Communication Networks in an Open Source Software Project

Jeffrey Roberts[1], Il-Horn Hann[2], Sandra Slaughter[1]
1 Tepper School of Business, Carnegie Mellon University
Pittsburgh, PA{jroberts, sandras}@andrew.cmu.edu
2 Marshall School of Business, University of Southern California
Los Angeles, CA, hann@marshall.usc.edu

Abstract. This study explores the nature of the social network and the patterns of communication that exist in an open source software development project, the Apache HTTP (WEB) server project. Our analysis of archival data on email communications between developers in the Apache HTTP server project suggests an interesting pattern of communication. We find that the core developers self-organize into three sub-groups that communicate intensely in completing the project. Our analysis also reveals that a few prominent developers who are centrally located in the network are driving communications within the project. We identify the implications of our findings and suggest areas for further research.

1 Introduction

Open source software (OSS) development, i.e., public software development projects where participants can read, modify, and redistribute the software source code [1] is arguably one of the most exciting phenomena in the software industry today. Open source has played a fundamental role in the development of the Internet by contributing to such remarkable software as TCP/IP, BIND, Sendmail, Linux, and the Apache WEB server. From a software engineering perspective, the open source community has harnessed the Internet like no other by making it the critical piece of its communication and collaboration infrastructure. This prima facie simple innovation has resulted in a revolutionary organization of software production and has sparked discussion on a wide variety of issues, ranging from project organization, software development methodology, information architecture, and standards to incentives and intellectual property rights. The open source movement has also been of great interest for academics. Researchers with diverse backgrounds such as computer science, psychology, sociology, and economics have started to investigate the topic, making open source development a truly interdisciplinary research field.

The first works in this rapidly developing field were descriptive in nature [e.g., 2] followed by theory driven explanations [e.g., 3] and early empirical research [e.g., 4–7]. Many of the early explorations into the inner workings of the open source development process have sought to explain the mechanisms by which open source projects attract and motivate volunteers to produce such seemingly high quality software [e.g., 2, 8]. One aspect, however, of the OSS phenomenon that has received

Please use the following format when citing this chapter:

Roberts, J., Hann, I.-H., and Slaughter, S., 2006, in IFIP International Federation for Information Processing, Volume 203, Open Source Systems, eds. Damiani, E., Fitzgerald, B., Scacchi, W., Scotto, M., Succi, G., (Boston: Springer), pp. 297-306

relatively little attention is the nature of the project communication in open source projects.

We are specifically interested in advancing the understanding of project communication and its role in managing the process of creating open source software. How open source developers communicate and interact is an interesting and important question given the geographic distribution of the developers and the unstructured process of software development in the open source context (compared to software development in a closed source setting). This study utilizes archival data to explore the nature of the social network and the patterns of communication that exist in one OSS project, the Apache HTTP (WEB) server.

2 Communication and Social Networks in OSS Projects

In his seminal work on embeddedness, Granovetter [9] outlines how the structural properties of social networks can be significant in explicating outcomes. Researchers have linked an individual's position within social networks to advantages such as promotions [10] or to disadvantages such as turnover [11]. From an embeddedness perspective, social interaction plays an essential role in one's ability to access organizational resources and hence impact one's performance [12]. OSS projects exist largely to perform a specific task or goal, like building an operating system (Linux), a WEB server (Apache) or WEB browser (Mozilla Firefox). The success of an individual within an OSS project requires significant project specific knowledge and/or access to others who may possess information required for success. The "knowledgeable" individual may be especially important in an OSS project as many customary artifacts and processes of software engineering, such as design documentation and methodologies, are typically non-existent [2].

To observe or measure this knowledgeable individual within an OSS project we use the network measure of centrality. In the context of an OSS project's communication network, centrality refers the relative prominence of a developer in the project's network structure [13]. In this case, the degree centrality of a developer measures the number of other developers to which that developer is in contact. So, degree centrality can be taken as a measure of a developer's involvement or participation in the project's communication network [14].

