

TII-3 Submitted April 2007

THE EMERGENCE OF GOVERNANCE IN AN OPEN SOURCE COMMUNITY

SIOBHÁN O'MAHONY
Assistant Professor
Harvard Business School
Baker Library 467
Boston, MA 02163
Tel: (617) 495 0875
somahony@hbs.edu

FABRIZIO FERRARO
Assistant Professor
General Management
IESE Business School
Avda. Pearson 21
08034 Barcelona
Tel: +34 650 70 28 09
fferraro@iese.edu

Executive Editors

Michael Lounsbury (Alberta) ♦ Nelson Phillips (Imperial) ♦ Andrew Hargadon (Davis)

Advisory Board

Steven R. Barley (Stanford University) ♦ James Evans (University of Chicago) ♦ Robert Friedel (University of Maryland) ♦ Raghu Garud (Penn State University) ♦ Peter Karnoe (Copenhagen Business School) ♦ Steve Maguire (McGill University) ♦ Kamal Munir (Cambridge University) ♦ Fiona Murray (MIT) ♦ Wanda Orlikowski (MIT) ♦ Jason Owen-Smith (University of Michigan) ♦ Walter W. Powell (Stanford University) ♦ Victor Seidel (Oxford University) ♦ Toby Stuart (Harvard Business School) ♦ Steve Vallas (George Mason University) ♦ Marc Ventresca (Oxford University) ♦ Jennifer Whyte (Imperial College London) ♦ Joanne Yates (MIT) ♦ Christopher Zott (INSEAD)

Technology Commercialization Centre

University of Alberta ♦ 3-23 School of Business ♦ Edmonton, AB ♦ T6G 2R6
Tel: 780-492-3375 ♦ Fax: 780-492-5709 ♦ www.bus.ualberta.ca/tcc ♦ tcc@ualberta.ca

February 28, 2007

This research has been supported by the Stanford Center for Work, Technology and Organization, the Stanford University Technology Ventures Program, the Social Science Research Council and the Harvard Business School Division of Research. The data collection and processing efforts of John Sheridan, Vikram Vijayaraghavan, Daniel Berch, Mariano Belinky, Jordi Colomer and editorial support provided by Clare Flaherty and Alyssa Razook were much appreciated. We thank Steve Barley, Beth Bechky, Bruno Cassiman, Michael Cohen, John King, Karim Lakhani, Mark Mortensen, Woody Powell, Marc Ventresca, JoAnn Yates, and the Harvard Qualitative Inductive Research (QUIET) group for their helpful comments on previous versions of this paper. We also thank Sara Rynes and our reviewers for their thoughtful and detailed comments which improved the paper. All errors or omissions are our own. Both authors contributed equally to this work.

ABSTRACT

We have a good understanding of organizing processes in bureaucratic organizations, but not in community forms. More specifically, we know little about how communities producing collective goods govern themselves. With a multi-method study of one open source software community, we found that members developed a shared basis of formal authority, but limited it with democratic mechanisms that enabled experimentation with shifting conceptions of authority over time. When members settle on a shared conception of authority, it is more expansive than their original design. This finding is reinforced with a statistical test of the predictors of leadership. By blending bureaucratic and democratic mechanisms, the governance system designed was able to evolve with the community's changing conceptions of authority.

Word count = 117

INTRODUCTION

One of the most significant problems in organizational scholarship concerns how social collectives govern, organize and coordinate the actions of individuals to achieve collective outcomes. Many classic works of organizational scholarship have grappled with this problem, leading to a variety of proposed optimal organizing forms and governance systems (Blau, 1955; Gouldner, 1954; March and Simon, 1958, Stinchcombe, 1959; Ouchi, 1979). Most attention has been devoted to bureaucratic forms with very little attention to community forms of organizing.

However, organizational forms that do not use a bureaucratic basis of authority have a long history (Coleman, 1993; 1974; 1970). Such perspectives have not received the attention they deserve (Marsden, 2005; Stern and Barley, 1996), perhaps because the institutional persistence of the dominant corporate bureaucratic form (DiMaggio and Powell, 1983; Zucker, 1977) inhibits other sources of variety (Stinchcombe, 1965). Relatively little is known about the process of organizing in communities – how social groups accomplish the critical task of coordinating the actions of multiple individuals to achieve important outcomes (Heath and Sitkin, 2001; Weick, 1979). Furthermore, understanding community forms of organizing is important to organizational theory because greater variety in organizational form increases the range of tools or solutions that society can bring to address social problems (Rao, 1998; Romanelli, 1991; Hannan and Freeman, 1989; Stinchcombe, 1965).

This research takes a step toward filling this gap by examining how a social group designs a shared basis of authority and thus, a governance system. In doing so, we heed Heath and Sitkin's advice (2001) to devote more attention to the general processes of organizing that help us understand how groups of people carry out their goals. Historically, the inability to develop a shared basis of authority has led many collectivist groups to fail (Coleman, 1980;

Swidler, 1979; Harrison, 1960; Etzioni, 1959). Thus governance in community forms not only offers a critical lens into an organization's conception of control (Fligstein, 1987, 1990) but also an indicator of how such communities can be sustained (Rothchild and Russell, 1986).

Our examination of the emergence of a governance system in an open source software community shows how a community uses a formal bureaucratic basis of authority to reinforce the community's meritocratic norms. However, this approach depends upon democratic mechanisms that not only limit that basis of authority, but allows the system to adapt with members' changing interpretation of leadership. After showing how the community introduces formal authority, we analyze conceptions of authority to understand how leadership was interpreted by community members over time. We find that while technical proficiency is an important criteria for leadership in such a group, it is not sufficient. Despite espoused preferences for 'hands-off leaders,' skill in building the organization becomes increasingly important over time. We show this through not only our analysis of espoused conceptions of authority, but by modeling the behaviors most likely to be associated with the community's leadership team. We conclude by furthering a grounded theoretical perspective of governance in communities and explore the relevance of these findings for traditional organizations.

COMMUNITY FORMS OF KNOWLEDGE SHARING AND PRODUCTION

To further this research agenda, we focus on communities involved in knowledge sharing and production. Community forms appear to be increasingly important to solving problems and sharing knowledge (van Maanen and Barley, 1984; Brown and Duguid, 1991; 2000; 2001; Hargadon and Bechky, 2006) and may be well suited for an economy that relies upon the production and diffusion of knowledge (Adler, 2001; Powell and Snellman, 2004). More

broadly, theorists now recognize that community forms can provide an alternative to market and hierarchical forms of organization and production (Adler, 2001; Powell, 1990; Bradach and Eccles, 1989; Ouchi, 1980). Yet little is known about how communities organized around production govern themselves.

Production communities typically shun bureaucratic elements such as an authoritative division of labor (Rothschild-Whitt, 1979). While there is much variance in their goals, collective forms of production tend to share a common means: namely, one that embraces democratic participation in both production and management (Rothschild and Russell, 1986). Traditional forms include kibbutzim and cooperatives (e.g. Ingram and Simons, 2000; Simons and Ingram, 1997; Kieser, 1989; Rothschild and Whitt, 1986; Rothschild and Russell, 1986; Swidler, 1979; Rothschild-Witt, 1979; Kanter, 1968) but their study has remained largely on the periphery of organizational theory's terrain (Coleman, 1993; 1974; 1970). More recently, research on communities engaged in learning, knowledge sharing has flourished.

Inside the firm, scholars have shown how members of occupational communities and communities of practice cooperate to further problem-solving, learning and skill development (Brown and Duguid, 2001; 2000; 1991; Wenger, 2000; 1998; Pickering and King, 1995; Lave and Wenger, 1991). Despite the fact that occupational communities extend outside the boundaries of an organization (Van Maanen and Barley, 1984), this construct has typically been used as a means to understand how individuals exert greater autonomy and control over their work inside organizations (Bechky, 2003; Orr, 1996). While communities of practice help individuals achieve their learning goals, they typically do so within the context of a firm's objectives (Lave and Wenger, 1991; Wenger, 1998, 2000). In both cases, community members are not engaged in managing production for its own sake, but for the benefit of their employers.

Thus, they are limited in the degree to which they can govern production. Because occupational communities and communities of practice operate within an authority structure that already exists, how such forms learn to govern themselves has not been a focus of this research.

Outside the firm, scholars have shown how online communities share information and social support (Fayard et al, 2004; Cummings, Sproull and Kiesler, 2002; Smith and Kollock, 1999; Parks and Roberts, 1997; Parks and Floyd, 1996; Rheingold, 1993). However, little empirical work has examined how such communities manage production. In modern production communities, contributors, independent of their employment context, voluntarily collaborate to create goods or services for either public or private benefit (von Hippel and von Krogh, 2003). Members are typically geographically dispersed and depend upon the Internet for a means of communication and coordination (Kollock, 1998). Within the last decade, online production communities have begun producing information goods such as scientific knowledge,¹ art,² general knowledge,³ and software.

Open source software communities are one example most often recognized by organizational theorists (von Hippel, 2005; von Hippel and von Krogh, 2003; Lee and Cole, 2003; Moon and Sproull, 2002; Kogut and Meitu, 2001). Scholarly analysis of open source software communities has examined why individuals are likely to contribute to such efforts (Lakhani and Wolf, 2005; Dalle and Jullien, 2003; Hertel, Niedner & Herrman, 2003; Hars and Ou, 2002; Lerner and Tirole, 2002), status dynamics (Stewart, 2005) and member contribution patterns (Shah, 2005; Mockus, Fielding and Herbsleb, 2002). We have learned that open source communities create informal and formal social structures to manage membership and joining

¹ www.sciencecommons.org

² www.creativecommons.org, ccmixtr.org, flickr.org

³ www.wikipedia.org

processes (von Krogh, Spaeth and Lakhani, 2003; O'Mahony and Ferraro forthcoming), but little has been done to understand how these projects are governed (Shah, 2006, a recent exception).

While these communities may be a fledgling form of organization, scholars and practitioners alike see production communities as increasingly important to an information and knowledge based economy (Powell and Snellman, 2004; Seidel and Stewart, 2003; Williams and Cothrel, 2000; Sawhney and Prandelli, 2000; Armstrong and Hagell, 1996). Production communities differ from prior research on community forms in three ways that make them theoretically distinct and ripe for study. First, unlike communities of practice or occupational communities, they do not share a common employer or workplace. Second, unlike online communities, production communities must integrate individual contributions into a common pool, which can heighten interdependencies (e.g. Thompson, 1967) and the need for coordination mechanisms. Third, production communities often 'own' the output of their work and work toward collective goals outside of the scope of their employer (O'Mahony, 2003).

Taken together, these three distinctions suggest that production communities need a way to manage their interdependence in order to achieve a common goal. Yet, without the benefit of markets or hierarchies, they may have fewer resources with which to do so. Scholarly conceptions of community forms as 'ideal types' tend to overweight the roles that norms, trust, mutuality and reciprocity play in organizing community activities without examining how people actually coordinate their work to achieve collective ends (Adler, 2001). To move beyond ideal types, we take seriously the complicated nature of directing individual efforts toward a common goal without the benefit of contractual or hierarchical reinforcement. In any organization, there is a constant tension between fulfilling individual goals and integrating them with a common

goal (March and Simon, 1958; Ouchi, 1979). A focus on governance allows us to explore how community forms resolve this tension.

GOVERNANCE IN COMMUNITY FORMS

Before an organization can have a form of governance, it must establish a shared basis of authority (Etzioni, 1959). Organizations without a consensual basis of authority lack an important condition necessary for their survival (Coleman, 1980; Harrison, 1960; Etzioni, 1959). Organizations with directly democratic forms of participation do not tend to scale well and are noted for their difficulty managing complexity and decision-making – all of which can hasten their demise (Whyte and Whyte, 1988; Rothschild and Russell, 1986; Rothschild and Whitt, 1986; Johnson and Whyte, 1977). The need to coordinate interdependent member activities and integrate member contributions in a production context is likely to exacerbate the need for a shared basis of authority.

However, modern collective forms of production do not tend to rely on any single one of the three bases of authority theorized by Weber (tradition, law, or charisma) (1978). These forms are created with few traditions to guide them and so do not inherit a basis of authority steeped in tradition. Since there is no authoritative division of labor, collective production communities do not rely upon what Weber would call a legally rational basis of authority rooted in position. And, finally, since many contributors may not have met in person, the ability to develop a basis of authority that rests on charisma (while possible) is limited. Unlike traditional conceptions of community rooted in a common place, modern production communities are more likely to be distributed than they are localized (Wellman et al, 1996; Wellman and Gulia, 1999).

Weber recognized that ‘anti-authoritarian’ systems that placed a high value on individual autonomy existed (1978), but “he did not systematically analyze the problems of authority and power peculiar to these groups” (Harrison, 1960: 233; also Satow, 1975 and Willer, 1967).

However, developing some form of authority is a particular problem for voluntary social groups that can lead to the compromise of their goals (Harrison, 1960).

