Open Innovation/Open Source Leadership

Research in Progress

U. Yeliz Eseryel
University of Groningen
Faculty of Economics & Business
Department of Innovation Management & Strategy
yeliz@eseryel.com

Abstract

Open source software development (OSS) teams provide a successful open innovation context. There is an increasing interest by companies in either collaborating with OSS teams or adopting similar technology-intensive open innovation practices. Yet, their key challenge is to build and maintain a sustainable and innovative community around these projects. Leadership is key to this process as leadership plays an important role in bringing disparate partners together in a network. Yet, there is a lack of understanding of how leaders emerge and facilitate open innovation in these settings. To fill this gap, the following research question is posed: What are the determinants of leadership emergence in the open innovation setting of OSS teams? This research-in-progress paper presents a brief overview of how this research question is answered based on an investigation of two teams at a single time point.

Keywords

Open innovation, open source software teams, leadership.

Introduction

Open innovation is the process of systematically encouraging and exploiting a wide range of internal and external knowledge sources for accelerating innovation (Chesbrough 2003; Conboy and Morgan 2011). Shorter innovation cycles, industrial research and the rising costs of development increasingly require organizations to move from closed to open innovation practices (Gassmann and Enkel 2001). In the information systems field, open source software development (OSS) teams provide a successful private-collective open innovation context where participants use their own resources to privately invest in creating a novel product and then revealing it to the public (Von Hippel and von Krogh 2003). OSS teams are teams of individuals who collaborate from around the world using information technologies to develop software, whose source code is available to view and change. OSS teams such as Linux, Apache, Mozilla have been widely successful in bringing together global expertise and keeping an international community engaged in an open innovation process. Open source is identified by West and Gallager (2006) as a 'great examplar of open innovation because of the shared rights to use the resulting technology as well as the collaborative development of the technology' (p. 322).

There is an ever-increasing interest by companies in either collaborating with OSS teams or adopting similar technology-intensive practices that foster open innovation (Dinkelacker et al. 2002; Gurbani et al. 2005; Morgan et al. 2011; Wesselius 2008). Yet, their key challenge in achieving this goal is to build and maintain a sustainable and innovative community around these projects (Preece 2000; Sirkkala et al.

2009). Leadership is key to this process as leadership plays an important role in bringing disparate partners together in a network (Gomes-Casserres 2003). Yet, there is a lack of understanding of how leaders emerge and facilitate open innovation in these settings. It is crucial to understand how leadership emerges for both open source software teams (Dinkelacker et al. 2002; Giuri et al. 2008) and open innovation settings in general (Von Hippel and von Krogh 2003). Lack of understanding in this area causes companies collaborating in OSS to be ill prepared for participation and direction setting in these communities. When the nature of leadership is not understood, it may add to the challenge of developing and maintaining innovative communities around the OSS projects. To fill this gap, the following research question is posed: What are the determinants of leadership emergence in the open innovation setting of OSS teams?

This research-in-progress paper presents a brief overview of how these research questions are answered with a study of two teams at a single time point. Preliminary answer to the first question is provided in this paper.

Background

An Overview of the General Leadership Literature

Leadership is in the eye of the beholder (Lord, Foti and Philips 1982). Followers' perceptions are the key to leader and team performance (Hollander 1992). Unfortunately, most popular and contemporary approaches to leadership have not explicitly considered the role of person perception processes (Brown and Lord 2001).

Leadership can be recognized from the behaviors which are revealed through normal, day-to-day interactions with others (Lord and Maher 1990). In OSS teams, these behaviors are embedded in how information technologies are used to communicate, coordinate, and execute the software development tasks. Perception process is important since perception of leadership produces behavior in followers and thus influence team performance (Hersey et al. 1979). Therefore, OSS members aspiring to emerge as successful leaders should understand which behaviors are perceived as leadership behaviors in these contexts, and how they can contribute to the open innovation process.

Based on a long stream of leadership research, leadership behaviors are categorized under task- and relationship-oriented behaviors (Yukl 2002). These two types of leader behaviors, which are described below, are also found relevant to global teams (e.g., Kayworth and Leidner 2001; Sudweeks and Simoff 2005; Tyran et al. 2003; Weisband 2002; Yoo and Alavi 2004).

Task-oriented behaviors are those that move the team forward in the accomplishment of its task, such as planning and scheduling work, coordinating subordinate activities, and providing necessary supplies, equipment, and technical assistance (Yukl et al. 2002), integrating the teams' contributions (Morse and Wagner 1978; Yoo and Alavi 2004), and defining timeline (Kayworth and Leidner 2000; Stogdill 1974). Task coordination is an important leadership behavior in organizational leadership, which is also observed in global team literature. Yoo and Alavi (2004) concluded that emergent global team leaders sent more task-oriented email messages, particularly ones related to coordination. Task coordination is identified as a leadership behavior in open source software development teams (Crowston et al. 2007a; Crowston et al. 2007b).