Recent advances in communication technologies and the Internet have greatly improved the ability of individuals to collaborate across time and geographical distance. There can be little doubt that these advances are responsible for the explosive growth in OSS projects, both in terms of numbers of projects and participants [15]. One prominent form of communication is email. In a recent on-line article, Bezroukov [16] compares the collaboration among OSS developers to that of academic researchers. One key observation made in this work is crucial role that email plays in OSS project management. In contrast, researchers exploring the role of email in scientific collaborations have found the email alone does not stimulate new relationships; rather, it serves to enhance existing relationships [17, 18]. Thus, an interesting and unresolved question is how email-based communication is conducted in an open source setting and the relationship between project communication characteristics (or patterns) and project processes and/or outcomes. This question is important because the developers in OSS projects are distributed, and email is the primary communication mechanism available for coordinating their work.

3 Research Setting

To evaluate the social and communication network in an open source context, we targeted one project from the Apache Software Foundation (ASF) as the basis for empirical investigations. The Apache HTTP (WEB) server and associated projects are some of the most successful OSS products to date. The Apache server, the original ASF project, and its derivatives, have a dominant 70% share of the WEB server market [19]. Since its inception, the Apache WEB server has had over 7,000 source code contributions from more than 400 different open source developers [20]. The ASF is a not-for-profit corporation that provides the legal, organizational and financial infrastructure for the software projects gathered under the ASF open-source umbrella. Each of the ASF projects operates autonomously controlling all aspects of product development including project management, requirements specification, architecture, design, development, testing, and configuration management. ASF projects are characterized by a "collaborative, consensus based development process, an open and pragmatic software license, and a desire to create high quality software that leads the way in its field" [21]. Membership in the ASF is by invitation only and is based on a strict meritocracy. Those contributors who exhibit a commitment to the ideals of open-source software development and sustained participation may be nominated for membership by another ASF member.

The ASF encompasses a significant number of subprojects related to the development and support of a full-featured WEB server product offering. Although any of the Apache subprojects might provide an interesting vehicle to explore communication patterns, we concentrated on the HTTP server project for the following two reasons. First, for the time period studied, the HTTP server project was one of the largest and most successful ASF projects both in the number of developers and contributions. Second, access to archival data for this project proved to be less problematic than for some of the smaller ASF projects.

4 Data Sources

One basic tenet of OSS is that the development process and resulting products are "open" and freely available. Fundamentally, OSS projects represent large-scale publicly distributed software development processes. As such, and in keeping with free and open access, all OSS work products are placed in the public domain under various "free software" licensing arrangements.

For the purposes of this study, a participant refers to anyone participating in the Apache developer discussion group during the period in question. Apache developers are those individuals who have made a source code contribution to the Apache project during the time period studied. The "Apache Core" includes those Apache developers who make up the nucleus of the Apache HTTP project. There are approximately 22 Core participants. These 22 individuals account for more than 80% of all source code submissions to the Apache HTTP project. To operationalize the communication between Apache developers, two constituent or dyad communication matrices (i.e., adjacency matrices) were constructed from Apache developer email archives to record email communications between each dyad or pair of developers. The Apache projects maintain email list-serves to conduct all project related activities. The software used to maintain the email lists is fully RFC-822 compliant and supports conversation threads. A series of scripts were written to reconstruct conversation threads, identify the participants and produce various "flavors" of matrices suitable for input into UCINET. For the purposes of this research, a person participating in a thread was recorded as having a communication with all other thread participants.

5 Results

In this section we briefly describe some of the characteristics of the Apache communication network for the period we studied. Of interest here is the fact that the structure of the communication network essentially supports or reinforces what we already know about the Apache project from examination of the patch level contributions. That is, imagine the project as a funnel or a set of concentric circles, progressively getting refined or smaller. In other words, as participation increases the number of participants decrease. The full communication adjacency matrix for the focal period contains 453 nodes (individuals) and has a network density of .0218. Given the number of individuals involved in this network, we could have anticipated a relatively sparse network [22]. As a refinement on this network, we reduced the nodes to only those participants who were known to be active contributors to the Apache project during the period in question. This reduced the matrix to 83 nodes having a much greater network density of .25. As a further still refinement, we reduced the nodes to only those participants who were known to be in the Apache Core during the period in question. This reduced the matrix to 22 nodes with an extremely dense structure measured at .72.