The ideology of voluntary social groups in America tends to be anti-authoritarian. The constituency of these groups is distrustful of centralization and further rationalization of their organizations. However, to achieve the imperative goals of these voluntary associations, bureaucracy is necessary, social tension increases, and the problems of authority and power become increasingly acute (Harrison, 1960: 232)

To retain the interest and commitment of voluntary members, any form of authority introduced must simultaneously preserve democracy and accountability to its members. To achieve the efficiencies for which bureaucracy is known, some form of rationalization is necessary (e.g. Chen and O’Mahony, 2007). How communities resolve this conflict is under explored, largely because of unresolved theoretical issues in the conceptualization of bureaucracy.

In a bureaucratic system, a legal and rationally based positional authority decouples the authority of a person from their position to prevent incumbent patrimony or favoritism (Weber, 1978). In Weber’s eyes, meritocracy was an inevitable outcome of bureaucratic rule. Those with more technical competence were rewarded with positional authority – that is the essence of the “exercise of control on the basis of knowledge” (Weber, 1978). Parsons (1947) and Gouldner (1954) were among the first to note that Weber’s notion of bureaucracy could lead to contradictory outcomes – arguing that positional authority and technical competence could be de-coupled and lead to bureaucracy without meritocracy.

Adler and Borys (1996: 62) attribute the field's conflicted approach to bureaucracy back to this source of ambiguity and propose a way to help reconcile conflicting evidence on the impact of bureaucratic rule. They argue that bureaucratic rules can be applied to either help people in their job (enabling) or as a mean to control (coercive). They predict that enabling bureaucracies are better suited for knowledge creation, while autocratic bureaucracies can only perform in predictable environments. Adler extended this line of thinking by exploring the contours of community forms, tentatively concluding that community forms of knowledge creation are likely to grow only if community norms are balanced by "hierarchical rules to ensure stability and equity" (2001: 228, also Adler and Heckscher, 2006). This suggests that community forms must blend some type of positional authority with more participatory and democratic means. However, little empirical work has extended this theoretical framing.

Recent scholarship on open source communities suggests that any governance system introduced must be meritocratic in order to attract high quality contributions from voluntary members (Lee and Cole, 2003; Moon and Sproull, 2002; Kogut and Meitu, 2001). By rewarding merit with greater status, responsibility, or opportunities to enhance their own development (Stewart, 2005; von Krogh, Spaeth and Lakhani, 2003), production communities can satisfy a contributor's need for recognition and reward in ways that their work lives may not (e.g. Drucker, 2001; DiMaggio and Anheier, 1990). Although it is widely recognized that successful open source projects often have strong leaders (Moon and Sproull, 2002; Mockus, Fielding and Herbsleb, 2005), few have examined the roots of such governance systems.

To further a grounded theoretical understanding of how communal forms organize, we examine how one open source community designed and implemented a governance system over a thirteen year period with both qualitative and quantitative data. We identify four distinct

phases: De facto Governance, Designing Governance, Implementing Governance and Stabilizing Governance. We find that the failure of autocratic rule in the De facto Governance phase spurred the design of a formal governance structure that included a positional basis of authority. To reinforce the community's meritocratic norms, this bureaucratic basis of authority was simultaneously limited with a directly democratic mode of governance. However, the role of a leader in the community remained open to interpretation. Thus, we evaluated espoused conceptions of authority in the Implementing Governance and Stabilizing Governance phases to learn how members interpreted the leadership roles they created. Finally, to move beyond analysis of the rhetoric of the community, we investigated what leadership *behaviors* became most valued by members of the community in their final phase of governance.

This research provides two distinct theoretical contributions. First, we clarify earlier speculation about how production communities organize (e.g. Adler, 2001; Adler and Borys, 1996) and find evidence of a limited form of bureaucracy that is more enabling than it is coercive. The simultaneous introduction of democratic and bureaucratic practices not only limits the reach of bureaucracy, it also allows members to experiment with varying interpretations of authority. These findings provide insight as to how community forms develop a shared basis of authority and a governance model – a fundamental question for organizational theory that has been long under addressed (Coleman, 1980; Harrison, 1960; Satow, 1975). Second, we show that even in a community of open source programmers that espouses the value of technical contributions above all else, members' conceptions of leadership change over time to increasingly value organization building contributions. Democratic mechanisms enable the community's governance system to adapt as members learn how to interpret leadership and

authority in a community context. This suggests an evolving and context-dependent notion of meritocracy and that democratic mechanisms serve an important adaptive function.

PART I: INDUCTIVE APPROACH

This research was initially guided by an inductive approach using ethnographic methods. An inductive approach is particularly apt for examining phenomena that are emergent or poorly understood (Strauss and Corbin, 1990), allows room for the unanticipated and is most suitable for grounded theory building (Edmondson and McManus, 2006) – the goal of this research. In Part I, we used ethnographic and archival data to explore how the governance system designed by this community operated over a 13 year period. We explore different conceptions of leadership that invoked varying degrees of authority over time. In Part II, with quantitative data gathered from the same setting, we pursue a deductive approach to examine the behaviors that were most important to community members in filling the leadership positions they created.

Research Setting. We used theoretical sampling (Strauss and Corbin, 1990; Glaser and Strauss, 1967/99) to select an open source community that had two features: a long tenure and leadership turnover to allow in depth exploration of governance. Successful leadership turnover suggests that leadership positions have become institutionalized or decoupled from the founder of the project. Of several open source software communities studied in the first author's prior research, Debian was selected for further analysis because its longevity (it was formed in 1993) exposed it to governance issues that may occur only over time. Preliminary fieldwork also revealed that the Debian community had, unlike other open source projects previously examined (e.g. Raymond, 1999; Moon and Sproull, 2002; Lee and Cole, 2003), experienced frequent

leadership turnover since the founder left the project in 1996. As one informant explained, this was perceived to be a strength of the project and contributed to its longevity.

I have complete confidence in the Debian Project to find its own path to the future. Four different Project Leaders over the past five years have done nothing to shake that confidence. We have shown ourselves quite capable of thriving despite changes in leadership, and without leaders who felt a strong desire to change Debian's vision.

As of this writing, nine different leaders have led Debian over 13 years, suggesting that the Debian community had a mode of governance independent of its founder.

Debian is a free operating system that uses the Linux kernel developed by Linus Torvalds - it is a Linux *distribution*. An operating system is the set of basic programs and utilities that make computers run. The kernel is the most fundamental program on the computer that manages demands on system resources: It allows you to run multiple programs simultaneously.

According to industry analysts, Debian has 25% of the market for Linux distributions, second only to the Red Hat distribution (produced by Red Hat, a publicly traded company) (Netcraft, 2005). Over 1,000 Debian community members in over 40 countries contribute to the development of the Debian Linux distribution (which comprises over 9,000 packages of code). Developers who contribute to the community consider themselves to be part of "an association or a club, much like your local LUG (Linux User Group) or Rotary, with the principle exception being that we hardly ever meet face to face."⁴ While the Debian community does not sell the code they produce, over 150 vendors commercially distribute the project's software. In return, Debian receives corporate support and equipment donations from at least ten firms, two of which are Fortune 500 firms.

⁴ From: Step 2: Debian New Maintainer process located at: <http://ukdebian.mirror.anlx.net/devel/join/nm-step2>.

PART I: INDUCTIVE METHODS

Data on the Debian Linux community were collected as part of a larger ethnographic study of the open source community. The first author spent over 90 hours observing and meeting informants from the free software and open source communities at 35 different events, including project meetings, user group meetings and conferences, between 2000 and 2005. She also conducted 81 semi-structured interviews of leaders within the open source and free software communities with a focus on membership, sponsorship, decision-making, and governance.

The data used for this paper are displayed in Table 1. This included 47 interviews with Debian developers (four former leaders, 14 active contributors and 30 user contributors). These data were supplemented by observations and analysis of 34 leader candidate platforms, 4 election debates, 32 meeting notes, 5 general resolutions, and 17,317 postings to the project mailing list. We searched mailing list data to identify topics related to leadership and governance and collected project data from online archives. These included the project's Constitution, Policy Manual, Reference Manual, Social Contract, Charter, and bylaws which were tapped to understand community operations.

Ethnographic methods guided the examination of interview and observation data and project documents with a special effort to capture the perspectives, actions, and interactions of all actors involved. Interviews were transcribed and, with other project data, coded and analyzed using Atlas TI software, a qualitative coding application to identify common organizing practices used within the Debian Linux community. Triangulation among multiple sources of evidence provided greater depth and accuracy (Yin, 1994; Campbell and Fiske, 1959) (see also Table 2).

In our first round of coding, we independently identified four phases to the evolution of Debian's governance system: 1) De facto Governance (1993 – 1996); 2) Designing Governance

(1997 – 1999); 3) Implementing Governance (1999 – 2003); and 4) Stabilizing Governance (2003 – 2006). We identified these phases by noting both the presence of defining governance events (Table 2) and by tracing shifts in conceptions of authority (Figure 1). Once we understood the elements of governance that the community introduced, we embarked on a second round of analysis to understand how this system was interpreted by members.

To learn how potential leaders within the system conceptualized their role, we independently coded leadership platforms and election debates from an 8 year period (1999 – 2006) to determine how conceptions of authority evolved. We included data available data from de facto leaders from the period 1993 – 1998. After resolving minor disagreement in classification, we identified five new conceptions of leadership that varied in their degree of authority and in their focus on organizational versus technical concerns: 1) hands-off leader; 2) technical manager; 3) visionary leader; 4) organization builder; 5) organizational leader. We asked an external researcher to classify the platforms independently and compared this classification with ours. The independent researcher rated the leadership platforms on the basis of the descriptions articulated in this section of the paper. Comparing this researcher's codes with ours, we calculated our inter-rater reliability using Cohen's Kappa, and obtained very satisfactory results (0.80 with an 84% agreement rate). After considering the relevance of the disagreements encountered, we deemed it safe to proceed with our original coding.

PART I: INDUCTIVE RESULTS

In this section, we analyze our qualitative data to show how the community transitioned from a de facto governance model to one that integrated positional authority with directly democratic means. We then trace how different conceptions of authority were subsequently

interpreted by community members. In Part II of the analysis, we use quantitative data from the first year (2003) of the final phase (Stabilizing Governance) to predict which developers were mostly likely to become a member of the leadership team.

Phase I: De facto Governance (1993 – 1996). We found that for its first five years, the Debian Project operated reasonably well without a formal means to represent its contributors in project governance. The founder of the project considered contributors' opinions as they were expressed, but had the final say on project decisions. When the founder left the project to pursue personal interests, he informally passed the leadership role to one of his trusted lieutenants who maintained a very active role on the project. The decision to transfer authority was made unilaterally and the second leader continued without any formal governance process to resolve disputes or provide contributing members with formal representation. However, he did hold a more expansive conception of the leadership role than did the founder. We found that subsequent misinterpretation of the appropriate role of a leader in the community led to the failure of autocratic leadership and the construction and delimitation of positional authority.

Without agreement on the leader's basis of authority (or the 'right to lead') and with an expansion of organization building activities, members resisted what some characterized as 'autocratic' leadership. Aware of the problem, the second leader recalled, "there was no formal structure whatsoever at that time. You had to lead where people would follow otherwise they would just walk off." When the second leader began referring to his role as "President" and initiated a series of organization building projects to manage Debian's growing scale, community members balked at his actions. By assuming the title of President, the second leader had essentially assigned himself positional authority without developing a shared basis for it.

The project leader felt this strain and, to both enhance his legitimacy with project members and develop a shared basis of authority, proposed that the community elect a board of directors. In a message to the community, he explained that “this election relieves me of the (often quite awkward) position of being the only person with any real authority over Debian”. His intent was to create “a leadership team for Debian that can survive the departure of any team member.” In the months that followed, members lost faith in his leadership and asked him to withdraw from the project’s first election. “A number of developers convinced me that it was time for a leadership change, and I thus withdrew from the project leader election.” While the second leader had initiated the design of a more formal means by which project members could achieve representation in governance, he may have introduced too many changes too quickly for project members concerned about a potential concentration of power. The recall of the second de facto leader led directly to a phase where members designed a governance system that could represent their interests.

Phase II: Designing Governance (1997 – 1999). We found that community members’ shared experience in recalling the second leader inspired the design of a governance system that could operate independent of any one person’s conception of leadership. The third leader to assume the title did so only by vowing to be less directive than the second and leading the collective drafting of a Constitution to formalize leadership roles, rights and responsibilities. As one community member recalled, “[the third project leader] said we should formalize things and come up with a structure that could address the problems faced by a Debian Project Leader...the project leader was looking at the way the project was going...and saying, no way is this going to work and we have to do something.” The third leader led a debate over the mailing list, drafting and revising the Constitution over the course of a year which was ratified by 357 developers. As

the third leader explained, “we basically ratified the constitution using itself. That was sort of a test case as well, which went pretty well, so we use it for project leader elections.” The governance system designed embraced two important and potentially contradictory elements: 1) formal positional authority and 2) limitation of that authority through democratic means.

Developing positional authority. Positional authority is rooted in a defined role, decoupled from a person’s individual characteristics – it separates the personal from the official and provides a legal-rational basis of authority (Weber, 1978; Coleman, 1980; Merton, 1940). The Constitution outlines positional authority by specifying the rights and limitations of contributing members and positions of power such as the Debian Project Leader (DPL). As one member explained, “We had this DPL project leader but we never really said what he could or could not do. He had been doing things, so it [the Constitution] kind of codified what he had been doing or what we wanted to be doing and what we did not want him to be doing.” With the Constitution in place, the project held their first election in January 1998, introducing the role of a Debian Project Leader (DPL) (as opposed to ‘President’). However, we found that Debian members were only interested in supporting a positional basis of authority if this role was also limited in ways that facilitated democratic control by the rest of the community.