Relationship-oriented behaviors are those that allow the team to maintain a positive social environment. Leaders of global and face-to-face teams are found to exhibit the following relationship-oriented behaviors: encouraging contribution (Bales 1950; Scialdone et al. 2007; Yukl et al. 2002), showing trust and confidence (Yukl et al. 2002), acting friendly, considerate and kind (Cartwright and Zander 1960; Kayworth and Leidner 2001; Scialdone et al. 2007; Yukl et al. 2002), helping to develop others, act as mentors (Kayworth and Leidner 2001; Yukl et al. 2002), keeping others informed (Yukl et al. 2002), resolving conflicts (Bales 1950; Bass 1990; Cartwright and Zander 1960; Morse and Wagner 1978) and giving or asking for opinions (Bales 1950; Cartwright and Zander 1960).

Open Source Team Leadership and Governance

While this section is limited to an overview of OSS governance and leadership, a general review of the OSS literature is conducted by Crowston and colleagues (2012). The OSS teams in this study have the following governance structure: they are voluntary teams under non-profit organization Apache Foundation (Bonaccorsi and Rossi 2006; Wasserman 2009). Apache projects are similar to cross-organizational teams with participants working for different companies, and their internal customers are the developers, and external customers may include the companies/clients of the developers and other end-users.

Most engineering cultures value technical contributions over all else, eschewing positional power (Wasserman 2003). This applies to the OSS setting where individuals need to make technical contributions as the minimum membership requirement (von Krogh et al. 2003). Solving challenging problems gain esteem in the eyes of other members (Fleming and Waguespack 2005) and translate into a higher potential for future leadership (Cox 1998; Fleming and Waguespack 2005; Gacek and Arief 2004; Moon and Sproull 2000; Scozzi et al. 2008).

OSS hierarchy is represented as a an onion-like structure at the center of which are the core developers who develop code, followed by bug-fixing co-developers, and bug-reporting active users and passive non-contributing users respectively (Crowston and Howison 2006). While core members may be candidates to be leaders, leadership of the core members cannot be claimed without further analysis (Fleming and Waguespack 2005; Howison et al. 2006). Similarly a leader may move from core to periphery, while still being perceived as a leader. While some OSS teams have administrative roles such as the "release manager" in Netbeans project and "project management committee chair" in Apache, these are administrative roles and not necessarily leadership roles (Apache Software Foundation 2007; Jensen and Scacchi 2005).

Connecting The Extant Leadership Literature to OSS Setting

Theories of organizational behavior and leadership may not apply similarly to the OSS setting just as general group effectiveness models do not apply to self-managing teams (Cummings 1981; Hackman 1986). We do not know how much of the extant leadership literature applies to OSS setting and how for the following reasons.

First, in the OSS teams, the agency-role played by information systems influence the symbolic interaction, which may influence how extant leadership theories transfer to this setting (Carte et al. 2006; Hooijberg et al. 1997; Zhang and Fjermestad 2006). This creates a gap in the information systems literature on the meaning and practice of leadership in global team setting (Bell and Kozlowski 2002; Yoo and Alavi 2004; Zaccaro and Bader 2003; Zhang and Fjermestad 2006). In this context, symbolic interaction refers to how individuals use and interact with information systems for their software development tasks, and through that interaction how they create, interpret and transform their understanding on what leadership means for them (Blumer 1937).

Sociomaterial choices regarding the selection, configuration and use of various information technologies change the social interaction and work practices of open innovation teams. These practices may impact, within the context of technology use, what kinds of behaviors constitute leadership. Yet, existing leadership literature generally disregards the sociomateriality of the information technologies that are influence all interactions of OSS team members.

Second, most leadership studies were done in broader contexts, rather than the focused work contexts of software development (Faraj and Sambamurthy 2006). OSS leadership may show unique patterns: participants bring specific slices of expertise and attempt to integrate their expertise in determining the scope, providing successful management and ensuring stakeholder satisfaction (Faraj and Sambamurthy 2006). The software development expertise that is brought about by these individuals also shapes their symbolic actions with the technology, and this in turn structures the sociomateriality of technology regarding team leadership. Therefore, focused studies of software development leadership are needed (Faraj and Sambamurthy 2006) to understand how to lead OSS open innovation setting.

Method

Given limited theory about OSS leadership, a qualitative approach, was chosen to identify relevant constructs that contribute to leadership emergence. At the core of qualitative paradigm is the notion that people assign meaning to the relevant phenomenon (leadership, in this case) (Tesch 1990). Furthermore, people's experiences are situated within social context (open innovation context in this case) (Tesch 1990). This study aims at understanding the social phenomenon of leadership emergence and its connection to open innovation from the viewpoint of participants, through detailed descriptions of leadership actions and through the richness of meaning associated with observable behavior (Wildemuth 1993). Qualitative interviews were based on conversation, and interviewees were meaning makers (Holstein and Gubrium 1995). Using qualitative interviews to derive interpretations from respondent talk (Warren 2001) is especially useful for developing theoretical insights when research focuses on areas that extant theory does not address well (Ozcan and Eisenhardt 2009).

