

The governance of free/open source software projects: monolithic, multidimensional, or configurational?

M. Lynne Markus

Published online: 14 June 2007
© Springer Science+Business Media B.V. 2007

Abstract This paper presents the results of a qualitative review and synthesis of the literature on open source governance, addressing four key questions: (1) How has open source software (OSS) governance been defined? (2) Has the phenomenon of OSS governance been conceptualized as a monolithic or multidimensional phenomenon? (3) What purposes is OSS governance hypothesized to serve? and (4) What are the dimensions of OSS governance, and how are these dimensions related to each other? The results of the review suggest a framework for future comparative and case study research on OSS governance, and they provide a basis for comparison with research on the governance of other distributed, community-based forms of content and creation.

Keywords Collective action · Common pool resources/public goods · Coordination theory · Governance · Institutional theory · Motivation · Open source software

1 Introduction

A substantial base of scholarship on open source software (OSS) governance now exists, including numerous cases studies and several cross sectional and quantitative analyses. The time has come to assess the findings and consider the implications for future research on the topic. This paper presents the results of a qualitative review and synthesis of the literature on open source governance. Four key questions are addressed. First, how has OSS governance been defined? Second, has the phenomenon of OSS governance been conceptualized as a monolithic or multidimensional phenomenon? Thirdly, what purposes has OSS governance been hypothesized to serve? And, finally, if OSS governance is considered a

M. L. Markus (✉)
Bentley College, 175 Forest Street, Waltham, MA 02452, USA
e-mail: mlmarkus@bentley.edu

multidimensional phenomenon, what are its dimensions, and are they independent of each other or interrelated in configurations?

Each of these questions is addressed below. In the process of answering the questions, a framework for future comparative and case study research on OSS governance is proposed. The results of this review can also serve as a basis for comparison with, and generalization of, research on the governance of other distributed, community-based forms of content and creation, such as wikis.

2 How has OSS governance been defined?

Meaningful syntheses of research results rely on a certain degree of definitional commonality for core concepts. Therefore, the first question posed in this review is: how have researchers defined OSS governance?

As Lynn et al. (2001) observed for the context of public administration, writers on the OSS movement have rarely defined the term OSS governance precisely. For example, empirical research on OSS governance has examined such diverse structures, rules, practices, and norms as the following:

- Impersonal institutional mechanisms such as software licenses (de Laat 2005; Franck and Jungwirth 2002) and not-for-profit foundations (O'Mahony 2005).
- Structures of roles and responsibilities (Mockus et al. 2002; Nakokoji et al. 2002) including decisions taken by project leaders like Linus Torvalds about changes in operating procedures (Shaikh and Cornford 2003b).
- Formal and informal rules about verifying developer identity (O'Mahony & Ferraro, forthcoming), assigning programming tasks to people (Crowston et al. 2005), according developer status to newcomers (von Krogh et al. 2003), and the peer review of code (Lee and Cole 2003).
- IT-enabled communication processes such as vision documents, bug reporting, voting, and version/release control (Dinh-Trong and Bieman 2004; Scacchi 2002; Yamauchi et al. 2000).
- Code signing (Lee and Cole 2003) and reputation signaling practices (von Krogh et al. 2003).
- Outcome control, self control, and social control (Lattemann and Steiglitz 2005; Xu et al. 2005), and
- Norms of reciprocity, such as a widely held belief in the value of sharing code with others (Shah 2006).

Lack of an accepted definition leaves it up to each researcher to decide what OSS governance means, making it challenging for others to assess the research findings. I propose, therefore, adapting Lynn et al.'s definition (2001, p. 6) of public sector governance for the OSS context. Thus, OSS governance can be defined as *the means of achieving the direction, control, and coordination of wholly or partially autonomous individuals and organizations on behalf of an OSS development project to which they jointly contribute.*

The advantage of this definition is that it does not prejudge the answers to four important questions that have been discussed in the governance literature more broadly, but not yet extensively explored in OSS governance research:

- Which is more important: OSS governance *structures* or OSS governance *processes* (Child and Rodrigues 2003), or are they equally important?
- Is OSS governance primarily *informal* (i.e., enacted through shared norms and social control), *formally documented* (in written policies and constitutions) or “*encoded*” in technology (e.g., voting software, version control software)? Or is it all three at once?
- Is the source of direction, control, and coordination in OSS governance *external* to individuals (e.g., licenses, incentives), *internal* (e.g., intrinsic motivations, self-control), or both?
- How important is *trust* in OSS governance (Gallivan 2001; Rota et al. 2002) in comparison to, for example, *monitoring and control* (cf. Ostrom 2005)? And, if so, what is the process by which trust operates in OSS governance?

My conclusion after reviewing the prior research is that OSS governance can be usefully characterized using all of these attributes (structure, process; formal in varying degrees; external, internal; trusting, monitoring), if not in every case, at least across the range of cases. In particular, I see no inherent contradiction on the “trust versus control” dimension. Trust and control or trust and monitoring/punishment have been observed to coexist in a variety of other contexts (O’Leary et al. 2002; Ostrom 1999).

3 OSS governance: a monolithic or multidimensional phenomenon?

A second core conceptual issue is whether the phenomenon of OSS governance should be conceptualized as a unitary phenomenon that manifests as a few “types” or “modes” or whether it should be understood as composed of numerous dimensions with diverse manifestations. Both perspectives can be identified in the OSS governance literature.

In the “monolithic” camp are scholars who describe “open source” as a distinct organizational form, mode, or type that can cleanly be differentiated from a few other clearly delineated types. For example, some writers contrast OSS *development* with the traditional way of developing software. Among others, Raymond (1998) and Scacchi (2004) emphasized the differences in the principles and processes of OSS development relative to proprietary development, and Crowston et al. (2005) argued that OSS development differs from proprietary development in that task assignment is voluntary rather than mandated by leaders. By contrast, von Hippel and von Krogh (2003) opined that open source is a distinctly different way of *conducting innovation*: OSS has some characteristics of both the private model of commercial innovation and the collective model of science, making OSS a private-collective hybrid form. OSS has also been described as a unique mode of *governing productive activity* (in the economic sense). For example, Demil and Lecocq (2006)

presented “bazaar governance” as an alternative to the familiar economic governance modes of markets, hierarchies, and networks.

The basic assumption or conclusion of the body of work that views OSS governance in monolithic terms is that the OSS form is operationally viable, economically efficient, or technically superior in comparison to other forms. Often in these analyses the open source license for intellectual property protection is cited as the essential characteristic or mechanism typifying the OSS mode and differentiating it from alternatives.

In the “multidimensional” camp are writers who emphasize the many types and variations of governance mechanisms within the broad category of OSS projects. These scholars have examined the diversity to be found in:

- The terms of OSS licenses, e.g., whether or not they correspond to the requirements of the OSI (de Laat 2005)
- The legal organization structures of OSS projects, e.g., the not-for-profit foundation, and their legal responsibilities (O’Mahony 2005)
- The relationships between vendor organizations and OSS communities, e.g., parasitic, symbiotic, or commensalitic patterns (Dahlander and Magnusson 2005)
- The role structures of OSS communities, such as the relative sizes of the project leadership team, the team of developers with committing privileges, the community of bug reporters, etc. (Mockus et al. 2002; Nakokoji et al. 2002), and
- The operation of technical business processes, including requirements determination, code inspection, and release coordination (Mockus et al. 2002; Scacchi 2002).

Although the multidimensional view of governance adds empirical fidelity to studies of OSS projects, it also adds complexity that calls for more theoretical and empirical analysis. One wonders, for example, whether the many dimensions of governance are independent of each other or are related in some way.

In short, the literature exhibits a wide range of views about what constitutes governance. In some literature, governance is portrayed as a single, unified phenomenon, most frequently concretized as OSS licensing mechanisms. In other literature, the concept of governance has been operationalized as many different phenomena, including organizational structures, role structures, and technical and managerial processes. This diversity of perspectives on governance makes it challenging for readers to draw firm conclusions from the growing body of literature.