Limiting positional authority through democratic means. We found that positional authority, once created, was limited in four ways that preserved democratic rule. First, the Constitution requires those in positional power to act in deference to the wishes of the collective. Project leaders must “make decisions which are consistent with the consensus of the opinions of the developers” (Debian Constitution). In practice, this tenant was emphasized often and the DPL’s ability to make unilateral decisions explicitly questioned. For example, years later, a

participant in the 2006 leadership debate, Filippo⁵ emphasized that “technical decisions shouldn’t be made by the DPL, period.” In response, another participant, Brent elaborated that “the DPL’s role is to lead discussion within Debian, and technical discussions are what makes Debian great. That’s different from actually making the final decision though, which generally isn’t the DPL’s place.” The concept of leader is thus based on consensus building as opposed to autocratic rule and the power of a leader only comes from his or her deference to democratic values.

Second, a Debian Project Leader is subject to the same rules as any member – they are not entitled to special privileges. In practice, this means that the DPL has no more power or authority over what goes into Debian’s code base than any other member. The authority that comes with the role is just enough to encourage consensus building. As Sam explained, “the DPL in this case is just a normal developer, with a higher soap-box. People are more likely to listen to him and will perhaps decide to follow his attempts at moderation.” Just like a developer, the only way a DPL can enact a policy change is through a General Resolution which must be approved by a majority of members. Developers vying for the leadership role realized that they thus needed a rationale as to why they wanted the position. For if a DPL did not have more authority than any other member, why would one want to pursue it? A winning candidate in 2006, admitted as much when exploring this question: “I don’t believe this [the DPL position] would dramatically alter my work for Debian, rather it would allow me a bit more flexibility in achieving the goals I just mentioned and thus let me give them a higher priority.”

Third, as a failsafe measure, the authority of the leader can be recalled by the collective through a General Resolution process. Any member has the right to propose a General Resolution which may counter a leader’s actions. As one project leader candidate noted, “the

⁵ All names of informants are disguised.

DPL is not particularly empowered to stand against the will of the developers, since they can overrule anything he or she does with a General Resolution [GR]. However, a GR is a weighty process and a DPL who consistently acts in discord with the will of the developers can do damage simply by wasting the Project's time." Our analysis of the history of use of General Resolutions noted that they were used very rarely (5 times in 8 years).

The fourth way that positional authority is constrained by democratic rule is through a countervailing source of authority. For example, project-wide decision-making power is split between the DPL and a technical committee that is empowered to "decide any technical matter where developers' jurisdictions overlap." The technical committee can not introduce new proposals but has the authority to resolve disputes. In order to overrule a developer, a supermajority (three-fourths) of the committee must agree – in effect mandating consensus building. As one long time member explained, the technical committee "actually has more power than the DPL in some regards" and further limits positional authority.

The technical committee is able to say, we have looked at both sides of the discussion, if we do it your way, every single maintainer in Debian has to recompile his packages, we just don't think that is a good idea. So they will say, you "know they will vote against you." And that is what it really is for, is when the DPL can't really say it, when the rest of Debian can't decide or wont decide or whatever, the DPL says fine, we will decide with the technical committee.

The DPL can call upon the technical committee to help reach a decision but can not force a decision, thus limiting a leader's ability to unilaterally settle disputes.

By approving the Constitution, project members developed a shared basis of positional authority. The four mechanisms described above limited the reach of bureaucratic rule by reinforcing democratic rule by community members as a whole. As the next two phases will

show, the integration of democratic mechanisms also ensured that the governance system, once implemented, could evolve with the wishes of the collective.

Phase III: Implementing Governance (1999 – 2003). In 1999, with the Constitution ratified, Debian developers began electing a project leader for one-year terms - effectively initiating a phase of experimentation with the leader role. Candidates nominated themselves and posted a leadership platform on the website outlining their goals for the project nine weeks before the end of the term. Platforms were debated in a three week polling period that provided ample time for members from all time zones to participate. A long polling period was important so that volunteers who did not work on the project every day, or even every week, had a chance to vote. As Table 3 demonstrates, participation in annual leader elections hovered from 51-60%, with a steeper decline (43%) in later years.

While most developers happily went about their work without too much thought to governance, a leader could now be singled out when the community ran into difficult or ambiguous situations and our data show that members valued this. The year following the passage of the Constitution, one DPL candidate noted that “it introduces a bit of official rules and politics, but I think it will allow us to work as the sort of organized anarchy that we have always used while adding some much needed safety nets.” Still, a governance system designed on paper had to be put into practice - Debian members had to interpret exactly what this new role would entail. Our ethnographic data indicated that our informants varied a great deal on their interpretation of the project leader role. To provide better traction on how the leader role was interpreted, we turn to our analysis of the leader candidate platforms.

Given the importance the community placed on delimiting positional authority in Phase II, Designing Governance, we expected community members to prefer a ‘*Hands-off*’ style of

leadership. However, our data suggested otherwise – for a ‘*Hands-off*’ conception of leadership never received a majority vote from community members. Table 4 indicates how often each conception of authority was represented in each election year and which one won the majority vote⁶. We found that a *Hands-off* conception of authority was characterized by a belief in the self-organizing ability of the community and the danger of expanding the authority of the Debian Project Leader. A leader candidate in 1999 stated that “he didn’t see the project as something that can be steered or directed” and submitted no “specific plan” in his leadership platform. Another candidate in 2001 stated that the role of the DPL was “important but limited, with the majority of the power to act rightfully left in the hands of developers.” Despite the community’s interest in limiting positional authority, no leader won an election with such a platform.

The *Technical Manager* conception of authority focused on the coordination of technical work. These leader candidates were concerned with shortening and enhancing the predictability of release cycles. Technical Managers focused on engineering management skills and tools to remove neglected packages, reduce bugs and improve the software development process. Accomplishing these goals would require more authority than a *Hands-off* conception of authority. For example, a candidate in the 1999 elections wanted to “to create a stable and bug free distribution for our users” and devoted most of his election platform to specific technical issues, arguing that “once there is a deep freeze, a group should be assigned to assess the bugs affecting the current frozen distribution.”⁷ Technical managers never won an election.

⁶ Although the first three leaders did not create platforms nor run for office, we coded their conceptions of authority based on our interview and mailing list data and included that data in Table 4. Inclusion of these data allow for a comparison of the evolution of interpretation of authority over the project’s life time (1993 – 2006).

⁷ Code that is deep freeze can not have new features added to it. This leader wanted to stop the addition of new features (which was more interesting to volunteer contributors) and assign people to work on fixing new bugs (which was less interesting and more mundane work) so that the release could be produced.

Visionary Leaders focused on the need for a cohesive project vision, without advocating specific technical or organizational building projects. These leader candidates argued that the community needed to define a clear vision for the future. However, such leaders suggested that it was their prerogative to define such a vision, implicitly questioning the community's ability to collectively define a vision. Visionary leaders felt comfortable taking unilateral initiatives consistent with their vision, without necessarily making them a collective effort. For example, a winning candidate in 2002, stated: "Debian has achieved some wonderful results, but whether we will continue to improve is largely a function of how motivated we are. My experience leading volunteer efforts suggests to me that the best motivators [have] a strong shared vision which each participant can feel connected to, and processes that allow contributors to see results from their efforts." Leaders with a visionary platform won elections in 1999 and 2002 but not after 2002.

Some candidates espoused a conception of authority that emphasized their skills as an *Organization Builder*, and their ability to develop organizational structures and processes to improve the community as a whole. To distinguish from the *Technical Manager* approach, these projects were more organizational than they were technical. For example, an unsuccessful candidate in the 2001 election, proposed culling developers from the project who were "no longer really contributing." "I propose that we experiment with and ultimately apply, automated tools for tracking package and developer activity, and act accordingly." This proposal would require explicit monitoring of members and was never implemented. Implementing such changes would obviously expand the DPL's authority to new domains. Candidates with organization building platforms won in 2000 and 2001.

Phase IV: Stabilizing Governance (2003 – 2006). Variation in conceptions of leadership that prevailed throughout the third phase began to reach settlement with the

emergence of a new conception of authority, *Organizational Leader*, in 2003 (Figure 1, Table 4). *Organizational Leader* platforms connected both organizational building and technical activities to the project's vision and goals. While *Organization Builders* often included extensive lists of process improvement projects, *Organizational Leaders* focused on the challenges of motivating volunteers and aligning their interests with those of the community. These leader candidates were more likely to recognize the importance of communication, culture, and relational skills relative to procedural improvements. One unsuccessful candidate in 2005 argued that “the overall goal is to make Debian a fun and rewarding context to work and spend time in. so much so that one misses and long for it when one can't be or work on in it.” To achieve this goal, this candidate recommended “small teams [...], a more friendly and helpful environment [...], making people aware of their leadership role [...and] having more frequent real-life meetings.”

Organizational Leaders openly acknowledged that although there were many things they would like to change, their ability to implement them was limited. They thus recognized that powers of persuasion would be critical, given the limited authority associated with the position. For example, an unsuccessful candidate in 2006 said that:

I must also acknowledge the fact that the DPL does not have the power to simply impose changes as he/she sees fit. The best that the DPL can do here is to encourage us to improve, sometimes by discussion and debate and sometimes by leading by example... My own priority would be to fix some of the social issues first, as these are most pressing and in my opinion the most likely to cause long-term damage to the project.

Like *Visionary Leaders*, *Organizational Leaders* were aware of the need to develop a compelling vision for the community. However, they did not think it was their job to do so alone. Rather, it was the job of the leader to corral and organize members around a common vision.

Organizational Leaders were also more cognizant of the limitations of the DPL's authority to actually implement organization building projects than were *Visionary Leaders*. This may reflect the community's learning of what a leader could actually accomplish in this environment as well

as a growing recognition of the importance of other member's motivations to contribute to the project. However, the amount of authority needed to carry out such a platform was still high. The last four elections (2003-2006) were won by *Organizational Leaders* and represent a limited consensus on the degree of authority appropriate for a project leader.

To verify whether a general trend from technical to organizational concerns was present across these conceptions of leadership, we conducted a second round of coding leadership platforms, focusing on raw word counts of organizational and technical issues. This analysis also confirmed that, over time, project leader candidates focused their leadership platforms increasingly on organizational issues as opposed to technical issues (Figure 2). For example, in 2006, 73% of the text of a platform, on average, was devoted to organizational issues, compared to only 37% in 1999. Appendix A provides details on our approach.

While the Debian community achieved some shared consensus on the preferred conception of leadership, this does not imply that the community was free from strife or conflict (e.g. Coleman, 2005). Our claim is merely that in Phase IV, when conflict over a leader's authority occurred, community members resolved these issues within the new governance framework that had been created and debated in Phases II and III. As evidence of settlement, in 2006, a group of developers used the General Resolution process to recall the project leader when they became unsatisfied with his management of the boundaries of the project. This was the first leader recall in ten years, but now a formal process existed to handle challenges to a leader's authority. The recall did not pass the General Resolution process, but provides evidence of members' ability and willingness to work within the governance system they designed.

Conclusion. Under a de facto governance system (1993-1997), Debian members had little shared basis of authority and autocratic leadership failed. When Debian members entered a

phase of designing governance (1997 – 1999), they focused on both developing and limiting a positional basis of authority that could be checked by democratic means. However, translating governance design into practice (1999 – 2003) introduced much interpretation and variation as to the level of authority appropriate for a project leader – for democracy encouraged variety. During this time, we identified five different active conceptions of leadership. Our multi-method approach revealed however, that our informants’ espoused preferences for a “hands-off leader” in the design phase was not supported by later systematic evaluation of leadership data.

After four years of implementation, a new conception of authority emerged in 2003 - *Organizational Leadership*. This year marked a transition to a shared conception of authority and a period of stabilized governance which survived a crucial test – the attempted recall of a leader. As Selznick explained, “giving life to a constitution is partly a matter of achieving general consensus regarding proper ways of winning power and making laws” (1957: 6). By the end of 2006, the Debian community had achieved this. With statistical techniques, we now explore the actual behaviors associated with becoming a leader in this community at the start (2003) of the final phase, Stabilizing Governance.

PART II: DEDUCTIVE APPROACH

From our qualitative data, it would appear that developers engaged in organizational building would be more likely to assume leadership than those who were more hands-off or spent more time on technical issues. To test this prediction, we modeled the results of one leader election four years after the system was designed (in 2003) with statistical analysis. This approach allowed us to move beyond analysis of discursive and rhetorical strategies used by leader candidates, to understand the *behaviors* that were important to community member’s conception of leadership.

Technical Contribution. The literature on open source communities tends to assume that these communities operate in a meritocratic manner (e.g. Lee and Cole, 2003; Kogut and Meitu, 2001; Raymond, 1999). Thus, positions of authority should be allocated according to merit. However, it is not clear what merit really means in this context – technical contributions or organization building? Our interview data suggested that informants were aware of a status hierarchy based on technical contribution and observed the movement of people through it. One developer, commenting on the 2001 election, noted that “I have seen a lot of developers go from nobodies to being absolutely huge on the project. You know for instance, [Dan Evans], one of the guys who is running for the DPL right now.”