Site Description

During the pilot phase of the study, I identified that teams range in the number of perceived leaders. Those with fewer leaders have more centralized leadership, and those with more leaders have more decentralized leadership. As one goal of the study was to understand what leadership constitutes in OSSs, I purposefully selected 1 team, that had a single leader (referred to as Central from here on) and 1 team that had a highly decentralized leadership (referred to as Decentral from here on). The logic behind this purposeful selection was that, if highly decentralized and highly centralized leadership teams are similar in the type of behaviors that constitute leadership, than these behaviors can be generalized across all OSSs regardless of their leadership centralization.

Central, the centralized leadership OSS team, provides directory solutions written entirely in Java. First developed in October 2002, it became an Apache project in April 2003. During this study, the project had 21 committers, and 3 emeritus committers. Central provides complex niche software with 254,000 lines of codes (as of September 2008). As a founder, Bob (all names are pseudo names) had been very actively involved throughout the project's life. He also held the project management chair role, the formal administrative role for the project. During the pilot stage and in informal communications with the team members and with Apache, Bob was identified as the very strong leader of the project. Yet, at the time of the formal data collections, two individuals, Bob and Eduardo, were identified as very strong leaders. Eduardo had been involved with the project soon after Central became an Apache project and had since been with the group. Eduardo had also formed a consulting business around the software that Central developed and even hired a full time employee, who also became an active member of Central.

Decentral, the decentralized leadership project, is a high-performance, full-featured text search engine library written entirely in Java. Developed by Darren in 1998, it became an Apache project in April 2005. Its user base includes many government agencies across the world and well-known companies such as AOL, Apple, Monster, and Wikipedia. Decentral had 12 committers, 6 contributing committers and 15 emeritus committers at the time of the study. Darren was highly active for a long time at the beginning of the project and helped form a team within Apache. In time, he developed other relevant sub-projects and focused more on one of them. At the time of the study, Darren was not very much involved with the day-to-day work and decisions of the team, but his continued existence as a leader was recognized by the team members. The team members reported that he follows the work and the communication that was going on, but would choose to be silent, unless the subject was very important. Then he would gently state his opinion, which would always easily be accepted by the team. In addition to Darren, Janka and Christian were the other individuals who were perceived as leaders by the team members. Darren was the founder and project management chair, who was more silently watching and as needed joining in. Janka and Christian were active contributors in the discussion and the work of Decentral.

Data Collection, Analysis and Writing

First, a pilot study is conducted in order to (1) understand the nature of leadership in Apache OSSs, (2) gain access to the research sites. Leaders from different teams are identified with snowball sampling, where each respondent helps locate others that fulfill theoretical criteria, which was leadership in this case (Arksey and Knight 1999; Weiss 1994). This was followed with interviews with two teams' members.

The interviews were semi-structured and lasted 60 to 90 minutes. Each interview was taped. I inductively analyzed the interviews with another independent coder. During this analysis we used Atlas-Ti for coding and memoing our observations of the data. Two levels of coding, open coding and selective coding were used (Charmaz and Mitchell 2001). Independently coded data was categorized through a process of constant comparison based on open coding principle (Strauss and Corbin 1990). Then, I discussed with the other independent coder all of the inductive codes until we reached full agreement. Lastly, we applied selective coding by abstracting these codes to a higher level, therefore synthesizing the data (Charmaz and Mitchell 2001) and identifying meaningful categories.

The interview protocol focused on understanding, from the point of view of the participants, who the perceived leaders are, what they do to help contribute to team innovativeness and why they are perceived as a leader, as well as the factors inherent in the team and team's functioning that may play a role in leadership. Furthermore, the pilot interviews enabled me to learn about the setting, and then be ready to ask in the team interviews more in-depth questions. For example, during the analysis of the pilot interviews, the importance of execution came about. Then in the team interviews, whenever this subject came up, I questioned further what kind of contributions are considered a sufficient, whether there are individuals who do only the small and unwanted tasks and if so if they are seen as leaders, or not.

The fact that interviews with general team leaders at Apache was followed by the two case studies, enabled me to reach saturation without having to do another round of interviews. I observed saturation of findings when the interviewees seemed to repeat what I had already known through the previous interviewees.

One key strength of the data collection and analysis effort was my ability to observe the actual interaction between the team members, work and coordination behaviors through the teams' publically available information technologies. Secondly, I had access to the key informants through social networking tools, such as facebook, twitter, google chat and skype. This allowed me, as researcher, to check back easily and quickly on my questions and assumptions with the team members during the memoing and coding process and based on my discussions with the second coder.

Based on the analysis, I wrote within-case studies of each of the team as a basis for cross-case comparisons (Eisenhardt 1989). I have also shared these case studies with the second coder for triangulation purposes. My findings on this and other studies were also shared with the Apache open source community the next year through presentations at their conference and participation in their workshops, which enabled further triangulation and feedback.