4 What is the purpose of OSS governance?

The literature exhibits different assumptions, implicit or explicit, about the goals or functions served by OSS governance. Three main positions can be distinguished. In the first, OSS governance is seen as the solution to collective action dilemmas about individuals’ or organizations’ incentives to contribute to, or appropriate the benefits of, OSS development. A second position holds that OSS governance is the solution

to coordination problems during OSS development. A third, emerging, position holds that, even after individuals or organizations have decided to participate in OSS development, OSS governance mechanisms may have inherent motivational potential, affecting decisions about which of several OSS projects to join or about how much effort to contribute to any one project.

4.1 Solving collective action dilemmas

As many authors have noted, a central dilemma of OSS development is what motivates developers to contribute their time and skills “for free”. Much research effort has been devoted to the identification of *developers’ motivations and private benefits* (Hann et al. 2004; Hann et al. 2002; Hars and Ou 2002 ; Hertel et al. 2003; Maass 2004; Roberts et al. 2006; Shah 2006; von Krogh et al. 2003). Other research has explicated how governance mechanisms such as OSS licenses (Franck and Jungwirth 2003; Lee and Cole 2003) and not-for-profit foundations (O’Mahony 2003, 2005) *provide appropriate incentives* for participation by developers and other investors.

Thus, OSS governance can be seen as a response or solution to *social or collective action dilemmas* created by lack of incentives for participation. These social dilemmas have sometimes been described in terms of “free riding” by individuals and/or firms with respect to either OSS development or OSS commercialization; other times, they have been discussed in terms of the “tragedy of the commons” (Baldwin and Clark 2006; de Laat 2005; Franck and Jungwirth 2002; O’Mahony 2003; von Hippel and von Krogh 2003). Debate exists about which social dilemmas may apply to the open source content (O’Mahony 2003) and about the extent to which they apply (de Laat 2005). Common to much of this literature, however, is the hypothesis that OSS governance solves motivational problems by giving people or organizations control over the ownership of the output of OSS development work, that is, ownership of OSS software and/or its benefits.

4.2 Solving development coordination problems

By contrast to research that views OSS governance as a solution to social or collective action dilemmas, some researchers portray OSS governance as a solution to the *routine challenges of coordinating interdependence* in getting OSS development work done (Crowston et al. 2005; Jensen and Scacchi 2004; Jørgensen 2001; Mockus et al. 2002; Sagers et al. 2004; Scacchi 2004; Shaikh and Cornford 2003a; Xu et al. 2005). This work parallels the literature on governance in interorganizational contexts, which has likewise emphasized the importance of governance in coping with routine operational coordination problems, in addition to solving ownership dilemmas (cf. Grandori 1997; Gulati and Singh 1998; Sobrero and Schrader 1998).

For example, one question that arises in the OSS context, according to Lee and Cole (2003) is how individual contributions to OSS projects can result in high quality OSS products. There is no assumption here that all OSS software is of high quality, only that the presence of many casual developers appears to increase the

likelihood that software quality will be poor. The relevant governance mechanisms for the prevention and correction of software defects include: *peer review* of software (Lee and Cole 2003), coordination of the *release schedule* (German 2003; Jørgensen 2001; Mockus et al. 2002; Moon and Sproull 2000; Scacchi 2004), the process of *requirements determination* (Scacchi 2002), the activity of *attracting developers* to particular development tasks (Crowston et al. 2005; Dahlander and Magnusson 2005; Jørgensen 2001), etc.

In other words, both in the social dilemmas literature and in the operational coordination literature, OSS governance is understood as a solution to a problem. However, whereas in the social dilemmas literature the problem is motivation to contribute and the solution is ownership of the software product, in the OSS coordination literature, the problem is loss of operational control, and the solution is techniques for managing the process of OSS development work. Thus, OSS governance can be conceptualized at two levels: (1) the level of community establishment and project constitution (e.g., defining goals, attracting investors) and (2) at the level of managing operational software development processes (e.g., managing code quality).