The same informant who commented on Dan’s evolution from a “nobody” to “becoming huge” also noted that “there is no hierarchy to really say you know I am bigger than you so I will win. And it is a good thing and a bad thing. A lot of Debian is I say a meritocracy. You know if my code is better than your code, they will use it. Right?” Our informants’ interpretation of meritocracy implied that the quality of one’s code would be predictive of leadership. Thus, we propose that contributors with greater technical contributions will be more likely to become leaders. To operationalize this hypothesis, we explicitly considered both the amount of technical contributions as well as the impact of a developer’s contributions on the community.

Hypothesis 1a: The greater the amount of a community member’s technical contribution, the higher the probability that a community member will become a leader.

Hypothesis 1b: The greater the impact of a community member’s technical contribution, the higher the probability that a community member will become a leader.

Organizational Building and Leadership. Our qualitative data suggested that developers valued different strengths in a leader. While some leadership candidates thought it was a “social position,” others thought it was a “merit badge rather than a position of trust.” In the 2002

election, one candidate noted that neither technical expertise nor social capital was sufficient: “The best candidate will not necessarily be the most popular or technically accomplished candidate, but the candidate who has the strongest grasp of what it will require to lead the Debian Project effectively.” This suggested that organization building might be as important as technical contributions but did not specify what activities would enable people to grasp “what it takes.” Our analysis of the organizational building and organizational leadership conceptions of authority in Part I showed that such leaders tended to focus on enhancing communication and face to face interactions to better coordinate individual efforts within the community.

One former leader explained the importance of knowing people and their needs in order to foster internal coordination on the project. “Internally it’s like coordination, and checking that everything is working well. So, for example, those people who were playing important roles, I’ve asked them, “What can I do for you?” For example the security team they are quite overloaded, so I was trying to find some people for them..... asking what is up? What can I do?” Although there are many facets to organizational building, we realized that in a distributed global community, it would be impossible to do any organizational building without posting messages online or meeting people face to face: these are necessary conditions for organization building.

Three recent studies of the emergence of informal leaders in online communities found that leaders post more messages than non-leaders (Cassell et al, 2005; Misiolek and Heckman, 2005; Yoo and Alavi, 2004). Furthermore, the degree of offline interactions are likely to influence the amount of effort people put into building online communities (Butler et al, in press). In their study of Internet Engineering Task Force (IETF) committees, Fleming and Waguespack (forthcoming) discovered that after a certain level of technical contribution, collaborations with high status contributors became a more important predictor of leadership.

Thus, we propose that community members who communicate more with each other in both offline and online forums are more likely to be on the leadership team.

Hypothesis 2a: The more a community member participates in online discussions, the higher the probability that a community member will become a leader.

Hypothesis 2b: The more people in the community one meets face to face, the higher the probability that a community member will become a leader.

If we find that Hypotheses 1a and 1b are supported, and Hypotheses 2a and 2b are not supported, then we could assume that community members favored a hands-off or purely technical management approach to leadership. This would contradict our findings on espoused conceptions of leadership in Part I. If we find that Hypotheses 2a and 2b are supported, then we will have more systematic confirmatory evidence of members' shifting concern for organizational building as a legitimate expansion of a project leader's authority.

PART II: DEDUCTIVE METHODS

Our field research helped us identify data sources that could be used to predict the leadership behaviors that were important to the community. These data sources included the project's organizational chart, developer database, bug tracking database, package popularity database, and identity authentication database. We collected data from these different sources and integrated them in one database, merging the observations on the basis of developers' names and email addresses. With these data, we estimated a logit regression model to predict who became a member of the community's leadership team in 2003. Since the dependent variable was dichotomous, a logistic regression was appropriate (Long 1997)⁸. Next, we explain the how we operationalized each variable.

⁸ We also estimated the full model with the rare event logit procedure developed by King and Zeng (1999) to deal with cases in which the dependent variable assumes the value 1 only in a small number of cases (less than 1%). The

Leadership Team. In measuring our dependent variable, we consider not only the position of Project Leader, but the community's full leadership team which included the project leader, project secretary, software release coordinator, developer accounts manager, and members of the technical committee. Release coordinators and developer accounts manager are volunteers for indefinite appointments. Release coordinators shepherd the project to closure and developer account managers are responsible for allocating account rights to the project's code repository. As explained earlier, the technical committee (selected by the project leader) has the constitutional right to resolve technical disputes. We viewed these positions as providing critical leadership for the community and our informants concurred. We coded these positions for the year 2003, the latest year for which a full set of data was available. In 2003, 831 developers were recognized as full fledged members of the community, but we could only find complete data on 815 developers. Rather than imputing values for the missing data, we tested our hypotheses on these 815 observations. We also compared the missing observations with the ones used in the analysis and found that only one of the developers in the leadership team had missing data. The developers with missing data were primarily non-leaders and, on average, maintained fewer packages, maintained packages with less impact, posted fewer messages online and had met fewer developers than developers with complete data. Given the direction of our hypotheses, the missing data do not introduce meaningful bias in the analysis since, if anything, they make our estimates more conservative.

Independent Variables. We operationalized technical contribution and organizational building activities with two different measures for each construct. Technical contribution was measured in terms of the number of software packages maintained by each developer and by the

model is significant and there is no substantial difference with the results we report with a traditional logistic regression in table 5.

number of people that used these packages. Together, these measures capture both the quantity of a developer's contribution and their impact on the community. We measured organizational building activities by the amount of communication developers engaged in on the project's central mailing list, and their degree centrality by the number of other developers the focal developer has ever met face to face.

Amount of Technical Contribution. As a measure of each developer's technical contribution to the project, we collected data from the project's bug tracking database on the number of software packages each developer maintained in 2002. A package is a discrete unit of code that can be maintained independently from the rest of the operating system but has a standardized interface that allows integration with other packages. To maintain a package is to manage the receipt and review of code contributions from other contributors and to "package" these smaller contributions into a discrete module.

Impact of Technical Contribution. To measure the impact of a developer's technical contribution, we measured the "popularity" of the packages a developer maintained. Since early 2001, Debian users could install a "popularity-contest package" that automatically calculates the number of people that download, install and use a particular package. The popularity count is akin to a citation count in academic circles: it signifies how many people in the community use or rely upon your work. We computed the sum of the votes for all the packages maintained by developers as a measure of the criticality of their contributions for others. Although these data might be biased by the fact that not all developers had installed the "pop-con package" as of 2003, the sample of data we obtained provides a good proxy for package use. Since 2004, the results of the popularity-contest have been used to determine which packages to include in Debian distribution CDs as well as to determine which packages to eliminate from the archives.

We tested to see whether there was a substantial difference between the results of the popularity contest in 2003 and more recent years (2004 and 2005), and found that the results were highly correlated, supporting our claim that even for 2003, this was a robust measure of a package's usage by others.

Online Communication. Since all members are globally distributed, on-line mailing lists are the key infrastructure that enables communication and coordination in the Debian project. On the mailing lists, developers discuss changes to the code, coordination issues, and the future direction of the project. Debian has over 100 different mailing lists, focused on different sub-groups of developers, and specific technical issues. To measure project-wide on-line communication we focused on one mailing list, *debian-devel*, to which all developers subscribe. We downloaded all the postings to this mailing list and measured developers' number of postings as a first proxy of organization building. Although discussion on *debian-devel* can also address technical issues, this mailing list is the one where most of the critical organizational discussions in the community were initiated, mainly because it has the largest audience. Therefore we believe that the number of emails a developer posted on this mailing list will provide a reasonable proxy for a developer's propensity to communicate publicly with the rest of the community. Our qualitative data suggested that leaders interested in organization building would be more likely to make proposals on the most central list.

Degree centrality. The other variable we used to measure a developer's involvement in the organizational life of the community is a count of the number of other community members a developer had met as of January 1st 2003. Our analysis in Part I indicated that meeting other developers face-to-face required some effort and was associated with interest in building the organization. To measure who met whom in the community, we took advantage of an interesting

practice that the project adopted in 1994 to authenticate member identities. In 2000, this practice, which uses public key cryptography, became a condition for becoming a Debian project member.

Public-key cryptography works with asymmetric key pairs (one public and one private). One party has a private key which is not shared and uploads a public key to a central keyring file. Other developers can then retrieve this public key and use it to confirm that messages sent from the first party are indeed from the sender. This confirms that the content of any electronic message is from the sender but does not verify the real world identities of key owners – the person who originated the key-pair. Thus, to make public-key cryptography a useful means of authenticating identity, a real-world identity must be linked to a public key.

The Debian project does this through the practice of key signing. Developers sign each others key in order to verify that they have met a person face to face. Members consider this to be an expression of trust: the signer meets the person, reviews government identification and then signs that they believe that a particular public key belongs to the person who claims it. These digital signatures are uploaded to a central file where all developers can access them. Developers can get their key signed in various ways. For example, the Debian website lists developers who want to have their key signed and willing key signers around the world. This way, Debian developers that are traveling for other reasons can arrange to meet each other. Community members also attend “key signing parties” where people meet to expressly sign keys, but these parties are often combined with other project activities (See Appendix B for an example invitation to a Debian key signing party). Since each key signing is dated and requires a face to face meeting, these data indicate when project members met each other. We collected data from the Debian keyring, consisting of keys signed by dyads from 1994 (when the practice began) until Jan 1st, 2003. We used these data to measure the degree centrality of each

developer, or the number of community members each developer has met face-to-face. Since this measure only counts the number of people one has ever met face to face in an otherwise fully distributed community, we consider it a very conservative measure.

Controls. With the data that were available, we controlled for a member's tenure on the project (in months). Because tenure did not follow a normal distribution we used its square root. We controlled for a contributor's geographical location, which could only be established at the continent level.

PART II: DEDUCTIVE RESULTS

Table 5 shows means, standard deviations and correlations among the variables we considered. By comparing the means of the leadership team with those of the average developers (column 1 and 3 in Table 5), we notice that in 2003, the leadership team had been in the community longer (on average 72.4 versus 38.8 months⁹) and maintained approximately the same number of packages (10.9 vs. 9.3 packages¹⁰). However, the packages that leaders maintained were used more widely than those of the average developer (411.6 vs. 34.8 users). Members of the leadership team were also much more active on the mailing list (52.5 vs. 3.9 messages posted) and had met many more Debian developers (43.1 vs. 9.2). The descriptive statistics are thus in line with our hypotheses that developers with greater technical contribution (in terms of impact but not effort), and more engagement in organization building are more likely to become a member of the leadership team.

⁹ Many of the independent variables were transformed to address non-normal distribution. In this section, in order to facilitate the comparisons between leaders and the rest of the developers, we report the descriptive statistics in the original scale. For the case of tenure, we took the power of the value in table 5, since $(\sqrt{x})^2=x$. Therefore: $(8.52)^2=72.4$ months.

¹⁰ For packages, usage and mailing list postings, we also report the descriptive statistics in the original scale, by taking the exponential of the value in the table 5 since $e^{\ln(x)}=x$. Therefore: $e^{2.39}=10.9$.

Insert Table 5 about here

Table 6 reports the logit coefficients and the odds ratios for the logistic regression models predicting membership in the leadership team in 2003.¹¹ In the final model, we included tenure, geographic location (by continent, with Europe as default category), the number and popularity of code packages maintained (to measure technical contribution), amount of participation in online discussions and the number of ties (degree centrality) of developers (to measure organization building) as independent variables.

Insert Table 6 about here

Technical Contribution. The results reported in Table 6 suggest that the sheer amount of technical contribution, as measured by the number of software packages maintained, has no effect on the propensity to become a member of the leadership team. In fact, in the full model in Table 6, it has a marginally significant negative effect. Hypothesis 1a therefore finds no support. However, the impact of a developer's technical contribution, as measured by the number of users who installed the packages a developer maintained, has a significant positive effect. If a package maintained by a developer is installed by 100 more users, the developer is 2.3 times more likely to become a member of the leadership team.¹² Hypothesis 1b is therefore supported.

¹¹ To help the interpretation of the logit coefficients, the odds ratios are reported in parentheses in the complete model (but only for coefficients that are statistically significant). Odds ratio are computed by taking the antilogarithm of the logit coefficient, thus for the effect of degree centrality on leadership in 2003, we can take the coefficient from model 5 in Table 6, and simply compute: $e^{0.055}=1.056$. Values exceeding 1 indicate an increased likelihood of becoming a member of the leadership team, while value less than 1 indicate decreased odds.

¹² The natural logarithm of 100 is 4.6 and one unit increase in the log of users increases the odds by 46%, therefore to interpret the meaning of the odds value we obtained for this measure we computed $4.6*0.50= 2.3$.