Preliminary Findings and Discussion

This section presents the preliminary findings regarding the first research question (Figure 1).

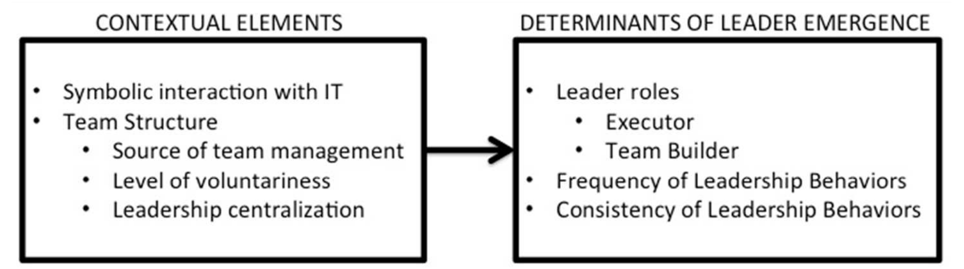
CONTEXTUAL ELEMENTS
• Symbolic interaction with IT
• Team Structure
• Source of team management
• Level of voluntariness
• Leadership centralization
DETERMINANTS OF LEADER EMERGENCE
• Leader roles
• Executor
• Team Builder
• Frequency of Leadership Behaviors
• Consistency of Leadership Behaviors

Figure 1. Mechanism of Leadership Perception Over Time

Determinants of ISD Leader Emergence

Both the pilot study and the two case studies resulted in the same general observation: Leaders exhibited at least one of the leader roles and also provided high levels of leadership behaviors and did so consistently over time.

Leader Roles

The individuals who were perceived as leaders played two different roles based on the most dominant behaviors they exhibit: (1) Executor, and (2) Team Builder.

A consistently observed role identified both during the pilot study and in both teams was that of an "Executor". Executors were those individuals who had good, doable ideas and who were able to show that these ideas worked by actually implementing these ideas themselves. As one identified leader suggested:

> *You don't become a leader by declaring "I'm now a leader" or by saying "I incepted the project so you must look to me for guidance". But it's actually how much and how long you work in the community that matters.*

Talking to the team members and questioning the nature of the executor role brought about the complicated nature of this role. Executors were those who were "stepping up" or "taking the initiative" to make major contributions or to bring about much needed change. By doing these, they accomplished more than just contributing to team's work, they also signaled information about themselves which created a leader perception and image in the eye of the other team members. Perhaps very importantly, they signaled that they were the domain experts, and that they had good ideas, which reportedly makes the team members take the suggestions of executors more seriously than those of others.

In order to double-check whether being an executor is necessary or if having good ideas were sufficient, I asked interviewees to identify some people who have many ideas, but who may not execute them. In cases the interviewees identified such people, they did not view those as leaders clearly because they were not contributing beyond ideas.

Based on the interviewer comments, the expertise and the execution behaviors of the executors were cyclic in nature: The executors were first seen as experts because of their positive contributions to work, and then the more they contributed the more they learned the details about an area, therefore they could make more contributions in that area. This would indicate that the leaders would keep on being perceived as leaders as long as they keep turning ideas into action.

> *Undoubtedly that project has such a leader, which is him [her], a very senior and experienced kind of person, who knows a lot about Apache and Java and about team management. How do I know this? I can see it in his (her) arguments on the listserv and on the work s/he does.*

On the other hand, executors were not too proud to do the little things that need to be done. In that sense, they carried the humility of an entrepreneur who writes the million-dollar business proposal and also answers phone calls and takes the trash out. The executor's behavior of doing the important work, and on the side be willing to do the boring tasks that needed to be done for the good of the project seemed to communicate to others that they care about the team:

> *Yeah, he cares about the project; he steps up and does the work that is no fun. Like changing headers in tens of documents. Somebody's gotta do it before I do the release, and he does it.*

The second leader role that was often mentioned was the team builder role. The main components of this role included providing mentorship and guidance to other members to enable them to contribute to the team and showing that their contributions were appreciated, once they contribute. These behaviors ranged from giving very timely and thorough support (mentorship) to simply saying 'thank you', 'well-done' or 'sorry' when needed:

> *As we were writing the code together s/he gave us guidance on questions where the specifications were completely ambiguous. S/He would say, "well this server behaves*

> *this way, that server behaves that way". S/He would give us feedback. This allowed us to come up with solutions that I otherwise wouldn't have.*

It seemed that most of the executors were also team builders. In only one of the cases, in the case of Decentral, one of the team leaders, who was both an executor and a team builder, rarely provided negative behaviors that were not in line with his/her team builder role: In one case he responded in a harsh way to one of the team members according to the interviewees, where the other team leader jumped in to ease the situation. There were also reports of two people with strong opinions about the project that did not overlap with the general opinions of this leader. In these cases the other team member chose to stop working on the team. However, beyond these reports, this leader of Decentral put much effort in creating a positive team environment and especially in finding and recruiting new team members who would contribute to the project. Several of the team members reported being brought into the team by this leader.