To get a visual sense of the proximity, in terms of shared communication, of the Apache Core developers, we conducted a Multidimensional scaling (MDS) metric analysis of the similarity of the Core developers' communication matrix. The goal of our MDS analysis was to detect meaningful underlying dimensions that help to explain observed similarities in patterns of communication frequency among the Apache Core developers. Several measures of similarity were explored including Pearson's product-moment correlation and mean-centered cross products.

Interestingly, this analysis reveals three identifiable sub-groups even within the relatively small Core of the Apache development team. These sub-groups are identified in a series of concentric circles in Figure 1.

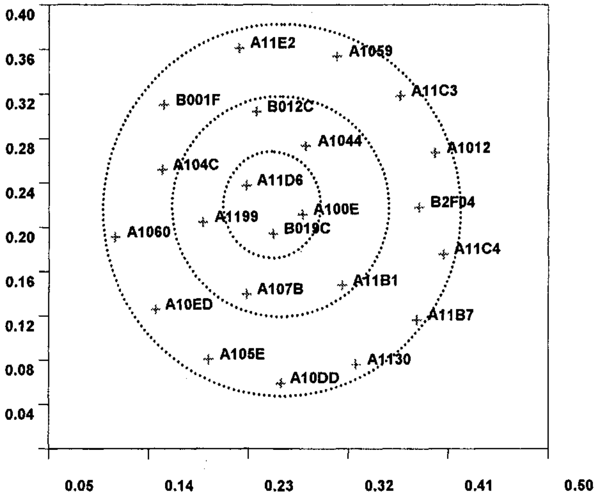

Fig. 1. Apache Core Communication Pattern Similarities – Metric MDS

We further visually explore the nature of the Apache Core developer's communication network by plotting the dyadic communication relationships between core developers using the MDS coordinates to position the developers in a graph. The resulting graph, or sociogram, represents the communication relationships among Apache Core developers (represented by points or nodes) and a "communicates-with" relationship (represented by connecting lines.) Figure 2 shows the full Apache Core communication network.

Fig. 2. Apache Core Communication Sociogram – Complete

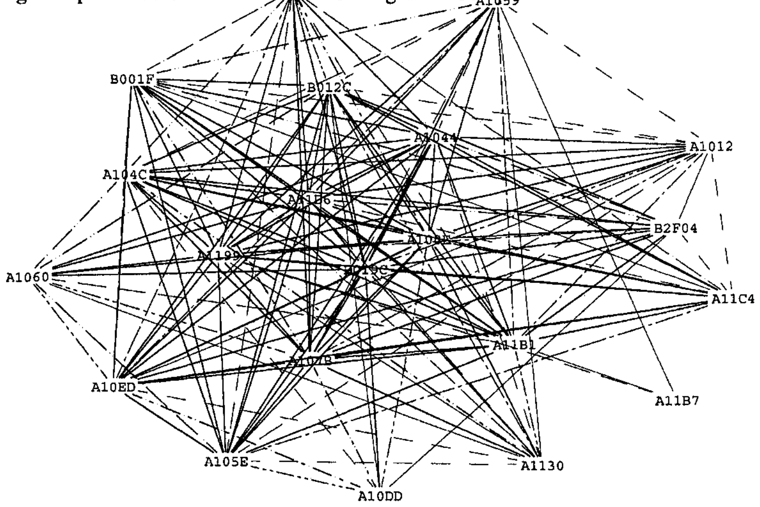

Figure 3 shows the network from the perspective of MDS Group 1. In this graph, members of Groups 2 and 3 each appear as a single collective entry. It is easily discernable from this graph that the MDS Group 1 developers constitute a fully connected communication graph.

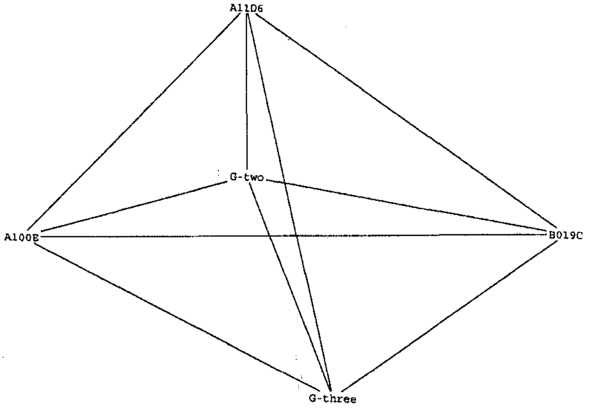

Fig. 3. Apache Core Communication Sociogram – MDS Group 1 Perspective

Similarly, Figures 4 and 5 show the network from the perspective of MDS Groups 2 and Group 3, respectively.