4.3 Creating a (better?) climate for contributors

A third view, not well developed in the literature, is that OSS governance might be seen by participants as an end in itself rather than as a solution to problems of collective action or of coordination. The insightful question posed by Franke and von Hippel (2003) about the kind of organization to which individuals will want to donate applies, not only to the decision of whether or not to contribute to an open source project at all, but also to the decision of *to which OSS project to contribute*, when there is a choice. For instance, Shah (2006) found that voluntary participants in “gated” (i.e., sponsored or restricted license) OSS communities were openly critical of their projects’ governance structures. In addition, those participants expressed preferences for contributing to OSS projects that were more “open”.

In other contexts, Manville and Ober (2003a, 2003b) discussed the motivational potential of democratic governance. Consider the analogy of choosing between jobs that differ in pay and working conditions. Although some job choosers may always prefer jobs with higher pay, others may take lower paying jobs that allow them greater opportunity to influence organizational decisions. From this point of view, one may hypothesize that the governance of OSS projects is motivationally important, not only because it can provide private benefits in terms of salary and career (Roberts et al. 2006), reputation, control over technology, or learning (von Krogh et al. 2003), but also because it can create what organization behavior theorists refer to as positive (or negative) *organizational climate* (Denison 1996). Organizational climate can promote (or hinder) individuals’ motivation to work, *independent of* benefits such as monetary incentives and recognition. In fact, social psychological research has shown that extrinsic rewards can actually *reduce* individuals’ intrinsic motivation and their contributions to a common resource (Ling et al. 2005). This finding suggests that a good OSS project climate engendered by democratic governance might actually be more effective at motivating contributions

than formal rewards and private benefits. (Note that von Krogh, Haefliger et al. 2003's "control over technology" dimension may be related to what I am calling OSS project climate.)

Pursuing this line of reasoning, one may hypothesize that OSS projects with governance characterized by greater degrees of democratic "openness", that is, perceived opportunities for participation, transparency of governance, and the accountability of decision makers (German 2003; Vincent and Camp 2004; West, forthcoming) are more motivational, that is, more successful at attracting participants, than OSS projects with less democratic governance. This is certainly not to say that all OSS project governance actually *is* open or democratic, nor that OSS projects generally *are* more open and democratic than proprietary projects. How democratic OSS governance is and whether democratic OSS governance is more motivational than autocratic OSS governance and/or proprietary development are empirical questions—and they are important questions that have not yet been answered in OSS governance research.

In sum, the literature suggests three different purposes for OSS governance, solving collective action problems, solving coordination problems in software development, and creating a better climate for contributors. There is nothing to say that OSS governance cannot address all of these purposes simultaneously. However, it does not seem likely that a single governance mechanism, such as an OSS license, could do so effectively. Thus, an inclusive view of the purposes of governance is more compatible with a multidimensional perspective on OSS governance than with a monolithic perspective. Oliver (1990) makes similar points about the multiple purposes of organizational governance.

5 What are the dimensions of OSS governance and how are they related?

If one assumes, then, that OSS governance is a multidimensional phenomenon that addresses a variety of purposes, the questions arise: what are the important dimensions or categories of dimensions of OSS governance, and how are they related? Do the dimensions vary independently of each other, such that many small variations can be observed, or are they interrelated in some way, leading to a limited number of types or *configurations*? Many writers on the topics of organizational governance (Greenwood and Hinings 1993; Meyer et al. 1993; Miller 1996; Oliver 1990) and of governance in situations involving social dilemmas (Ostrom 1986, 1999, 2005) have proposed thinking of governance in terms of configurations. This direction should be pursued in OSS governance research.

As an example of potential OSS governance configurations, consider O'Mahony and West's (2005) proposal of two major types—organic forms (community initiated) and synthetic forms (sponsored or controlled by for profit firms). The organic and synthetic types were differentiated on numerous dimensions (intellectual property ownership, type of license, etc.), some of which co-varied. Note that such types, comprised of multiple different, non-linearly related dimensions, are conceptually different from unidimensional "monolithic" types.