Team builder is identified to be an important team leader role, which almost always accompanied the executor role. Only on one occasion team builder role was mentioned as the main reason why somebody was identified as a leader. In this occasion, the listserv that is used by the team was under a troll attack. (In internet slang, troll is someone who posts inflammatory or irrelevant messages in an online community, with the primary intent of provoking readers into an emotional response or of otherwise disrupting normal on-topic discussion (Anonymous 2012). This person stopped the process, which was becoming disruptive to the team by sending everyone private messages saying, "Please do not respond to any messages of the troll". Thus, s/he managed to get rid of the troll. Again, s/he was the only individual who was more of a team builder than an executor. While s/he also contributed to the team's work, s/he did not do so with high frequency.

Frequency and Consistency of Leadership Behaviors

An element that came through continuously in conversations related to executors is that they exhibit numerous leadership behaviors and do so consistently over long periods of time.

> *Because I've seen his (or her) work for a long time now, I know that s/he knows what s/he is talking about.*

Both of the team's founders provided high level of both work execution and team building behaviors over long periods of time and they were both identified as leaders informally an per the group's nomination, acting as formal group leaders.

Contextual Elements Relevant to leader Emergence

Role Played by IT

The level of IT use and the functions for which IT is used determines the role played by IT. In both Decentral and Central, all team members were very comfortable with using IT and they were advanced users of technology. Information technologies were used as their main medium for communication (listservs), task coordination among team members (issue trackers, namely JIRA) and the execution of work in a distributed manner (software versioning system, namely SVN). Both teams used websites and email listservs to communicate and share information with their users. Central also used other tools for deciding on and planning its future improvements.

When asked in an open-ended way about the task coordination behavior, by asking how the team members coordinate their actions, the interviews mentioned that they are using emails and issue trackers to communicate, find the open to-do items and to complete those items they thing should be done. To further check if the any of the leaders do the well-known task-oriented behavior of "task coordination", I repeated my question, this time in a more direct way: "Does anyone help with this coordination?" the answer was clear.

> *Why should anyone have to coordinate? The issue trackers are up there; everybody can identify a new issue or resolve an open one.*

Very similarly, when I asked a Central member about how the team decides their vision for the future of the software, an interviewee described in detail what the upcoming vision is, and how they decided on it "all together, using the road-mapping tools". To be sure, I picked parts of the vision that the interviewee mentioned and asked "who decided on this?", the answer was the same, "We all decided together". Thus even in the more centralized leadership team, IT tools create an environment of shared -decision making. This is in line with the previous researchers who suggest that the use of publicly available information technologies for all their work created high visibility and transparency for both team members and for researchers (von Krogh and Spaeth 2007). Furthermore, this visibility and transparency removes the need for an individual to take on task coordination role as a leadership behavior. Instead, those who believe in the sociomateriality of the information and communication technologies may suggest that information technologies take on the leadership role of task coordination. Therefore, task coordination behavior does not become a determinant of leadership emergence in OSSs.

Team Structure

The team structure may be described with a number of elements such as (1) the source of team management, (2) the level of voluntariness, and (3) the centralization of leadership. While this list is not a comprehensive list of elements that define team structure, these three elements were identified as important elements related to OSS teams. In both Central and Decentral, the teams were self-managing, fully voluntary individuals and they differed in the centralization of their leadership. While I could not find any differences in leadership behaviors valid between centralized and decentralized team, it was clear that the team structure explained some of the difference of OSS leadership from the organizational team leadership. The perceived leaders explained how it was impossible to coordinate tasks by asking others to do some work because it was voluntary. Instead they said

> *You cannot tell anybody what to do here. Nobody is the boss of another!*

In this case, the team structure creates an anti-leadership norm, by suggesting that task coordination by determining what others will do is not only not a determinant of leadership emergence in these OSSs, but also it is an unacceptable behavior.

Conclusion and Implications

Findings show that there are three key determinants of OSS leader emergence: leadership role that the leaders play based on their most dominant leadership behaviors, and the frequency and consistency by which they exhibit these roles. This finding supports earlier research, that suggests that individuals are perceived as leaders by others based on sustained and strong technical contributions (Scozzi et al. 2008).

The key leadership role, being an executor is an important distinction between traditional organizational leadership literature and OSS research. Specifically, there is a long term leadership that focuses on the visionary role of the leaders (Northouse 2010; Westley and Mintzberg 1989). Strategic leaders typically provide a new vision based on their knowledge which is accumulated over the years (Westley and Mintzberg 1989). In the strategic leadership literature, the "execution" behavior of the leader is important only because it is an enabler of vision creation. On the other hand, in this study, being an executor helps individuals emerge as leaders even without a vision creation behavior that includes developing and communicating vision. This contradicts earlier research in open source software research, which suggests that the main duties of a leader in OSS projects include providing vision (Crowston et al. 2012).

