary providers spend 5 minutes or less to do this. We next turn to consider the benefits potentially flowing to information providers from investing this small amount of time to answer a question posted on the Apache help forum. As was discussed in our review of the literature (section 3) several types of benefit may be motivating information providers to respond:

- *I expect reciprocity* (questions # 1-3 in table --). Both specific and generalized reciprocity can reward providing something of value to another. Since, as we will see shortly, the information providers did not know information seekers before providing help, the most relevant source of literature is that on "generalized" exchange (Ekeh 1974). In such exchanges, help given to a person is reciprocated by someone else in the group and not by the particular recipient of the original help. Generalized exchange is used to explain why, for example, stranded motorists get helped by strangers: the person helping is expecting that when they are stranded, someone will help them in turn (Kollock 1999).

- *I am "helping the cause"* (question #4). Individuals involved in open source software projects often strongly identify themselves as belonging to a community (Raymond 1999). Constant et al. (1996) demonstrated that people

who have a strong attachment to an organization will be more likely to assist other with organization related problems. It has also been argued that people who develop a strong attachment to a virtual group are more likely to participate and provide assistance to others (Wellman and Gulia 1999).

- *I will gain reputation* (questions 5 & 10). Information providers sign the answers they post, and may gain in reputation by answering frequently or well. Gains in reputation can be rewarding in and of itself, and may also lead to benefits such as enhanced career prospects.[7] A number of researchers have argued that gaining a reputation within a community, including a online community, is an important incentive for active participation. (Reheingold 1993, Raymond 1997, Constant et al 1996, Lerner and Tirole 2000).

- *Answering questions is intrinsically rewarding* (questions # 6-9). Interviewees with expertise in Apache suggested to us that intrinsic rewards – induced feelings of competence, fun, or being rewarded by "taking a break" were important motivators for answering questions. This view finds support in the research of Csikszentmihalyi, who has explored the characteristics of activities individuals engage in because they offer the intrinsically rewarding experience of "flow." Answering questions on the Apache help forum does appear to fit a number of the characteristics of "microflow" activities that have been found to be intrinsically rewarding (Csikszentmihalyi 1975,1990, 1996).

- *It's part of my job* (question 11). Several companies are now selling commercial versions of Apache software. Typically this entails offering a packaged distribution of Apache, plus documentation and support. It is possible that such companies might assign people to answer questions posted on the Apache help forum as part of their job responsibilities.

We asked information providers to express their agreement or disagreement

[7] Some Apache help forum users we interviewed suggested that an "Alpha-Male" variant of reputation building behaviours might be visible among information providers. Some providers, they said, wanted to be known as "the" expert in a particular aspect of Apache. To build and preserve such a reputation, these providers would strive to quickly answer all questions associated with "their" area. They would also seek to drive out other providers who offered answers in that area by quickly posting comments on the answers provided by those others in a way that, while outwardly cooperative, would also indicate their own technical superiority and prowess in the particular area. In other words, such a person acted like an "Alpha-Male" by attempting to drive out all other information providers from his chosen field of expertise. We saw no evidence of such behaviour in our small sample - in the sense that we saw no clustering of answers by subject area.

regarding each of these possible motivations, with the results shown in table 14.

Table 14: Providers' views regarding their motives for providing answers to help seekers on Apache Usenet

(7 point scale, neutral = 4, strongly agree = 7)

I was motivated to answer because:	*Frequent Providers*	*Other Providers*
(1) I help now so I will be helped in the future	4.52***	5.15
(2) I have been helped before in CIWSU – so I reciprocate	4.85	5.14
(3) I have been helped on Usenet before – so I reciprocate	4.61	5.16
(4) I answer to enhance my career prospects	3.76	3.57
(5) I want to enhance my reputation in OSS/Apache community	4.71	4.57
(6) I answer because its fun	4.81	4.38
(7) I answer to promote OSS	5.14	4.76
(8) I answer to take a break	4.65	4.22
(9) I answer because it is part of my job	2.23	2.52
(10) I have expertise in this area	4.47	3.92
(11) I am the authority in this area	2.47	2.01
(12) I answered because I thought the poster would not get a good answer if I did not	4.52	4.08

*** Significant difference at p <0.10

In general, we can see that providers were in moderate agreement with most of the motivations listed in table 14. The statement that "it is part of my job" was quite strongly disagreed with by most, which indicates that helping was indeed discretionary for respondents. Top providers differed in expectable ways from other providers, for example they felt that they had more expertise.