A first step toward viewing OSS governance configurationally is to identify the dimensions of OSS governance that might be involved in any observed configurations. Can the many OSS governance mechanisms identified in prior research be grouped in meaningful ways? Several theorists pursued such an approach in other substantive domains (Oliver 1990; Ostrom 2005). Because they were dealing with different governance contexts (interorganizational relationships and the management of “common pool” natural resources like fisheries and forests, respectively), the specific categories those researchers posited are unlikely to apply to OSS governance. However, their general methodology of identifying logical groupings of rules related to governance purposes and outcomes seems useful and relevant.

My review of the empirical research OSS development suggested that the elements of OSS governance could be grouped into at least the following six categories of formal and informal structures and rules:

- Ownership of assets—this category includes intellectual property licenses and formal legal organizational structures (e.g., foundations).
- Chartering the project—this term refers to statements of vision about the goals of the project, what the software product should look like, etc. (German 2003; Nakokoji et al. 2002; Scacchi 2002).
- Community management—this category involves rules about who can be members, how their identity will be verified, what roles they can play, how they can change roles, etc. (Mockus et al. 2002; O’Mahony & Ferraro, forthcoming; von Krogh, Spaeth et al. 2003).
- Software development processes—in this category are structures and rules that address the important operational tasks of development, such as requirements elicitation, assignment of people to tasks, processes for managing software changes, release control, etc. (Jørgensen 2001; Shaikh and Cornford 2003a; Yamauchi et al. 2000).
- Conflict resolution and rule changing—this category involves rules and procedures for resolving conflict and for creating new rules (Jensen and Scacchi 2004; Shaikh and Cornford 2003b), and
- Use of information and tools—in this category are rules about how information will be communicated and managed and how tools and repositories will be used (Jørgensen 2001; Shaikh and Cornford 2003a; Yamauchi et al. 2000).

Thus, prior literature in other contexts suggests the value of viewing governance in terms of non-linear configurations of various categories of rule sets. Prior research on OSS governance mechanisms suggests at least six categories of rules and structures that may prove to be important in empirical research on OSS governance configurations. As discussed below, the resulting framework can guide future research in several ways.

6 Discussion

First, the framework proposed above can help assess the completeness of our knowledge of OSS governance mechanisms. In my earlier discussion of the

definition of OSS governance, I argued that OSS governance could include: (1) both structures and processes, (2) informal, formal, and encoded rules, (3) externally applied as well as internalized rules, and (4) mechanisms of both trust and verification/control. By looking for these attributes within each of the six sets of rules, future researchers may be able to discover new OSS governance mechanisms.

Second, the framework can serve as a basis for testing previously proposed OSS governance configurations or for identifying new ones. For instance, Dahlander and Magnusson (2005), O'Mahony and West (2005), and Shah (2006) have each proposed different configurations of OSS governance mechanisms. Would these typologies mesh with each other in an empirical study conducted with expanded set of governance categories and dimensions? Would different configurations be identified? The answers to these questions await future research.

Another question that can be addressed with the aid of this framework concerns the relationship between OSS governance mechanisms and the specific purposes that governance is hypothesized to serve. For example, one might propose that the sets of governance rules are related in an overlapping way to the three purposes of governance discussed above, as presented in Table 1. Pursuing this line of thought may identify situations in which particular governance mechanisms are helpful for one purpose but hurtful for another. Such research could support better policy guidance to those who set up and lead OSS communities.

Perhaps most important, the framework might also prove useful for answering questions about the relationship between OSS governance and OSS project success. Do the observed variations in OSS project governance mechanisms matter? That is, are these variations systematically *related to the success or failure of OSS projects*? If so, how do they contribute to success, and which governance mechanisms are more effective than others, under what conditions? In her research on common-pool resources, Ostrom (1999) found no evidence that any particular rules were more effective than others. She did however find that the *absence of rules in particular rule categories* was associated with ineffectiveness. Does a similar pattern hold across OSS projects? For example, Jørgensen (2001) noted that the presence of rules about how incremental changes were to be made coupled with the absence of rules about how major development work was to be approached resulted in incrementalism in the FreeBSD project. Future research is needed to identify whether the presence, absence, or specific content of rules in any or all of the six categories listed above is consequential for OSS project effectiveness.