Moreover, the second determinant of leadership emergence brings about the need for future research on for longitudinal studies of leadership to show how leadership may change due to the frequency and consistency of leader behaviors. The intended next steps in this research include the testing of this model of which factors contribute to leadership emergence over time through quantitative methods. For example the question of which executor and team builder behaviors bring about leadership emergence in the long run, is one that will be answered through structural equation modeling technique.

The model shows that the contextual factors influence which behaviors influence these leadership determinants. These contextual factors are the team structure of the OSS and the role that IT plays in the team. This gives an opportunity to practitioners to change the nature of the IT tools and the team structure, in order to adjust the leadership responses. An important feature of IT in this study is the transparencies that it provides, which eliminates the need for certain leadership behaviors, and distributes the performance of these behaviors to the whole team, thereby enabling the leaders to focus on more crucial leadership functions.

Regarding the limitations of the study, due to varying structures of OSSs, the readers are cautioned about generalizing the study's findings mainly to organizational OSS teams and OSSs which are similar in terms of umbrella organization, structure, norms, end-customer, and member motivation to the Apache teams used for this study.

References

Anonymous. 2012. "Troll (Internet)." Retrieved May 2, from en.wikipedia.org/wiki/Troll_(Internet)

Apache Software Foundation. 2007. "What Are the Duties of the Pmc Chair and How to Perform Them." Retrieved February 1, 2007 from http://www.apache.org/dev/pmc.html - chair

Arksey, H., and Knight, P. 1999. *Interviewing for Social Scientists: An Introductory Resource with Examples*. Thousand Oaks, CA: Sage.

Bales, R.F. 1950. "A Set of Categories for the Analysis of Small Group Interaction," *American Sociological Review* (15:2), pp. 257-263.

Bass, B.M. 1990. *Bass & Stogdill's Handbook of Leadership*, (3rd ed.). New York: The Free Press.

Bell, B.S., and Kozlowski, S.W.J. 2002. "A Typology of Virtual Teams: Implications for Effective Leadership," *Group & Organization Management* (27:1), Mar, pp. 14-49.

Blumer, H. 1937. "Social Psychology," in *Man and Society,* E. Schmidt (ed.). New York: Prentice-Hall, pp. 144-198.

Bonaccorsi, A., and Rossi, C. 2006. "Comparing Motivations of Individual Programmers and Firms to Take Part in the Open Source Movement: From Community to Business," *Knowledge, Technology & Policy* (18:4), pp. 40-64.

Brown, D., and Lord, R. 2001. "Leadership and Perceiver Cognition: Moving Beyond First Order Constructs," in *How People Evaluate Others in Organizations,* M. London (ed.). Mahwah, NJ: Lawrence Erlbaum Assoc Inc, pp. 181-202.

Carte, T.A., Chidambaram, L., and Becker, A. 2006. "Emergent Leadership in Self-Managed Virtual Teams: A Longitudinal Study of Concentrated and Shared Leadership Behaviors," *Group Decision and Negotiation* (15:4), Jul, pp. 323-343.

Cartwright, D., and Zander, A.F. 1960. *Group Dynamics: Research and Theory*. Evanston, IL: Row, Peterson & Co.

Charmaz, K., and Mitchell, R.G. 2001. "Grounded Theory in Ethnography," in *Handbook of Ethnography,* P. Atkinson, A. Coffey, S. Delamont, J. Lofland and L. Lofland (eds.). London: Sage Publications, pp. 160-174.

Chesbrough, H.M. 2003. *Open Innovation*. Boston, MA: Harvard Business School Press.

Conboy, K., and Morgan, L. 2011. "Beyond the Customer: Opening the Agile Systems Development Process," *Information and Software Technology* (53:5), pp. 535-542.

Cox, A. 1998. "Cathedrals, Bazaars and the Town Council." Retrieved Jan 10, 2012, from http:// slashdot.org/features/98/10/13/l423253.shtml

Crowson, K., Wei, K., Howison, J., and Wiggins, A. 2012. "Free/Libre Open Source Software Development: What We Know and What We Do Not Know," *ACM Computing Surveys* (44:2), pp. 7:1-7:35.

Crowston, K., Heckman, R., and Misiolek, N. 2007a. "A Structurational Perspective on Leadership in Virtual Teams," *IFIP Working Group 8.2 Working Conference on Virtuality and Virtualness,* K. Crowston and S. Sieber (eds.), Portland, OR.

Crowston, K., Heckman, R., Misiolek, N., and Eseryel, U.Y. 2007b. "Emergent Leadership in Self-Organizing Virtual Teams," in: *Twenty Eighth International Conference on Information Systems* Montreal, Canada: AIS, p. Paper 60.

Crowston, K., and Howison, J. 2006. "Hierarchy and Centralization in Free and Open Source Software Team Communications," *Knowledge, Technology & Policy* (18:4), Winter2006, pp. 65-85.

Crowston, K., Wei, K., Howison, J., and Wiggins, A. 2012. "Free/Libre Open Source Software Development: What We Know and What We Do Not Know," *ACM Computing Surveys* (44:2), pp. 7:1-7:35.