All self reporting regarding motivations must be viewed with caution: respondents may be inclined to emphasize the "right" socially correct or conventional motivations (Drake, Finkelstein et al. 1982). This concern is reinforced for us by an apparent contradiction between stated motives and related evidence with respect to reciprocity.

In table 14, the most agreed-with statements include the three statements having to do with reciprocity "I help because I have been helped and/or expect to be helped (statements 1-3 in table 14)[8]. Information seekers do show a higher level of agreement than do information providers, but the level of agreement shown by providers is hard to square with rational expectations of specific, tit-for-tat reciprocity behaviors: 96.7% (n = 116) of the information providers reported that they did not know the individual they were helping. Also, it is unlikely that generalized reciprocity was at work here. Recall that seekers and providers had different characteristics. Recall also that, of the CIWS-U posters in the period 1996-9, 57% sought information only, 22% provided information only, and only 21% did both (posting an average of 2.50 questions and 7.95 answers). Possibly respondents are really saying that they feel reciprocity is involved because they have gained by learning from reading the questions and answers posted by others on Usenet, and can reciprocate by answering questions. Or possibly, this self-report data is simply unreliable.

8.0: Discussion

In this research we have explored provision of a "necessary but mundane" task – provision of online technical support - by and for users of Apache open source software. What we have found has implications both in the very specific context of provision of technical support for Apache and other open-source software, and the more general arena of the design of innovation systems by and for users that are independent of manufacturer involvement.

[8] The exact text of each of these questions was: (1) "Others have helped me in the past on other Usenet groups and I feel an obligation to reciprocate by answering questions on Apache Usenet;" (2) "Others have helped me in the past on CIWS-U and I feel an obligation to reciprocate by answering questions on Apache Usenet." (3) "If I answer a question on CIWS-U others are more likely to help me when I post a question in the future."

8.1: Implications for the provision of Apache online help

Our empirical study has revealed an online help system that does work quite well for those who participate. Most questions posted are answered quickly and most answers received are judged to be valuable by information seekers. However, is used by and for a relative handful of people given the now very large number of sites and webmasters using the software (in early 2000, 8 million sites run by perhaps 800K webmasters). We also find that the system relies heavily on around 100 information providers who in aggregate post 50% of the messages, with the very top few frequent information providers – notably Mark Slemko, an Apache Group member who has posted answers to about 2,500 questions in 4 years – answering a very large number of questions each. This finding may or may not imply lack of robustness in the system as a whole. After all, most elements of open source depend on the contributions of a relative handful – certainly the writing of Apache does.

It is not clear where non-participants get solutions to questions they may have. Perhaps the many available books and other resources, such as the Apache FAQ data base, solve the problems of the vast majority of users who therefore do not need online help. Or, perhaps the vast majority of users find using online help intimidating – we just do not know at this stage. However, we reason that the present design of Apache Usenet would need modification if question volume rose, say 100X. (Currently, about 400 questions are being asked per month (figure 2). An increase to 40,000 per month would mean that the 800,000 extant webmasters would ask questions at a rate of only about one-half question per year each – not an unreasonable figure on the face of it.)

Recall that successful completion of an information transaction on Apache Usenet help requires completion of three basic tasks: (1) a question must be posed; (2) the information sought must be matched to an appropriate and willing

provider of information; (3) an answer must be provided. In the case of the Apache Usenet help system, the burden of matching up an information seeker and an information provider and the actual provision of an answer has been placed on the information provider: each potential information provider finds questions he or she can and will answer by reading or scanning questions that have been posted on Apache Usenet help, and then posts an answer.

Among possible system designs that place the burden of question and answer match-up on the provider, the Apache design is a high-cost variant. Most such systems use one or more layers of relatively inexpert intermediaries as filtering and matching facilitators. For example, in the provision of medical services, general practitioners are often used as the gateway to select which of many possible expert providers are appropriate for a particular patient's medical problem. Similarly, many telephone-based "help" lines provided by manufacturers use generalists to screen a direct questions to the proper expert information provider. In contrast, the Apache system involves question and answer match-up directly by expert providers "in person."

The Apache approach has great benefits for information seekers. First, it saves questioners the time and trouble of dealing with intermediaries. Second, each participating expert information provider sees each question in an unfiltered form, and so may recognize a problem that would not be apparent to a less expert intermediary. On the other hand, the Apache approach has a cost for information providers: multiple experts expend the time to read many questions that they are not able or willing to answer. It seems reasonable that an increase in question volume of 100x or more would swamp the ability and willingness of multiple experts to scan the entire list of incoming questions.