Organizational Building and Leadership. Our results reported in Table 6 (models 4 and 5), do not provide strong supporting evidence for Hypothesis 2a. Although in model 4, the variable on-line communication has a significant positive effect on the likelihood of joining the leadership team, this effect loses statistical significance after including degree centrality in model 5. Therefore we have to conclude that the sheer amount of communication does not lead to the attainment of leadership positions. We attribute our inability to find a significant effect for online communication to the quality of our measure. Indeed, our ethnographic evidence leads us to conclude that communication with the community through the mailing list is an important antecedent of the attainment of leadership positions, but only if this impact is recognized by other community members as significant. Raw measures of communication conflate critical conversations and flame wars, and in future research, selected content-coding of the messages could help tease apart these effects.

To test Hypothesis 2b, we used the conservative measure of face to face meetings obtained from the cryptography keyring. In Table 6 (model 5), we found that meeting ten more developers in the community, increased the likelihood of becoming a member of the leadership team by 56% (odds ratio = 1.056). Three mechanisms may be responsible for this effect. One, Debian developers may be more likely to vote for someone they have met face-to-face, because they have developed a more trusting relationship (e.g. Weisband and Atwater, 1999; Rocco, 1998; Jarvenpaa and Leidner, 1998). Two, candidates who have met more people face to face may be more proactive in creating coalitions or in seeking votes from other members (e.g. Thompson, 2005). Three, reciprocal concerns may be at play: members may be more likely to vote for developers they've met because they believe that, once in a leadership role, these developers may be able to help them in the future. Given the data with which we are working, we

are not in a position to adjudicate among these mechanisms. However, all three mechanisms are consistent with our theoretical argument that both technical contribution and organizational building is required for a developer to attain leadership positions.

Most informants stated that face to face meetings were helpful in facilitating coordination on the project and that they gave “a less impersonal feel to the names we see on mailing lists.” As one leader who promised to raise funds to support travel said, “meetings in real life ... are very important. Future interactions are just different once you have gotten drunk with someone.” However, others expressed reservations, fearing that “non-electronic social relationships” could create or exacerbate inequalities in status.

Debian should be a meritocracy, and *merit should be measured by one's contributions to the Project, not by a developer's nationality or non-electronic social relationships*. Debian was born on the Internet and could not have existed without it. The ease of electronic communication is our greatest asset, and the most effective leveler of inequities that we possess. We must not abandon this essential attribute in favor of provincialism. (*2004 Debian Leadership Candidate's Leadership Platform*)

Despite our informants' espoused preference for a “pure” meritocratic system, which was echoed in the literature, the attainment of leadership positions cannot be explained solely by one's technical contribution to the community. Members of the leadership team were also more likely to engage in organization building activities. This evidence reinforces our earlier finding on the evolution of the conception of authority described in the inductive section of the paper.

DISCUSSION

A large body of scholarship has examined the introduction of democratic or participatory mechanisms into bureaucratic organizations (Barker, 1993; Adler et al, 1997, 1999; Cappelli and Neumark, 2001). We have studied the reverse process – the introduction of bureaucratic

mechanisms into community forms. This inquiry is important both because scholars increasingly recognize the viability of community forms of production as alternatives to markets and hierarchy (Adler, 2001; Powell, 1990; Bradach and Eccles, 1989; Ouchi, 1980) and because our knowledge of how community forms manage production has rested upon ideal type conceptions.

Despite the fact that there has been a surge of interest in communities devoted to sharing knowledge, improving their work practice, sustaining their occupational identity and exchanging social support, we know very little about how communities engaged in production govern themselves. For many years, scholars have acknowledged that even those social systems that disregard authoritarian rule need some type of governance system in order to coordinate, manage scale and sustain themselves (March and Simon, 1958; Harrison, 1960). However, little research has shed light on how communities engaged in production manage this tension. This research bridges this gap by showing how one open source community designs and implements a governance system.

Blending democratic and bureaucratic organizing mechanisms. One reason why community forms of production have not received as much attention as traditional capitalist forms (e.g. Stern and Barley, 1996; Perrow, 1991) is their inability to resolve problems of power, authority and governance (Rothschild-Whitt, 1979; Rothschild and Russell, 1986 and Rothschild-Whitt, 1986). With ethnographic research, we showed how a production community designed a governance system that incorporates a constitutionally endowed basis of authority with democratic mechanisms to ensure control by the majority. We found that, in Phase I, there was little shared basis of authority which led to conflict and the recall of the second leader. Barnard (1938) argued that authority was most effective when it was unquestioned or reached a

‘taken for granted’ state. Only after creating a constitution in Phase II, did the role of a Debian Project Leader acquire Barnard’s theorized taken for granted state – albeit in a limited fashion.

Debian’s governance system incorporates mechanisms that reinforce both bureaucratic and democratic values. A rational basis of authority is created by defining leadership positions and their associated rights, fulfilling the community’s need to have someone with the authority to represent the project. However, democratic rule is reinforced by: 1) requiring leaders to act with deference to the community; 2) granting leaders limited authority over technical matters; 3) allowing members to recall the leader’s authority and 4) creating a source of countervailing power. Positional authority is restricted to facilitating coordination and project-wide decisions, aiding in the resolution of conflict among individuals, and representing the project to outside parties. Thus most authority remains laterally distributed. By combining elements of democratic and bureaucratic control, this system allows both regimes to co-exist (e.g. Bradach and Eccles, 1989; Stark, 1999). Thus, we propose that to successfully introduce a bureaucratic basis of authority into a community form, members must also design democratic mechanisms to delimit that basis of authority. However, our research suggests that these democratic mechanisms play a dual role – not only ensuring that the governance system represents the community’s interests, but also by providing the system with an adaptive mechanism. We found that by blending democratic with bureaucratic mechanisms, a community based governance system was able to adapt its conceptualization of authority to the changing conceptions of its members.

Reconceptualizing meritocracy. We did not want to just understand how a production community develops a governance system, but also how this system worked in practice. We identified six different conceptions of authority over time, and found that leaders embraced more organizational building behaviors over time. The tension between technical and organizational

building contributions to the project was not only apparent in the discourse of leader candidates, but confirmed by analysis of which developers became leaders in 2003. Developers were more likely to become a member of the leadership team when their technical contributions were widely used by other members, as opposed to the mere volume of their efforts. Contrary to a simplistic meritocratic explanation, developers who engaged in organization building behaviors were more likely to become members of the leadership team. Thus, Debian may be a meritocracy, but merit is not measured solely by ones' technical contribution. Research on open source software communities tends to assume that contributions that are most valued by community members are purely technical. Our study shows otherwise. In this community, the informal work of coordinating individual efforts and linking them to community goals became a vital role of leadership, particularly as the project matured.

This exploration of how meritocracy works in practice contributes directly to the debate started by Parsons (1947) and Gouldner (1954), extended most recently by Adler and Borys (1996), as to the precise nature of the relationship between meritocracy and bureaucracy. Are positions of authority the reward for technical competence? The answer turns on the definition of technical competence as the cornerstone of merit. If organizational competence becomes more important to community members than technical competence, than we would argue that meritocracy still reigns. This research suggests that we need to think about conceptions of meritocracy in a more nuanced way. Any examination of meritocracy must develop a context-specific understanding of how merit is conceptualized. Rather than questioning whether the outcomes of bureaucratic rule are in fact consistent with meritocracy, this research suggests that a more fruitful pursuit would be to examine how conceptions of merit evolve.

What is counter intuitive is that a community so wary of the effects of positional authority that they would actively take steps to limit it, would demonstrate a growing preference for leaders who expand the reach of their authority over time. Members of this community were technical experts with many of the latest software development and coordination tools at their disposal. They also made use of many different types of media to manage their work. Yet coordinating their efforts and managing interdependence grew problematic without the help of people managing the organizational structure. This suggests that while technology may have changed the ability of such groups to coordinate their efforts over space and time, even the most savvy online community may not be immune to well known general principles of organizing (Michels, 1911; March and Simon, 1958; Ouchi, 1979).

Limitations. This research is based on one in depth case and so can only provide an existence proof of the design and implementation of a community governance system. While the Debian community could be viewed as an extreme case, we would argue, with Starbuck (1993, 1998) that the study of extreme cases can help to resolve research deficiencies in other methods such as overgeneralizations or neglect of individuality, complexity and variety. Starbuck argues that many organizations strive to differ from others and fill critical niches that would otherwise not be identified without such research (1993, 1998). The Debian community fits these criteria well. We deliberately chose a community that had evidence of leadership turnover. Many production communities initiated by strong founders (e.g. Linux and Wikipedia) have not experienced leadership turnover and thus have not developed an institutionalized basis for leadership. However, this does not preclude the relevance of our findings to founder led communities, for these design choices may be in their future. We would expect our findings to

apply to other types of distributed communities engaged in knowledge creation or production, but future comparative studies can confirm the extent to which these findings are generalizable.

Implications for traditional organizations. Our findings on the importance of face to face interaction in predicting community leadership confirm earlier predictions that few communities exist purely in an online form and suggests the need to understand how hybrid forms function (Fiol and Connor, 2005; Griffith and Neale, 2001). Our findings are consistent with the results of experimental studies that found that group members liked each other more when communicating face to face versus electronically (Weisband and Atwater, 1998) and that even a single face to face meeting can trigger more trustworthy behavior online (Rocco, 1998). For a leader to become elected by a community that restricts the authority of their leader, they must undoubtedly earn the trust of project members.

Perhaps members are more confident that their leader will not abuse the little authority entrusted to them if they have met face to face. These results may not be surprising to those who have proposed that face to face communication is the ‘gold standard’ of communication (e.g. Nardi and Whittaker, 2002; Nohria and Eccles, 1992), but thus far the evidence supporting this argument has been primarily experimental. Our in-depth case study provides ecological validity to these findings. While external validity speaks to a study’s ability to generalize, ecological validity speaks to a study’s ability to approximate a ‘real-life’ situation under study (Brewer, 2000; Shadish, Cook and Campbell, 2002). Although these are independent constructs, when one improves ecological validity, external validity often improves as well.

Our findings also have implications for autonomous work groups and their leaders in traditional organizations (Hackman, 1978). Autonomous work groups have responsibility for a whole task and the autonomy to make decisions on how they carry out their work (Hackman,

1978; Alper, Tjosvold and Law, 1998). Recent research has identified leader behaviors that are critical to the success of such groups such as building relationships, scouting for information, persuading constituents to offer their support (Druskat and Wheeler, 2003), and fostering team members' self-observation and evaluation (Manz and Sims, 1987). Researchers have also shown that individuals who emerge as leaders in these settings are likely to be extroverted, emotionally stable, open to experience and conscientious (Judge, Bono, Ilies and Gerhardt, 2002), as well as high in cognitive ability (Taggar, Hackett and Saha, 1999) and self-monitoring (Day, Schleicher, Unckless and Hiller, 2006). With few exceptions, this research takes an individual differences approach (Ilies, Gerhardt and Le, 2004) without providing insight as to the behaviors that could play a role in the emergence of leaders in these setting. Our study focuses on the behaviors associated with becoming a leader in a community where leaders are designated by the group. However, our findings on the role of technical and organizational building could fruitfully be explored in the context of autonomous work groups. Another interesting avenue of research could stem from exploring how conceptualizations of authority evolve in self managed teams.

This research explains how governance and leadership emerges in distributed communities that have no obvious basis of authority - and this is relevant to the management of professional expertise in many contexts. As Scott points out, hierarchical systems have been increasingly "giving way to more decentralized and horizontal systems, particularly among organizations in the newer industries" (2004: 12). We would predict that the more information and knowledge is distributed, the more likely it is that democratic approaches will be appropriate. Future research should examine how democratic decision mechanisms can be integrated with more traditional forms of bureaucratic and professional or occupational control (Van Maanen and Barley, 1984). As porous and dynamic organizational boundaries (Santos and Eisenhardt, 2005) enable a

growing body of constituents and specialists relevant to decision making, governance systems that support a community of distributed experts are likely to continue to be important.

APPENDIX A

Leadership Platform Coding

With the coding of conceptions of leadership in hand, we noticed a shift from technical to organization building and leadership concerns. Table 4 shows gradual movement from a limited conception of authority to a more expansive degree of positional authority. However, competing conceptions of authority continued to exist until Phase IV. To further understand the contours of this shift, we studied the content of leadership platforms over time at another level. Each author independently coded the 34 leadership platforms and classified each section of text according to whether they were mainly focused on addressing organizational issues, or engaging in technical discussions and debates, or providing biographical information.

For instance an example of text item coded as “organizational issues” would be: “I think one of the main tasks of the Project Leader is to coordinate and motivate people - to lead. This is why I said that while the external functions of the DPL are important, the internal functions are even more important. While it is quite hard to lead a project consisting of so many people with so diverse expectations and personalities, I think that it can actually be done. Thus, my main aim as Debian Project Leader is to lead, motivate and coordinate.” (DPL platform 2003). An example of text item coded as “Technical issue” would be “Similarly, while everyone complains about the release cycles of Debian, it is not clear at all how frequently we should release. For example, SuSE and Red Hat have introduced offerings of their distributions that have similar release cycles to that of Debian. There are many reasons why you can upgrade your system only every two years. Yet, there are also users who want the newest software and like to upgrade twice a year” (DPL platform 2003). Finally an example of text coded as “biographical information” is the following: “I hold a Master degree in Philosophy and have recently completed a Master of Science in Psychology. I'm currently doing a Master of Software Systems Engineering at the

University of Melbourne and am looking forward to pursuing a PhD in Software Engineering about Debian and Free Software afterwards” (DPL platform 2003). We counted the number of words dedicated to each of these three themes, and compared their evolution. To ensure the robustness of our results, we asked another researcher to code the platforms independently and compared his results to ours. There was a 0.85 correlation in the results on organizational issues, 0.95 for technical issues, and 0.90 for biographical. Given these results, we deemed it safe to proceed with our original coding.