Table 1 Hypothetical relationship between OSS governance purposes and OSS governance categories

OSS governance purposes vs. oss governance dimensions	Solve collective action dilemmas	Solve coordination problems	Create a climate for contributors
Ownership rules	x		x
Chartering rules	x		x
Community rules	x	x	x
Software development process rules		x	
Conflict rules and rules about rules		x	x
Information and tools rules		x	

Another set of questions that the framework can help answer relates to the origins and evolution of OSS governance. Are the observed variations in governance across OSS projects mainly accidents of history or do they correspond systematically to *differences in the context or nature* of OSS projects? To what extent are OSS governance structures *consciously designed* to address specific environmental or technical challenges versus *evolving organically* through path dependence and institutional pressures?

And finally, the framework can serve as a point of comparison with findings from research on other contexts to which the notions of openness and community management have been applied (O'Mahony, this issue), including organizational governance (Manville and Ober 2003b; Markus et al. 2000) and open content creation environments such as wikis. The ability to generalize to other contexts can be facilitated by the availability of conceptual frameworks such as the one proposed here.

7 Conclusion

The major conclusion to be drawn from this review is that the study of OSS governance is no longer virgin territory. Although there are many unanswered questions, much is already known, and there is no excuse for future work that does not build on past foundations.

In my opinion, future research should be designed with an explicitly comparative mindset (Ragin 1987). Even researchers who conduct single, in-depth case studies of individual OSS communities should proceed in a way that would support the activities of subsequent researchers aiming to identify OSS governance configurations or to develop general explanations of OSS governance and outcomes. That will require describing OSS governance, as well as the context and outcomes, along as many dimensions as possible. To support such description, a “descriptive framework” for case study research (Yin 1999) can easily be constructed from the material presented in this review. In addition, the review has advanced a number of hypotheses that can be tested in future comparative work. In short, much important research on OSS governance remains to be done, but past research has given us a strong foundation and clear directions for future work.

References

- Baldwin, C. Y., & Clark, K. B. (2006). The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science*, 52(7), 1116–1127.
- Child, J., & Rodrigues, S. B. (2003). Corporate governance and new organizational forms: Issues of double and multiple agency. *Journal of Management and Governance*, 7(4), 337–360.
- Crowston, K., Wei, K., Li, Q., Eseryel, U. Y., & Howison, J. (2005). *Coordination of free/libre open source software development*. Paper presented at the Twenty-Sixth International Conference on Information Systems, Las Vegas, NV.
- Dahlander, L., & Magnusson, M. G. (2005). Relationships between open source software companies and communities: Observations from Nordic firms. *Research Policy*, 34(4), 481–493.