As a separate matter, information providers on the Apache Usenet help system have been and are willing to subsidize the match-up task because they learn by scanning the questions and answers posted by others. Over time, this benefit

may decrease if and as there is "less to learn." This may happen if the rate of change in the environment faced by Apache or the rate of change in Apache itself decreases. A comment by Eric Raymond on his experience with help from users of his open source program, fetchmail, is suggestive in this regard.

> "Actually… the list [of fetchmail beta-testers] is beginning to lose members from its high of close to 300 for an interesting reason. Several people have asked me to unsubscribe them because fetchmail is working so well for them that they no longer need to see the list traffic! Perhaps this is part of the normal life-cycle of a mature, bazaar-style project." (Raymond 1999, p. 46-7)

If the present model of Apache help gets less effective because of an increase in "non-subsidized" match-up costs for information providers, there are alternatives that can be developed and offered to get the very important help task done. It is clear that, under the current system, the benefit to cost ratio of information seekers is very favorable – currently they save 35X more time than they expend. This suggests that some system changes that partially or fully shift the match-up burden from providers to seekers might be acceptable. For example, a partial shift could be made by the introduction of a filter that screened incoming questions and only forwarded those to each provider that matched that provider's expressed areas of interest. And/or, the system could gradually and seamlessly switch over to a system that completely shifts the costs of question and answer match-up to information seekers by an increased use of (improved?) FAQ and online help question and answer archives if and as provider willingness to respond to new posted questions declines.

8.2: <u>General implications for user-to-user innovation systems</u>

A general concern about systems in which users innovate rather than suppliers is that suppliers have a natural incentive to diffuse their findings – they

make more money if they do - while users may have no such incentive. If users who innovate do not share what they know, the implication for user-only innovation systems is that any other user facing the same problem must invest in developing a solution anew. As the number of users that must duplicate answers goes up, clearly the system-level efficiency of a solution system involving only users goes down. An important finding in this study is that at least some users in the Apache user community who had information sought by another user were willing to publicly reveal that information to the inquirer – and at the same time to all users.

We have seen that some users are demonstrably willing to share solutions they have generated – because they do in fact publicly post answers to questions posted by other users. We have also argued that this willingness makes sense if they think that their information has no proprietary value *or* if they think that other users have the same information and may reveal it if they themselves do not do so. At least the latter belief appears to prevail among the respondents in our sample.

As was noted by Kollock (1999) and by Thorn and Connoly (1987), the willingness of information providers to contribute what they know as a public good should be affected by the cost to them of doing so. In our study, we found that the time-cost to post information on the internet to all potentially interested parties appeared low on the face of it: typically 1-5 minutes. Note that most users answered at this low cost because they only posted information they already knew "off the shelf" – they did no new problem-solving or searching in order to provide additional help the poster. This low time investment by helpers matches findings by Constant et al (1996). In their study of the Tandem Computers corporate help line, they found that the average time devoted to posting an answer to a question was 9 minutes (ibid p. 124).

Of course, low-cost provision of "off-the-shelf" solutions will only work for user communities in which some users *do* know the solution to problems posed by

other users, and when the proper solution can be identified and transmitted at low cost. It is an interesting finding of this study that this appears to typically be the case for Apache. These conditions may not hold for all problem types and user communities. Thus, in some communities the problems encountered by some users may be unique to them and no off-the-shelf solution may exist. Or, even if a solution does exist in the user community, a problem may not lend itself to a clear-enough description to allow a remotely-located expert to match up problem and solution at a low cost. For example, consider that there are some problems in fields ranging from machine diagnosis to medical diagnosis where experts find they must physically go to the problem site to make first-hand observations before they can understand the problem well enough to offer an appropriate solution (von Hippel 1994, Tyre and von Hippel, 1997).

8.3: Suggestions for further research

The study of user-only help systems presented here helps us to understand elements of open source software development systems characterized by an apparently low incentive to contribute. At the same time, it presents a puzzle specific to user to user help systems that seems worthy of further research. The Apache Usenet help system currently handles the volume of questions presented to it quite well. At the same time, the volume of questions posed to it by users seems remarkably low. Does Apache server software for some reason not generate many questions among users? Are Apache users too sophisticated to need help? Are questions going elsewhere? Answers to these questions would aid the development and evolution of the user help element of user-to-user innovation systems.