Figure 2 shows that, over time, leaders’ conceptions of authority grew increasingly focused on organizational versus technical issues. To test whether these changes were statistically significant we regressed the proportions of text dedicated to organizational, technical issues and biographical information on a trend line, measured with a scale from 1 to 8. The models testing for changes in the proportion of technical issues and biographical information were not significant, but when we regressed the proportion of text dedicated to organizational issues on a trend line, the resulting model was statistically significant ($F < 0.01$; $R^2 = .76$) and a year coefficient of .047 ($t < .01$), meaning that on average, the proportion of text dedicated to organizational issues increased every year by 4.7%.

APPENDIX B

Keysigning Party Invitation

Subject: Reminder for Signers, Signees
From: xxxxxx@yyy
Date: 16 Feb 2001 14:53:56 -0800

So, people who need their key signed, remember to:

1. Send me your key by tonight, so I can assemble them into a nice keyring + printout for the signers.
2. Bring the following tomorrow:
 - * POSITIVE form of ID, like a driver's license or passport
 - * Your own copy of your key fingerprint
3. Tomorrow I'll have:
 - * Nice name + key fingerprint printouts for signers to use.
4. Signers should bring:
 - * A pen or pencil to mark off the fingerprint printouts once they've checked IDs.

If you've never done a keysigning party before, here's how it works: the keysigner and keysignee meet. Keysigner checks the signee's ID against his/her printout, to make sure that he/she is talking to the right person. Then, signer makes a check next to the signee's name, to indicate "identified". Keysignee then reads out key fingerprint to keysigner. If the fingerprints match, and the ID is good, they make another check mark next to the name on their printout, indicating "key fingerprints match." ONLY IF BOTH CHECKMARKS EXISTS should the signer sign the key.

After the party, the next steps are:

- ✓ Every signer downloads the unsigned keyring from the Web URL I'll publish.
- ✓ Every signer signs the keys they verified (i.e., checked their print out).
- ✓ Every signer sends me their edited keyring with now-signed keys.
- ✓ I combine the signatures and make a new keyring, with all the signed stuff, and make that available on the Web.

Note that to be a Debian New Maintainer you only need one signature from one Debian maintainer. However, it's good for the Web of Trust for you to give and get lots of (valid) signatures.

TABLE 1: Data Sources

Data Source	Observation Hours	Interviews	Project Records
	90		
Debian Project Leaders (DPL)		4	
Debian Contributors		14	
User contributors		30	
Total		48	
Leader candidate platforms (1999 – 2006)			34
Election debates (2000, 2003, 2005-6)			4
Meetings (1998 – 2005)			32
Project General Resolutions (1999 – 2006)			5
Mailing list postings (1998 – 2006)			17,317
Project documents (Constitution, Policy Manual, Social Contract)			100+ pages

TABLE 2: Phases of Governance in an Open Source Community

Defining Event	Example	Type of Evidence	EvidenceStrength
Phase I: De facto Governance 1993-1997			
Autocratic leadership emerges and is challenged	“But the big thing, which changed with [the 2 nd DPL], like he was really managing everything and kept tight control, which, was plausible because Debian was much smaller. Then actually [with the third leader] he never wanted to have someone like [that] again, and that is why he started to write the constitution, which splits the power.” (<i>Debian developer member</i>)	I, S, D	Medium
Phase II: Designing Governance 1997-1999			
Formal authority is developed	“The Project Leader may... define an area of ongoing responsibility or a specific decision and hand it over to another Developer or to the Technical Committee; ..lend authority to other Developers; ...make any decision which requires urgent action; [or] for whom no one else has responsibility.” (<i>Debian Constitution, Article 5.1</i>)	I, S, D	High
Formal authority is limited through democratic means	“Together, the Developers may: 1) Appoint or recall the Project Leader. 2) Amend this constitution, provided they agree with a 3:1 majority. 3) Make or override any decision authorised by the powers of the Project Leader.” (<i>Debian Constitution, Article 4.1</i>)	I, S, D	High
Phase III: Implementing Governance 1999-2003			
Varying conceptions of formal authority are debated	“It’s not clear exactly what the DPL is supposed to do. What do people expect? Because Debian was so big, everyone expects something else. So, some people say, “Yeah the DPLs should only go to conferences, and present Debian to the outside world, but he shouldn’t do anything internal because that is working anyway.” (<i>Debian developer member</i>)	I, S, D	High
Community members elect leaders through democratic means	“I’ll cast my DPL [Debian Project Leader] vote towards the end of the cycle, as usual. Here’s why: At the start of the cycle, I hadn’t made up my mind.....I haven’t finished reading the candidate platforms and debate material yet....This year, since once again I couldn’t attend them live, I have to read them afterward, which takes time....Voting in Debian is just like voting anywhere else, you often have to do a lot of reading to understand the issues” (<i>Debian developer member</i>)	I, S, D	High
Phase IV: Stabilizing Governance 2003-2006			
A shared conception of formal authority emerges	When community members became disgruntled with a leader’s actions, they worked within the system by proposing a General Resolution to recall the leader. However, members voted to discuss the matter further and, in a second vote, the leader’s authority was re-affirmed (<i>General Resolutions, 2006</i>).	I, S, D	Medium

Key

I = Interview Data
 S = Secondary Source
 D = Project Documentation

High – this finding was apparent in a preponderance of the data
 Medium – this finding was consistently repeated
 Low – this finding was suggestive

TABLE 3: Electoral Participation in Debian Leader Elections 1999 - 2006

Election Year	# of DPL Candidates	# of Eligible Developers	# Who voted	Voting Participation Rate
1999	4	347	208	60%
2000	4	347	216	62%
2001	3	530	311	59%
2002	3	939	475	51%
2003	4	831	488	59%
2004	2	911	482	53%
2005	6	965	506	52%
2006	6	972	421	43%

TABLE 4: Debian Project Leader Conceptions of Authority 1993 – 2006

Conceptions of Authority	Degree of Positional Authority	Exemplar Evidence	Years												
			1993-1996	1996-1997	1998	1999	2000	2001	2002	2003	2004	2005	2006		
			De facto leaders			Number of leadership candidates									
			1	1	1	4	4	3	3	4	2	6	6		
Hands-Off Leader	Limited	“The DPL should have a fairly hands-off approach...the DPL will need to delegate some of his responsibilities to others.” “The DPL is the public figure head of Debian.” (2000 Leader Election)				••	•	•		•		•			
Technical Manager	Medium	“A chronic problem in Debian has been our overly long release cycle, if elected this shall be my major focus.” (2000 Leader Election)	✱			•	•		•				•		
Visionary Leader	Medium	“It has been a long time since Debian had a clearly articulated vision. It’s time to fix that...there is no reason that we can’t eventually be a truly universal operating system! I can’t imagine a more compelling vision for our future.” (2002 Leader Election)				✱			✱	•					
Organization Builder	High	“We have learned a harsh lesson about the organizational structure in Debian..Debian is continuously growing and changing and we need to know how to handle that. Old structures and ideas will need to be revised, and new ones will arrive” (2000 Leader Election)			✱		✱•	✱•	•	•	•	•	•		
Organizational Leader	High	“Leadership in voluntary organizations like Debian lacks the instruments of power that are used in business environments....Debian’s goal is to produce an excellent free operating system and have fun doing so. The leader’s task is to keep that vision alive and vivid and to encourage people to want to go there.” (2005 Leader Election)								✱	✱	✱• ••	✱• ••		
Autocratic Leader	Very High	“Well project leader was essentially dictator. Because there was no formal structure whatsoever at that time.” (1996-7 Leader)		✱											

Key = ✱ conceptions of authority articulated by the leader who won the election that year; • conceptions of authority articulated by a single leader candidate in that election year

Note: Shaded years indicate years of informal de facto leadership prior to the construction of formal authority via the Constitution.

TABLE 5
Descriptive Statistics and Correlations^a

Variable	Mean (Leaders)	s.d.	Mean (All obs.)	s.d.	Min	Max	1	2	3	4	5	6	7	8	9
1. Leadership Team (2003)	1	0	0.01	0.10	0	1	-								
2. Tenure (square root of months, 2002)	8.52	1.29	6.23	1.53	2.15	10.66	0.15**	-							
3. North-America	0.50	0.54	0.32	0.47	0	1	0.04	0.07*	-						
4. Europe	0.38	0.52	0.52	0.50	0	1	-0.03	-0.03	-0.71**	-					
5. Other	0.13	0.35	0.16	0.37	0	1	0.00	-0.05	-0.29**	-0.46**	-				
6. Technical Contribution (N. of packages, 2002) ^b	2.39	1.55	2.23	1.47	0	6.93	0.01	-0.01	-0.06	0.04	0.03	-			
7. Impact of Tech. Contribution (N. of users, 2002) ^b	6.02	2.54	3.55	2.68	0	9.51	0.09*	0.01	-0.00	0.02	-0.03	0.73**	-		
8. Online Communication (N. of email posted, 2002) ^b	3.96	2.12	1.35	1.68	0	6.43	0.16**	-0.05	0.02	0.02	-0.05	0.38**	0.47**	-	
9. Degree Centrality (# of developers met) (2002)	43.13	40.45	9.55	13.43	0	114	0.25**	0.14**	-0.08*	0.15**	-0.10**	0.24**	0.21**	0.33**	-

^a n=815

^b Log-transformed variable

*=p<0.05, **=p<0.01

TABLE 6
Logistic Regression of Leadership in 2003 on Selected Independent Variables ^a

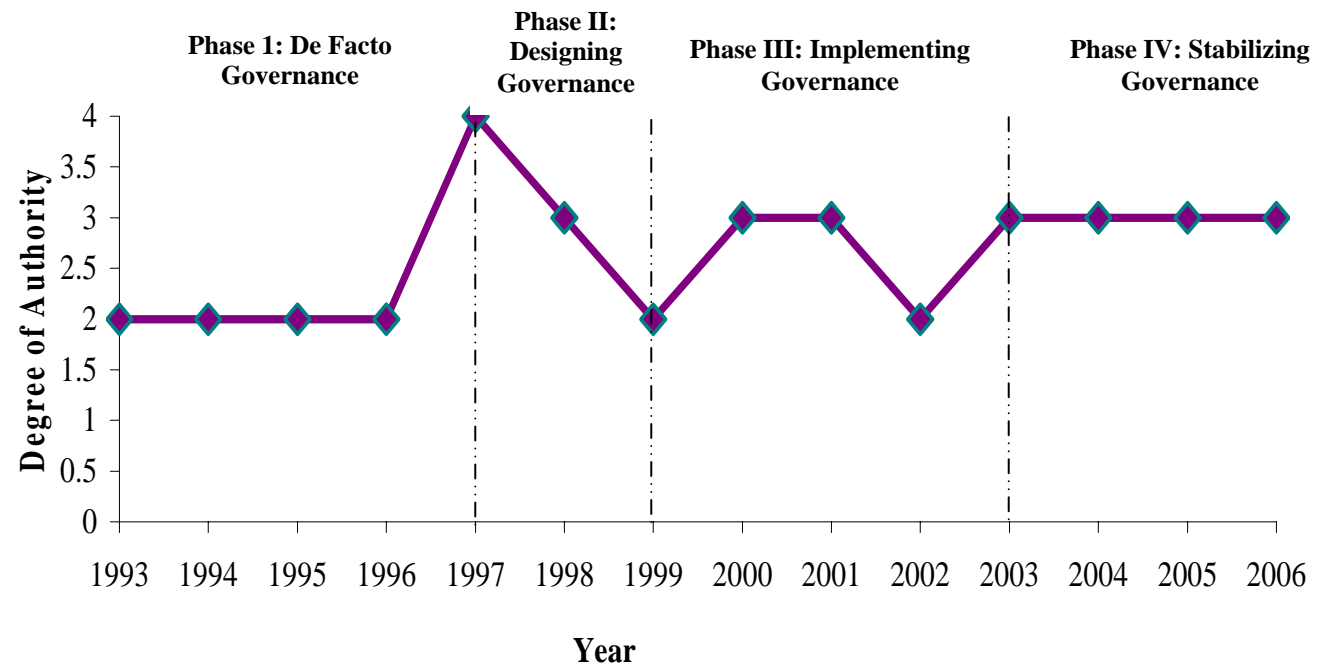
Variable	Model 1		Model 2		Model 3		Model 4		Model 5		Odds ratio
	b	s.e.	b	s.e.	b	s.e.	b	s.e.	b	s.e.	
Tenure (Square root of month)	1.248***	(0.345)	1.247***	(0.329)	1.319***	(0.403)	1.550***	(0.558)	1.652***	(0.508)	5.216***
North America ^b	1.016	(0.842)	1.013	(0.826)	0.775	(0.846)	0.806	(0.889)	1.587**	(0.714)	4.891**
Other Continent ^b	0.658	(1.257)	0.649	(1.231)	0.641	(1.241)	0.244	(1.259)	1.448	(1.201)	4.256
Technical Contribution ^c			0.009	(0.197)	-0.720**	(0.342)	-0.984***	(0.345)	-1.341***	(0.446)	0.262***
Impact of Technical Contribution ^c					0.546***	(0.165)	0.248*	(0.141)	0.408**	(0.170)	1.504**
Online Communication ^c							0.827***	(0.318)	0.565	(0.405)	1.760
Degree Centrality									0.055***	(0.015)	1.056***
Intercept	-14.491***	(3.182)	-14.496***	(3.234)	-15.872***	(3.782)	-17.886***	(5.260)	-19.848***	(4.564)	
Wald Chi (df)	13.32 (3)		16.61 (4)		31.25 (5)		27.78(6)		58.62 (7)		
Pseudo R-squared	0.23		0.23		0.34		0.47		0.58		