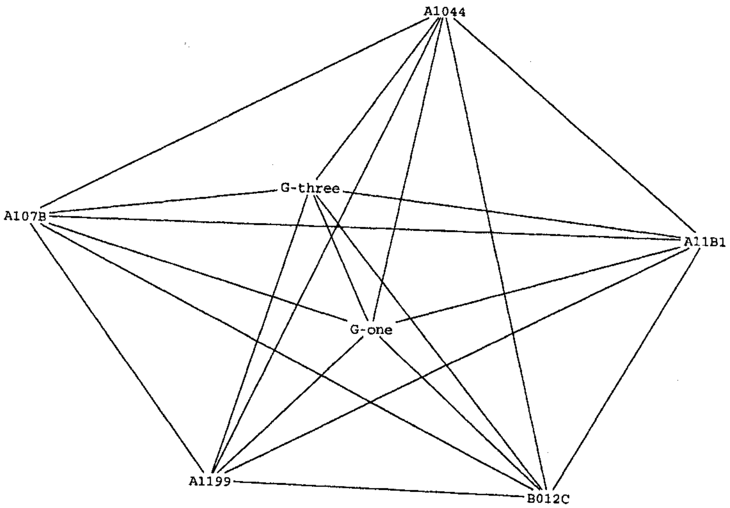

Fig. 4. Apache Core Communication Sociogram – MDS Group 2 Perspective

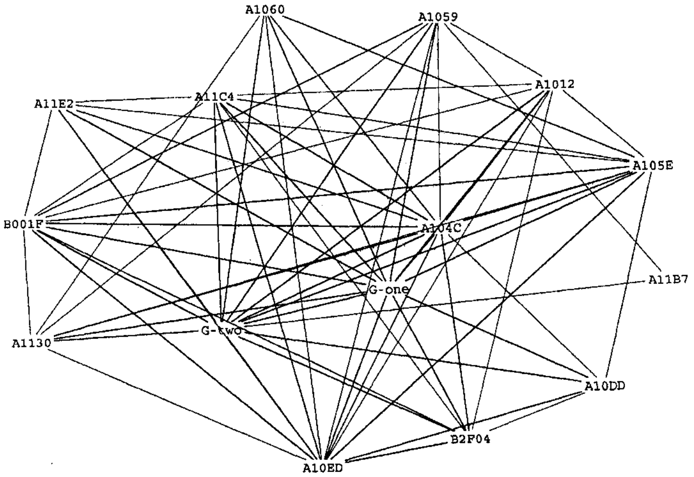

Fig. 5. Apache Core Communication Sociogram – MDS Group 3 Perspective

6 Discussion

From the network density measures, MDS plots of communication pattern similarity, and sociograms displaying the communication network structure of the Apache Core developers, we observe that the Apache Core maintains a relatively dense communication structure with active participation from all Core members. Further, from the MDS procedure we conclude that this Apache Core exhibits three identifiable sub-groups with varying degrees of influence and similarity within the communication network. For example, as shown in Figure 3, Group 1 consists of three developers (A100E, A11D6, B019C). Although Group 1 is smaller than the other two groups (Group 2 has five developers, and Group 3 has thirteen developers) the three developers in Group 1 are among the most central or prominent in the overall Apache core communication network in terms of their network centrality scores (see the central location of these three developers in the network illustrated in Figure 2). This suggests that a small number of prominent individuals are influencing communication patterns for the project. In general, our findings are consistent with Krackhardt's "Iron Law of Oligarchy", which is the tendency for groups to ultimately end up under the control of a few people.

Open source represents an exciting opportunity for research in a wide variety of disciplines. This paper applies social network analysis to understand how developers communicate in an open source project. Since the developers in open source projects are geographically distributed and may never meet face-to-face, it is important to understand how they communicate to organize and coordinate their efforts. Our analysis of the Apache HTTP server project suggests an interesting pattern of communication where the core developers self-organize into sub-groups that communicate intensely in completing the project. Our analysis also reveals that communications within the project are driven by a few prominent developers in one sub-group who are centrally located in the network. These results suggest interesting opportunities for future research. For example, one could examine whether developers in other OSS projects organize their communication patterns similar to the HTTP server project. One could also consider the influence of communication patterns on aspects of project performance or outcomes. Lastly, measures of influence and position within an OSS project's social networks may help explicate relationships between individual developer participation and performance.