Cummings, T.G. 1981. "Designing Effective Work Groups," in *Handbook of Organizational Design,* I. Nystrom, P.C. and W.H. Starbuck (eds.). London: Oxford University Press, pp. 250-271.

Dinkelacker, J., Garg, P.K., Miller, R., and Nelson, D. 2002. "Progressive Open Source," *Proceedings of the International Conference on Software Engineering (ICSE '02)*, Orlando: ACM Press, pp. 177-184.

Eisenhardt, K.M. 1989. "Building Theories from Case Study Research," *Academy of Management Review* (14:4), pp. 532-550.

Faraj, S., and Sambamurthy, V. 2006. "Leadership of Information Systems Development Projects," *IEEE Transactions on Engineering Management* (53:2), pp. 238-249.

Fleming, L., and Waguespack, D. 2005. "Penguins, Camels, and Other Birds of a Feather: Brokerage, Boundary Spanning, and Leadership in Open Innovation Communities." Boston: Harvard Business School.

Gacek, C., and Arief, B. 2004. "The Many Meanings of Open Source," *IEEE Software* (21:1), pp. 34-40.

Gassmann, O., and Enkel, E. 2001. "Towards a Theory of Open Innovation: Three Core Process Archetypes." Retrieved June 24, 2013, from http://www.alexandria.unisg.ch/Publikationen/274

Giuri, P., Rullani, F., and Torrisi, S. 2008. "Explaining Leadership in Virtual Teams: The Case of Open Source Software," *Information Economics and Policy* (20:4), pp. 305-315.

Gomes-Casserres, B. 2003. "Competitive Advantage in Alliance Constellations," *Strategic Organisation* (1:3), pp. 327-335.

Gurbani, V.K., Garvert, A., and Herbsleb, J.D. 2005. "A Case Study of Open Source Tools and Practices in a Commercial Setting," *5th Workshop on Open Source Software Engineering*, St. Louis: ACM New York, pp. 1-6.

Hackman, J.R. 1986. "The Psychology of Self-Management in Organizations," in *Psychology and Work: Productivity, Change and Employment,* M.S. Pallak and R.O. Perloff (eds.). Washington, DC: American Psychological Association, pp. 87-136.

Hersey, P., Blanchard, K.H., and Natemeyer, W.E. 1979. "Situational Leadership, Perception, and the Impact of Power," *Group and Organization Studies* (4:4), pp. 418-428.

Hollander, E.P. 1992. "Leadership, Followership, Self, and Others," *The Leadership Quarterly* (3:1), pp. 43-54.

Holstein, J.A., and Gubrium, J.F. 1995. *Active Interview*. Thousand Oaks, CA: Sage.

Hooijberg, R., Hunt, J.G., and Dodge, G.E. 1997. "Leadership Complexity and Development of the Leaderplex Model," *Journal of Management* (23:3), pp. 375-408.

Howison, J., Inoue, K., and Crowston, K. 2006. "Social Dynamics of Free and Open Source Team Communications," *Proceedings of the IFIP 2nd International Conference on Open Source Software,* E. Damiani, B. Fitzgerald, W. Scacchi and M. Scotto (eds.), Lake Como, Italy: Springer, Boston, USA, pp. 319-330.

Jensen, C., and Scacchi, W. 2005. "Collaboration, Leadership, Control, and Conflict Negotiation in the Netbeans. Org Community," in: *38th Hawaii International Conference on System Sciences*.

Kayworth, T., and Leidner, D. 2000. "The Global Virtual Manager: A Prescription for Success," *European Management Journal* (18:2), pp. 183-194.

Kayworth, T.R., and Leidner, D.E. 2001. "Leadership Effectiveness in Global Virtual Teams," *Journal of Management Information Systems* (18:3), pp. 7-40.

Lord, R.G., and Maher, K.J. 1990. "Alternative Information Processing Models and Their Implications for Theory, Research, and Practice," *Academy of Management Review*:15), pp. 9-28.

Moon, J.Y., and Sproull, L. 2000. "Essence of Distributed Work: The Case of Linux Kernel," *First Monday* (5:11).

Morgan, L., Feller, J., and Finnegan, P. 2011. "Exploring Inner Source as a Form of Intraorganisational Open Innovation," *ECIS 2011* Helsinki, Finland: AIS Electronic Library, p. Paper 151.

Morse, J.J., and Wagner, F.R. 1978. "Measuring the Process of Managerial Effectiveness," *The Academy of Management Journal* (21:1), pp. 23-35.

Northouse, P.G. 2010. *Leadership: Theory and Practice*, (5th ed.). Thousand Oaks, CA: Sage Publications.

Ozcan, P., and Eisenhardt, K.M. 2009. "Origin of Alliance Portfolios: Entrepreneurs, Network Strategies, and Firm Performance," *Academy of Management Journal* (52:2), pp. 246-279.