^a n=815, Robust Standard Errors reported for all models, Odds ratios reported for the complete model only

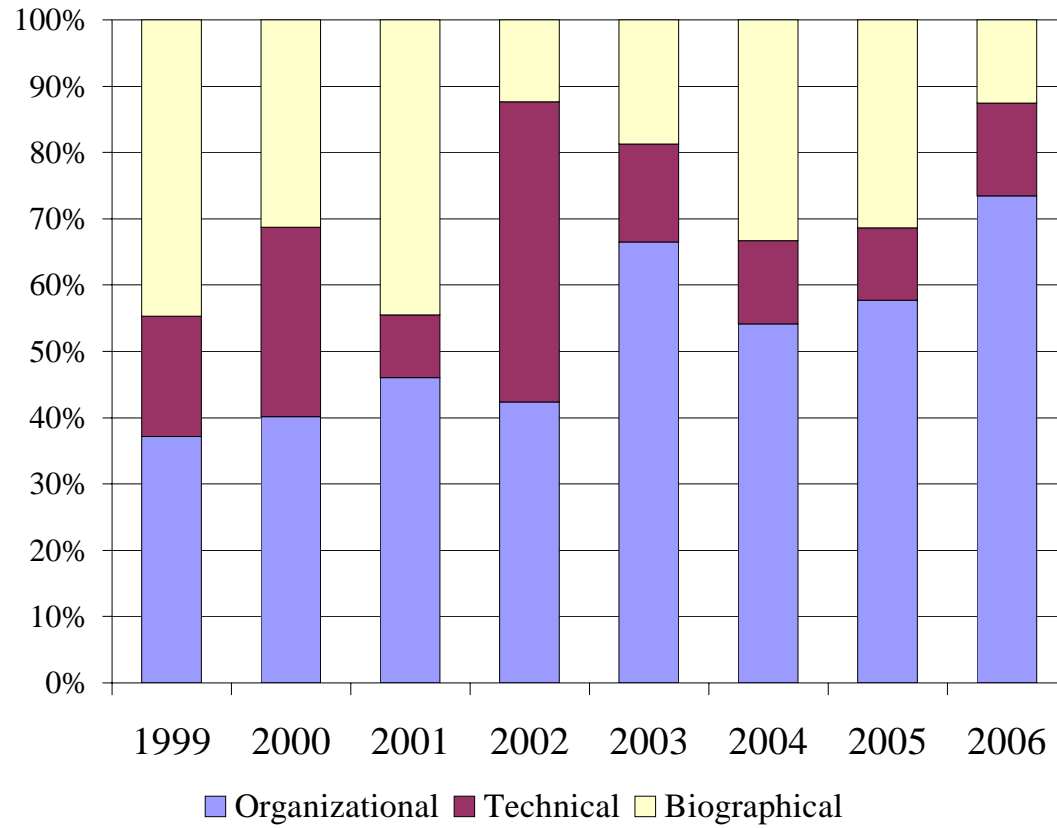
^b Compared to developers located in Europe (default category)

^c Log-transformed variable

*=p<0.1, **=p<0.05, ***=p<0.01

FIGURE 1: SHIFTS IN AUTHORITY OVER TIME**Key:**

- 1 Low
- 2 Medium
- 3 High
- 4 Very High

FIGURE 2: Word Usage in Leader Candidate Platforms

REFERENCES

- Adler, P.S. & B. Borys. 1998. Two Types of Bureaucracy: Enabling and Coercive. *Administrative Science Quarterly*, 41: 61-89.
- Adler, P.S. & B. Borys. 1996. Two Types of Bureaucracies: Enabling and Coercive. *Administrative Science Quarterly*, 41: 61-89.
- Adler, Paul S. 2001. Market, Hierarchy, and Trust: The Knowledge Economy and the Future of Capitalism, *Organization Science*, 12: 215-234.
- Adler, P.S., B. Goldoftas & D. Levine. 1997. Ergonomics, Employee Involvement, and the Toyota Production System: A case study of NUMMI's 1993 Model Introduction. *Industrial and Labor Relations Review*, 50(3) April: 416-437.
- Adler, P.S., B. Goldoftas & D. Levine. 1999. Flexibility versus efficiency? A case study of model changeovers in the Toyota Production System. *Organization Science*, 10(1) Jan-Feb: 43-68.
- Adler, Paul S. and C. Heckscher. 2006. "Towards Collaborative Community" in *The Firm as a Collaborative Community: Reconstructing Trust in the Knowledge Economy*, Ed. Charles Heckscher and Paul S. Adler, New York, NY: Oxford University Press, Pp 11 – 105.
- Alper, S., D. Tjosvold & K.S. Law. 1998. Interdependence and controversy in group decision making: Antecedents to effective self-managing teams. *Organizational Behavior and Human Decision Processes*, 74: 33-52.
- Armstrong, A., & Hagell, J. 1996. The Real Value of On-Line Communities. *Harvard Business Review*, 74(3): 134-141.
- Barker, James R. 1993. Tightening the Iron Cage: Concertive Control in Self-Managing Teams. *Administrative Science Quarterly*, 38: 408-37.
- Barnard, C. I. 1938. *The functions of the executive*. Cambridge, Mass: Harvard University Press.
- Bechky, B. 2003. Object Lessons: Workplace Artifacts as Representations of Occupational Jurisdiction. *American Journal of Sociology*, 109: 720-52.
- Blau, Peter M. 1955. *The Dynamics of Bureaucracy*. Chicago: University of Chicago Press.
- Bradach, J. R., R. Eccles. 1989. Markets versus hierarchies: From ideal types to plural forms. W.R. Scott, Ed. *Annual Review of Sociology* 15: 97-118.

- Brewer, M. (2000). Research Design and Issues of Validity. In Reis, H. and Judd, C. (eds) Handbook of Research Methods in Social and Personality Psychology. Cambridge:Cambridge University Press.
- Brown, J. S., & Duguid, P. 1991. Organizational learning and communities-of-practice: Toward a unified view of working, learning, and innovation. *Organization Science*, 2: 40-57.
- Brown, J. S., & Duguid, P. 2000. *The Social Life of Information*. Boston: Harvard Business School Press.
- Brown, J. S., & Duguid, P. 2001. Knowledge and Organization: A Social-Practice Perspective: *Organization Science*, 12: 198-213.
- Butler, B. S., Sproull, L., Kiesler, S., & Kraut, R. Community Effort in Online Groups: Who Does the Work and Why? In S.A.L. Weisband (Ed.), *Leadership at a Distance*. Erlbaum, In Press.
- Campbell, D.T. & D.W. Fiske. 1959. Convergent and discriminant validation by the multi-trait-multi-method matrix. *Psychological Bulletin*, 56: 81-105.
- Cappelli, Peter and David Neumark. 2001. Do High-Performance" Work Practices Improve Establishment-Level Outcomes. *Industrial Labor Relations Review* 54(4):737-775.
- Cassell, J., Huffaker, D., & Tversky, D. 2005. How to win a world election: Emergent leadership in an international online community. Paper presented at the Proceedings of the Second Communities and Technologies Conference, Milan.
- Coleman, E. Gabriella, 2005 Three Ethical Moments in Debian. Available at SSRN: <http://ssrn.com/abstract=805287>
- Coleman, J. S., 1970. Social Inventions. *Social Forces*, 49: 163-173.
- Coleman, J. S. 1974. *Power and the Structure of Society*. New York: W. W. Norton & Company Inc.
- Coleman, J. S. 1980. Authority Systems. *Public Opinion Quarterly*, 44: 143-163.
- Coleman, J. S. 1993. The rational reconstruction of society: 1992 Presidential Address. *American Sociological Review*, 58: 1-15.
- Cummings, J., L. Sproull & S. Kiesler. 2002. Beyond Hearing: Where Real-World and Online Support Meet. *Group Dynamics*, 6: 78-88.

- Dalle, J. M., & Jullien, N. 2003. 'Libre' software: Turning fads into institutions? *Research Policy*, 32: 1-11.
- Day, D. V., Schleicher, D. J., Unckless, A. L., & Hiller, N. J. 2002. Self-monitoring personality at work: A meta-analytic investigation of construct validity. *Journal of Applied Psychology*, 87: 390-401.
- DiMaggio, P. J., & Powell, W. 1983. The Iron Cage Revisited: Institutional Isomorphism and Collective Rationality in Organizational Fields. *American Sociological Review*, 48: 147-160.
- DiMaggio, P. J., & Anheier, H. 1990. The Sociology of Nonprofit Organizations and Sectors. *Annual Review of Sociology*, 16: 137-159.
- Drucker, P. F. 2001. *The Essential Drucker*. New York: HarperCollins
- Druskat, V.U. & J.V. Wheeler. 2003. Managing from the boundary: The effective leadership of self-managing work teams. *Academy of Management Journal*, 46: 435-457.
- Edmondson, A., & McManus, S.E. Forthcoming. Methodological Fit in Management Field Research. *Academy Management Review*.
- Etzioni, A. 1959. Authority Structure and Organizational Effectiveness. *Administrative Science Quarterly*, 4(1): 43-67.
- Fayard, A. L., DeSanctis, G., & Roach, M. 2004. The language games of online forums. Duke University, Durham, NC.
- Fiol, C. M., & O'Connor, E. J. 2005. Identification in face-to-face, hybrid, and pure virtual teams: Untangling the contradictions. *Organizational Science*, 16(1): 19-32.
- Fleming, L., & Waguespack, D. forthcoming. Penguins, Camels and Other Birds of a Feather: The Emergence of Leaders in Open Innovation Communities. *Organization Science*
- Fligstein, N., 1990. *The Transformation of Corporate Control*. Cambridge, Mass: Harvard University Press.
- Glaser, B. G., & Strauss, A. L. 1967/99. *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Hawthorne, NY: Aldine de Gruyter.
- Gouldner, A. 1954. *Patterns of industrial bureaucracy*. New York: Free Press.
- Griffith, T. L., & Neale, M. A. 2001. Information processing in traditional, hybrid, and virtual teams: From nascent knowledge to transitive memory. In B. M. Staw & R. I. Sutton (Eds.), *Research in Organizational Behavior*, 23: 379-421. Stamford, CT: JAI.

- Hackman, J. R. 1978. The design of self-managing work groups. In B. King, S.S. Streufert and F.E. Fiedler (Eds.), *Managerial Control and organizational democracy*. Washington D.C.: Winston & Son.
- Hannan, M., & Freeman, J. 1989. *Organizational Ecology*. Cambridge, Mass: Harvard University Press.
- Hargadon, A. & B. Bechky. 2006. When collections of creatives become creative collectives: A field study of problem solving at work. *Organization Science*.
- Harrison, P. M. 1960. Weber's Categories of Authority and Voluntary Associations. *American Sociological Review*, (25)2: 232-237.
- Hars, Alexander & Shaosong Ou. 2002. Working for Free? - Motivations of Participating in Open Source Projects. *Proceedings of the 34th Hawaii International Conference on System Sciences, International Journal of Electronic Commerce*, 6 (3): 25-39.
- Heath, C. & S. Sitkin. 2001. Big-B versus Big-O: What is organizational about organizational behavior? *Journal of Organizational Behavior*, 22: 43-58.
- Hertel, B., Niedner, S., & Herrmann, S. 2003. Motivation of software developers in Open Source projects: An Internet-based survey of contributors to the Linux kernel. *Research Policy*, 32(7): 1159.
- Ilies, R., Gerhardt, M. W., & Le, H. 2004. Individual differences in leadership emergence: Integrating meta-analytic findings and behavioral genetics estimates. *International Journal of Selection and Assessment*, 12: 207-219.
- Ingram, P. & Simons, T. 2000. State Formation, Ideological Competition, and the Ecology of Israeli Workers' Cooperatives. *Administrative Science Quarterly*, 45: 25-53.
- Jarvenpaa, S. L., Knoll, K., & Leidner, D. E. 1998. Is there anybody out there?: The implications of trust in global virtual teams. *Journal of Management Information Systems*, 14: 29-64.
- Johnson, A. G., & Whyte, W. F. 1977. The Mondragon System of Worker Production Cooperatives. *Industrial Labor Relations Review*, 31: 18-30.
- Judge, T. A., Bono, J. E., Ilies, R., & Gerhardt, M. W. 2002. Personality and leadership: A qualitative and quantitative review. *Journal of Applied Psychology*, 87: 765-780.
- Kanter, R. M. 1968. Commitment and social organization: A study of commitment mechanisms in utopian communities. *American Sociological Review*, 33(4): 499-517.