7 References

1. OSI, The Open Source Definition, The Open Source Initiative, (Accessed: May 2001); http://opensource.org/docs/definition_plain.html

2. E. Raymond, *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary* (O'Reilly, Cambridge, 1999).

3. J. Lerner and J. Tirole, The Simple Economics of Open Source, The National Bureau of Economic Research, Inc. (Accessed: April 2001); http://papers.nber.org/papers/W7600

4. K. Lakhami, and E. von Hippel (2000). How Open Source Software works: "Free" user-to-user assistance. MIT Sloan Open Source Project, (Accessed: October 2001); http://opensource.mit.edu/papers/lakhanivonhippelusersupport.pdf.

5. A. Mockus, R. Fielding and J. Herbsleb, A Case Study of Open Source Software Development: The Apache Server, *Proceedings of the Proceedings of the 22nd International Conference on on Software Engineering*, Limerick Ireland (2000).

6. S. Koch and G. Schneider, Results for Software Engineering Research into Open Source Development Projects Using Public Data, Open Source Research Community, MIT Sloan Open Source Project, (Accessed: April 2001); http://opensource.mit.edu/papers/koch-ossoftwareengineering.pdf

7. I. Stamelos, L. Angelis, et al., Code Quality Analysis in Open Source Software Development, *Information Systems Journal* **12**(1), (2002).

8. B. Fitzgerald and J. Feller, Open Source Software: Investigating the Software Engineering, Psychosocial and Economic Issues, *Information Systems Journal* 11(4), (2001).

9. M. Granovetter, Economic Action and Social Structure: The Problem of Embeddedness, *American Journal of Sociology* **91**(3), 481-510 (1985).

10. R.S. Burt, *Structural holes: The social structure of competition* (Harvard University Press, Cambridge, 1992).

11. D. Krackhardt and L.W. Porter, The snowball effect: Turnover embedded in communication networks, *Journal of Applied Psychology*, 71, 50-55 (1986).

12. D. Brass, Being in the Right Place: A Structural Analysis of Individual Influence in an Organization, *Administrative Science Quarterly*, 29, 518-539 (1984).

13. J. Scott, Social Network Analysis (Sage Publications, Thousand Oaks, 2000).

14. L.C. Freeman, *Centrality in Social Networks: Conceptual Clarification, Social Networks* **1**, 215-239 (1979).

15. R.A. Ghosh, Interview with Linus Torvalds: What motivates free software developers?, *First Monday* **3**(3), (1998).

16. N. Bezroukov, Open Source Software Development as a Special Type of Academic Research, *First Monday* **4**(10), (1999).

17. R.E. Kraut, C. Egido, et al., Patterns of Contact and Communication in Scientific Research Collaboration, *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work.* J. Galegher, R. E. Kraut and C. Egido Eds., (L. Erlbaum Associates, Hillsdale), 149-171 (1990).

18. K. Carley and K. Wendt, Electronic Mail and Scientific Communication: A Study of the SOAR Extended Research Group, *Knowledge: Creation, Diffusion, Utilization* **12**(4), 406-440 (1991).

19. Netcraft, The Netcraft WEB-server Survey, (Accessed: August 2005); http://news.netcraft.com/archives/2005/08/01/august_2005_WEB_server_survey.html
.

20. I. Hann, J. Roberts, S.A. Slaughter and R. Fielding, Economic incentives for participating in open source software projects. *Proceedings of the 22nd International Conference on Information Systems*, Barcelona (2002).

21. Apache Software Foundation, (Accessed: March 2001); http://www.apache.org/foundation/, Apache Software Foundation.

22. S.R. Barley, J. Freeman, et al, Strategic alliances in commercial biotechnology, *Networks and Organizations: Structure, Form, and Action*, N. Nohria and R. G. Eccles Eds., (Harvard Business School Press, Boston), 311-347 (1992).