Preece, J. 2000. *Online Communities: Designing Usability, Supporting Sociability*. Chichester: Wiley.

Scialdone, M.J., Na, L., Howison, J., Crowston, K., and Heckman, R. 2007. "Group Maintenance in Technology-Supported Distributed Teams," in: *2008 Academy of Management Annual Meeting: The Questions We Ask.* Anaheim, CA: Syracuse University School of Information Studies.

Scozzi, B., Crowston, K., Eseryel, U.Y., and Li, Q. 2008. "Shared Mental Models among Open Source Software Developers," *41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, Hawaii, pp. 306-316.

Sirkkala, P., Aaltonen, T., and Hammouda, I. 2009. "Opening Industrial Software: Planting an Onion," in *Open Source Ecosystems: Diverse Communities Interacting,* C. Boldyreff, K. Crowston, B. Lundell and A. Wasserman (eds.). Boston: Springer, pp. 57-69.

Stogdill, R.M. 1974. *Handbook of Leadership: A Survey of Theory and Research*. New York: Free Press.

Strauss, A., and Corbin, J. 1990. *Basics of Qualitative Research*. Newbury Park, CA: Sage Publications.

Sudweeks, F., and Simoff, S.J. 2005. "Leading Conversations: Communication Behaviours of Emergent Leaders in Virtual Teams," in: *38th Hawaii International Conference on System Sciences*.

Tesch, R. 1990. *Qualitative Research: Analysis Types and Software Tools*. New York: The Falmer Press.

Tyran, K.L., Tyran, C.K., and Shepherd, M. 2003. "Exploring Emerging Leadership in Virtual Teams," in *Virtual Teams That Work,* C.B. Gibson and S.G. Cohen (eds.). San Francisco, CA: Josey-Bass, pp. 183-195.

Von Hippel, E., and von Krogh, G. 2003. "Open Source Software and the 'Private-Collective' Model: Issues for Organization Science," *Organization Science* (14:2), pp. 209-223.

von Krogh, G., and Spaeth, S. 2007. "The Open Source Software Phenomenon: Characteristics That Promote Research," *Journal of Strategic Information Systems* (16:3), pp. 236-253.

von Krogh, G., Spaeth, S., and Lakhani, K.R. 2003. "Community, Joining, and Specialization in Open Source Software Innovation: A Case Study," *Research Policy* (32:7), pp. 1217-1241.

Warren, C.A.B. 2001. "Qualitative Interviewing," in *Handbook of Interview Research: Context & Method,* J. Gubrium and J. Holstein (eds.). Thousand Oaks, CA: Sage, pp. 83-101.

Wasserman, A.I. 2009. "Building a Business of Open Source Software," in *Cases in Technological Entrepreneurship,* C. Petti (ed.). Salento, Italy: Edward Elgar Publishing, pp. 107-121.

Wasserman, T. 2003. "Ietf Problem Resolution Process." Retrieved October 1, 2006, from http://www.ietf.org/Internet-drafts/draft-ietf- problem-process-01.txt

Weisband, S. 2002. "Maintaining Awareness in Distributed Team Collaboration: Implications for Leadership and Performance," in *Distributed Work,* P.J. Hinds and S. Kreisler (eds.). Cambridge, MA: MIT Press, pp. 311-333.

Weiss, R.S. 1994. *Learning from Strangers: The Art and Method of Qualitative Interview Studies*. New York: Free Press.

Wesselius, J. 2008. "The Bazaar inside the Cathedral: Business Models for Internal Markets," *IEEE Software* (25:3), pp. 60-66.

West, J., and Gallagher, S. 2006. "Challenges of Open Innovation: The Paradox of Firm Investment in Open-Source Software," *R&D Management* (36:3), pp. 319-331.

Westley, F., and Mintzberg, H. 1989. "Visionary Leadership and Strategic Management," *Strategic management journal* (10:S1), pp. 17-32.

Wildemuth, B. 1993. "Post-Positivist Research: Two Examples of Methodological Pluralism," *LIbrary Quarterly* (63), pp. 450-468.

Yoo, Y., and Alavi, M. 2004. "Emergent Leadership in Virtual Teams: What Do Emergent Leaders Do?," *Information and Organization* (14:1), pp. 27-58.

Yukl, G. 2002. *Leadership in Organizations*. Upper Saddle River, NJ: Prentice Hall.

Yukl, G., Gordon, A., and Taber, T. 2002. "A Hierarchical Taxonomy of Leadership Behavior: Integrating a Half Century of Behavior Research," *Journal of Leadership and Organizational Studies* (9:1), pp. 15-32.

Zaccaro, S.J., and Bader, P. 2003. "E-Leadership and the Challenges of Leading E-Teams: Minimizing the Bad and Maximizing the Good," *Organizational Dynamics* (31), pp. 377-387.

Zhang, S., and Fjermestad, J. 2006. "Bridging the Gap between Traditional Leadership Theories and Virtual Team Leadership," *International Journal of Technology, Policy and Management* (6:3), pp. 274-291.