- Kieser, A. 1989. Organizational, Institutional, and Societal Evolution: Medieval Craft Guilds and the Genesis of Formal Organizations. *Administrative Science Quarterly*, 34(4): 540-564.
- King, G., & Zeng, L. 1999. "Logistic Regression in Rare Events Data," Department of Government, Harvard University, available from <http://GKing.Harvard.Edu>.
- Kogut, B., & Metiu, A. 2001. Open-Source Software Development and Distributed Innovation. *Oxford Review of Economic Policy*, 17: 248-264.
- Kollock, Peter. 1998. Social Dilemmas: The Anatomy of Cooperation. *Annual Review of Sociology*, 24: 183-214.
- Lakhani, K., & Wolf, B. 2005. Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects. In H. Feller, B. Fitzgerald, S.A. Hissam & K.R. Lakhani (Eds.), *Perspectives on Free and Open Source Software*. Cambridge, Mass: MIT Press.
- Lave, J., & Wenger, E., 1991. *Situated Learning: Legitimate Peripheral Participation*. Cambridge: Cambridge University Press.
- Lee, G., & Cole, R. E. 2003. From a Firm-Based to a Community-Based Model of Knowledge Creation: The Case of the Linux Kernel Development. *Organization Science*, 14:633-649.
- Lerner, J., & Tirole, J. 2002. The simple economics of open source. *Journal of Industrial Economics*, 52: 197-234.
- Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: Sage Publications.
- March, James G. & Herbert A. Simon. 1958. *Organizations*. New York: John Wiley and Sons.
- Manz, C.C., & H.P. Sims. 1987. Leading workers to lead themselves: The external leadership of self-managing work teams. *Administrative Science Quarterly*, 32: 106-129.
- Merton, Robert K. 1940. Bureaucratic Structure and Personality. *Social Forces*, 18:560-568.
- Marsden, P. 2005. The Sociology of James S. Coleman. *Annual Review of Sociology*, 31: 1-24.
- Michels, Robert. 1911. *Political Parties: A Sociological Study of the Oligarchical Tendencies of Modern Democracy*. New York, NY: The Free Press.

- Misiolek, N., & Heckman, R. 2005. Patterns of emergent leadership in virtual teams. Paper presented at the Proceedings of the 38th Hawaii International Conference on System Sciences, Hawaii.
- Mockus, A., Fielding, R. T., & Herbsleb, J. D. 2002. Two Case Studies of Open Source Software Development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology*, 11: 309-346.
- Moon, J. Y., Sproull, L. 2002. Essence of distributed work: The case of the Linux kernel. In P. Hinds & S. Kiesler (Eds.), *Distributed Work: New Research on Working Across Distance Using Technology*: 381-404. Cambridge, MA: MIT Press.
- Nardi, B., & Whittaker, S. 2002. The place of face to face communication in distributed work. In P. Hinds & S. Kiesler (Eds.), *Distributed Work: New Research on Working Across Distance Using Technology*: 83-110. Cambridge, Mass: MIT Press.
- Netcraft. 2005. Strong growth for Debian. Retrieved from http://news.netcraft.com/archives/2005/12/05/strong_growth_for_debian.html on 9/114/06.
- Nohria, N., & Eccles, R. G. 1992. Face-to-Face: Making Network Organizations Work. In N. Nohria & R. G. Eccles (Eds.), *Networks and organizations: structure, form, and action*: 288-308. Boston: Harvard Business School Press.
- O'Mahony, S. 2003. Guarding the Commons: How community managed projects protect their work. *Research Policy*, 32(7): 1179-1198.
- O'Mahony, S., & Ferraro, F. Forthcoming. Managing the Boundary of an Open Project. In W. Powell, J. Padgett (Eds.), *Market Emergence and Transformation*.
- Orr, J. 1996. *Talking About Machines: An Ethnography of a Modern Job*. Ithaca, N.Y.: ILR Press.
- Ouchi, W. 1979. A Conceptual Framework for the Design of Organizational Control Mechanisms. *Management Science* 25: 833-848.
- Ouchi, W. 1980. Markets, bureaucracies, and clans. *Administrative Science Quarterly* 25: 125-141.
- Parks, M. R., & Floyd, K. 1996. Making Friends in Cyberspace. *Journal of Communication*, 46: 80-97.
- Parks, M. R., & Roberts, L. D. 1997. Making MOOsic: The development of personal relationships on-line and a comparison to their offline counterparts. *Journal of Social and Personal Relationships*, 15(4): 517-537.

- Parsons, Talcott. 1947. Introduction. Pp. 58-60 in *Theory of Social and Economic Organization*. Max Weber. New York: Oxford University Press.
- Perrow, C. 1991. A society of organizations. *Theory and Society*, 10:725-62.
- Pickering, J. M., & King, J. L. 1995. Hardwiring weak ties: Interorganizational computer-mediated communication, occupational communities, and organizational change. *Organization Science*, 6(4): 479-486.
- Powell, W. W. 1990. Neither Market nor Hierarchy: Network Forms of Organization. *Research in Organizational Behavior*, 12: 295-336.
- Powell, W. W., & Snellman, K. 2004. The Knowledge Economy. *Annual Review of Sociology*, 30: 199-220.
- Rao, H., 1998. Caveat Emptor: The Construction of Nonprofit Consumer Watchdog Organizations. *American Journal of Sociology*, 103: 912-961.
- Raymond, E. S. 1999. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. Sebastopol, CA: O'Reilly & Associates.
- Rheingold, H. 1993-2000. *The Virtual Community: Homesteading on the electronic frontier*, edn. Cambridge, MA: The MIT Press.
- Rocco, Elena. 1998. Trust breaks down in electronic contexts but can be repaired by some initial face-to-face contact. Proceedings of the SIGCHI conference on Human factors in computing systems, Los Angeles, CA. Retrieved 9/14/06 from <http://delivery.acm.org/10.1145/280000/274711/p496-rocco.pdf?key1=274711&key2=3679328511&coll=&dl=acm&CFID=15151515&CFTOKEN=6184618>.
- Romanelli, E. 1991. The Evolution of New Organizational Forms. *Annual Review of Sociology*, 17: 79-103.
- Rothschild-Whitt, J. 1979. The Collectivist Organization: An Alternative to Rational-Bureaucratic Models. *American Sociological Review*, 44: 509-527.
- Rothschild, J., & Russell, R. 1986. Alternatives to Bureaucracy: Democratic Participation in the Economy. *Annual Review of Sociology*, 12: 307-328.
- Rothschild, J., & Whitt, J. A. 1986. *The Cooperative Workplace: Potentials and dilemmas of organizational democracy and participation*. New York: Cambridge University Press.
- Santos F. M., & Eisenhardt, K. M. 2005. Organizational Boundaries and Theories of Organizing. *Organization Science*, 16(5): 491-508.

Satow, R. L. 1975. Value-Rational Authority and Professional Organizations: Weber's Missing Type. *Administrative Science Quarterly*, 20: 526-531.

Sawhney, M., & Prandelli, E. 2000. Communities of creation: Managing distributed innovation in turbulent markets. *California Management Review*, 42: 24-54.

Scott, W. R. 2004. Reflections on a Half-Century of Organizational Sociology. *Annual Review of Sociology*, 30:1-21.

Seidel, M. L., & Stewart, K. J. 2001. The C-Form: Emergence of a Post-Industrial Organizational Form. Paper presented at the 2001 Academy of Management Meetings, Washington, D.C.

Selznick, P. 1957. *Leadership in Administration: A Sociological Interpretation*. Berkeley, CA: University of California Press.

Shadish, W., Cook, T., and Campbell, D. (2002). *Experimental and Quasi-Experimental Designs for Generalized Causal Inference* Boston:Houghton Mifflin.

Shah, S. 2005. Open Beyond Software. In C. Dibona, D. Cooper & M. Stone (Eds.), *Open Sources 2*. Sebastopol, CA: O'Reilly Media.

Shah, S. K. Forthcoming. Attracting and Retaining Volunteer Software Developers. *Management Science*.

Simons, T., & Ingram, P. 1997. Organization and ideology Kibbutzim and hired labor, 1951-1965. *Administrative Science Quarterly*, 42(4): 784-814

Smith, M., & Kollock, P. 1999. *Communities in Cyberspace*. London: Routledge.

Starbuck, W. H. 1993. Keeping a butterfly and an elephant in a house of cards: The elements of exceptional success. *The Journal of Management Studies*, 30(6): 885-921.

Starbuck, W. H. 1998. Learning from Extreme Cases. Retrieved 12/16/2005, from <http://pages.stern.nyu.edu/~wstarbuc/extreweb/Extreweb.htm>.

Stark, D. 1999. Hierarchy: Distributing Intelligence and Organizing Diversity. In J. Clippinger (Ed.), *The Biology of Business: Decoding the Natural Laws of Enterprise*: 153-179. San Francisco, CA: Jossey-Bass Publishers.

Stern, R. N., & Barley, S. R. 1996. Organizations and Social Systems: Organization Theory's Neglected Mandate. *Administrative Science Quarterly*, 41: 146-162.

Stewart, D. 2005. Social Status in an Open Source Software Community. *American Sociological Review*, 70(5): 823-842

- Stinchcombe, A. L. 1959. Bureaucratic and Craft Administration of Production: A Comparative Study. *Administrative Science Quarterly* 4: 168-187.
- Stinchcombe, A. L. 1965. Social Structure and Organizations. In J. G. March (Ed.), *Handbook of Organizations*: 142-193. Chicago: Rand McNally & Company.
- Strauss, A., & Corbin, J. 1990. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. (2nd ed.). Thousand Oaks, CA: Sage Publications.
- Swidler, A. 1979. *Organization without Authority: Dilemmas of Social Control in Free Schools*. Cambridge, Mass: Harvard University Press.
- Taggar, S., Hackett, R., & Saha, S. 1999. Leadership emergence in autonomous work teams: Antecedents and outcomes. *Personnel Psychology*, 52: 899-926.
- Thompson, J. D. 1967. Organizations in action; Social science bases of administrative theory. *History and Development of Organizational Behavior*, 6(192). New York: McGraw-Hill.
- Van Maanen, J., & Barley, S.R. 1984. Occupational Communities: Culture and Control in Organizations. *Research in Organizational Behavior*, 6: 287-365.
- von Hippel, E. 2005. *Democratizing Innovation*. Cambridge, Mass: MIT Press.
- von Hippel, E., G., & von Krogh, G. 2003. Open Source Software and the "Private-Collective" Innovation Model: Issues for Organization Science. *Organization Science*, 14: 209-223.
- von Krogh, G., Spaeth, S., & Lakhani, K. R. 2003. Community, Joining, and Specialization in Open Source Software Innovation: A Case Study. *Research Policy*, 32: 1217-1241.
- Weber, M. 1946. In G. Hansand & M. C. Wright (Eds.), *Bureaucracy*. New York: Oxford University Press.
- Weber, M., Guenther, R., & Wittich, C. 1978. *Economy and society: An outline of interpretive sociology*, 2: 24-78. Berkeley: University of California Press.
- Weick, K.E. 1979. *The social psychology of organizing*, 2d ed edn. Reading, MA: Addison-Wesley Pub. Co.
- Weisband, S., & Atwater, L. 1999. Evaluating self and others in electronic and face-to-face groups. *Journal of Applied Psychology*, 84: 632-639.
- Wellman, B., Salaff, J., Dimitrova, D., Garton, L., Gulia, M., & Haythornthwaite, C. 1996. Computer Networks as Social Networks: Collaborative Work, Telework, and Virtual Community. *Annual Review of Sociology*, 22: 213-238.

- Wellman, B., & Gulia, M. 1999. Net Surfers Don't Ride Alone: Virtual Communities as Communities. In B. Wellman (Ed.), *Networks in the Global Village*. Boulder, CO: Westview.
- Wenger, E. 1998. *Communities of Practice: Learning, Meaning, and Identity*. Cambridge, UK: Cambridge University Press.
- Wenger, E., 2000. Communities of Practice and Social Learning Systems. *Organization*, 7: 225-246.
- Whyte, W. F., & Whyte, K. K. 1988. *Making Mondragon: The Growth and Dynamics of the Worker Cooperative Complex*. It Blau haca, NY: ILR Press.
- Willer, D. E. 1967. Max Weber's missing authority type. *Sociological Inquiry*, 37: 231-39.
- Williams, R. L., & Cothrel, J. 2000. Four Smart Ways to Run Online Communities. *Sloan Management Review*, 41: 81-91.
- Yin, R. K. 1994. *Case study research: Design and methods* (2nd ed.). Beverly Hills, CA: Sage Publishing.
- Yoo, Y., & Alavi, M. 2004. Emergent leadership in virtual teams: What do emergent leaders do? *Information and Organization*, 14: 27-28.
- Zucker, L. G. 1977. The Role of Institutionalization in Cultural Persistence. *American Sociological Review*, 42: 726-743.