More generally, it would be useful to conduct similar empirical studies that would highlight other puzzling aspects of how a "user only" innovation system functions. For example, how is coordination achieved among open source software contributors; how can problems be segmented into module of a size that fit the sources and incentives of individual users to effectively contribute. Collectively, we think that such studies will contribute to enabling improved designs for innovation systems "for and by users" – no manufacturer required!

Bibliography

Apache Group (1999). Apache FAQ : http://www.apache.org/docs/misc/FAQ.html#support.

Behlendorf, B. (1999). Personal communication (Brian Behlendorf is a co-founder of the Apache Group.)

Brown, J. J. and P. H. Reingen (1987). "Social Ties and Word-of-Mouth Referral Behavior." Journal of Consumer Research **14**(3): 350-362.

Constant, David, Lee Sproull and Sara Kiesler, (1996) "The Kindness of Strangers: The Usefulness of Electronic Weak Ties for Technical Advice," Organization Science Vol 7, No 2, March-April.

Csikszentmihalyi, M. (1975). Beyond Boredom and Anxiety: The Experience of Play in Work and Games. San Francisco, Jossey-Bass, Inc.

Csikszentmihalyi, M. (1990). Flow: The Psychology of Optimal Experience. New York, Harper and Row.

Csikszentmihalyi, M. (1996). Creativity: Flow and the Psychology of Discovery and Invention. New York, HarperCollins.

Drake, A. W., S. N. Finkelstein, et al. (1982). The American Blood Supply. Cambridge, MA, The MIT Press.

Ekeh, P. P. (1974). Social Exchange Theory: The Two Traditions. Cambridge, MA, Harvard University Press.

Fielding, R. (1999). "Shared Leadership in the Apache Project." Communications of the ACM **42**(4): 42-43.

Granovetter, M. (1973). "The Strength of Weak Ties." American Journal of Sociology(78): 1360-1380.

Granovetter, M. (1982). The Strength of Weak Ties: A Network Theory Revisited. Social Structure and Network Analysis. P. Marsden and N. Lin. New York, John Wiley and Sons**:** 105-130.

Harrison, M. (1995). The Usenet Handbook: A User's Guide to NetNews. Sebastapol, CA, O'Reilly.

Kollock, P. (1999). The Economies of Online Cooperation: Gifts and Public Goods in Cyberspace. Chapter 7, Communities in Cyberspace, M. A. Smith and P. Kollock eds. London, Routledge.

Lakhani, K.R. (1999). "Sustaining the Virtual Commons: End User Support for Apache Web Server Software on the Usenet." Unpublished master's thesis, Massachusetts Institute of

Technology, Cambridge, Massachusetts.

Lerner, Josh and Jean Tirole (2000) “The Simple Economics of Open Source,” NBER working paper 7600, www.nber.org/papers/w7600 (March).

Morin, R. (1993). “The meaning of freeware.” Unix Review **11**(12): 101-102.

Netcraft (2000). April 2000 Netcraft Web Server Survey. N. http://www.netcraft.co.uk/Survey/.

Raymond, E. (1999). The Cathedral and the Bazzar. O’Reilly, Sebastapol, CA

Reheingold, H. (1993). The Virtual Community: Homesteading on the Electronic Frontier. New York, Adison-Wesley.

Schrader, S. (1991). “Informal Technology Transfer Between Firms: Cooperation Through Information Trading.” Research Policy **20**(2): 153-170.

Smith, M. A. (1999). Invisible crowds in cyberspace. Communities in Cyberspace. M. A. Smith and P. Kollock. New York, Routledge.

Stevenson, W. B. and M. C. Gilly (1991). “Information Processing and Problem Solving: The Migration of Problems Through Formal Positions and Networks of Ties.” Academy of Management Journal **34**(4): 918-928.

Tyre, M. and E. von Hippel (1997) "The Situated Nature of Adaptive Learning in Organizations" Organization Science , vol 8, No 1 (January-February) p. 71-83

von Hippel, E. (1994) "Sticky Information" and the Locus of Problem Solving: Implications for Innovation" Management Science 40, no.4 (April): 429-439

von Hippel, E. (1988) Sources of Innovation, New York, Oxford University Press.

von Hippel, E. (1998) “Economics of Product Development by Users: The Impact of “Sticky” Local Information” Management Science, vol 44, No. 5 (May) p. 629-644.

Wellman, B. and M. Gulia (1999). Virtual communities as communities. Communities in Cyberspace. M. A. Smith and P. Kollock. New York, Routledge.