- de Laat, P. B. (2005). Copyright or copyleft?: An analysis of property regimes for software development. *Research Policy*, 34(10), 1511–1532.
- Demil, B., & Lecocq, X. (2006). Neither market nor hierarchy or network: The emergence of bazaar governance. *Organization Studies*, 27(10), 1447–1466.
- Denison, D. R. (1996). What is the difference between organizational culture and organizational climate? A native's point of view on a decade of paradigm wars. *Academy of Management Review*, 21(3), 619–654.
- Dinh-Trong, T., & Bieman, J. M. (2004). *Open source software development: A case study of FreeBSD*. Paper presented at the 10th International Symposium on Software Metrics, Como, IT.
- Franck, E., & Jungwirth, C. (2002). *Reconciling investors and donators—The governance of open source*. Zurich: Universitat Zurich, Lehrstuhl fur Unternehmensfuehrung and -politik.
- Franck, E., & Jungwirth, C. (2003). Reconciling rent-seekers and donators—The governance structure of open source. *Journal of Management and Governance*, 7(4), 401–421.
- Franke, N., & von Hippel, E. (2003). Satisfying heterogeneous user needs via innovation toolkits: The case of Apache security software. *Research Policy*, 32(7), 1199–1215.
- Gallivan, M. J. (2001). Striking a balance between trust and control in a virtual organization: A content analysis of open source software case studies. *Information Systems Journal*, 11, 277–304.
- German, D. M. (2003). The GNOME project: A case study of open source, global software development. *Software Process Improvement and Practice*, 8, 201–215.
- Grandori, A. (1997). An organizational assessment of interfirm coordination modes. *Organization Studies*, 18(6), 897–925.
- Greenwood, R., & Hinings, C. R. (1993). Understanding strategic change: The contribution of archetypes. *Academy of Management Journal*, 36(3), 1052–1081.
- Gulati, R., & Singh, H. (1998). The architecture of cooperation: Managing coordination costs and appropriation concerns in strategic alliances. *Administrative Science Quarterly*, 43(4), 781–814.
- Hann, I.-H., Roberts, J., & Slaughter, S. (2004). *Why developers participate in open source software projects: An empirical investigation*. Paper presented at the Twenty-Fifth International Conference on Information Systems, Washington, DC.
- Hann, I.-H., Roberts, J., Slaughter, S., & Fielding, R. (2002). *Economic incentives for participating in open source software projects*. Paper presented at the Twenty-Third International Conference on Information Systems, Barcelona, SP.
- Hars, A., & Ou, S. (2002). Working for free? Motivations for participating in open-source projects. *International Journal of Electronic Commerce* 6(3), 25–39.
- Hertel, G., 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), 1159–1177.
- Jensen, C., & Scacchi, W. (2004). *Collaboration, leadership, control, and conflict negotiation in the Netbeans.org community*. Paper presented at the 4th Workshop on Open Source Software Engineering W8S Workshop - 26th International Conference on Software Engineering, Edinburgh, Scotland, UK.
- Jørgensen, N. (2001). Putting it all in the trunk: Incremental software development in the FreeBSD open source project. *Information Systems Journal*, 11(4), 321–336.
- Lattemann, C., & Steiglitz, S. (2005). *Framework for governance in open source communities*. Paper presented at the 38th Hawaii International Conference on Systems Sciences, Kauai, HI.
- Lee, G. K., & Cole, R. E. (2003). From a firm-based to a community-based model of knowledge creation: The case of Linux kernel development. *Organization Science*, 14(6), 633–649.
- Ling, K., Beenen, G., Ludford, P., Wang, X., Chang, K., Li, X., et al. (2005). Using social psychology to motivate contributions to online communities. *Journal of Computer-Mediated Communication*, 10(4), article 10. Retrieved October 26, 2006, from <http://jcmc.indiana.edu/vol10/issue4/ling.html>.
- Lynn, J., Laurence, E., Heinrich, C. J., & Hill, C. J. (2001). *Improving governance: A new logic for empirical research*. Washington, D.C.: Georgetown University Press.
- Maass, W. (2004). *Inside an open source software community: Empirical analysis on individual and group level*. Paper presented at the 4th Workshop on Open Source Software Engineering at the 26th International Conference on Software Engineering (ICSE04), Edinburgh, UK.
- Manville, B., & Ober, J. (2003a). Beyond empowerment: Building a company of citizens. *Harvard Business Review*, 81(1), 48–53.
- Manville, B., & Ober, J. (2003b). *A company of citizens: What the world's first democracy teaches leaders about creating great organizations*. Boston, MA: Harvard Business School Press.

- Markus, M. L., Manville, B., & Agres, C. (2000). What makes a virtual organization work - Lessons from the open source world. *Sloan Management Review*, 42(1 Fall), 13–26.
- Meyer, A. D., Tsui, A. S., & Hinings, C. R. (1993). Configuration approaches to organizational analysis. *Academy of Management Journal*, 36(6), 1175–1195.
- Miller, D. (1996). Configurations revisited. *Strategic Management Journal*, 17, 505–512.
- 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(3), 309–346.
- Moon, J. Y., & Sproull, L. (2000). Essence of distributed work: The case of the Linux kernel. *First Monday*, 5(11). Retrieved October 15, 2006, from http://www.firstmonday.org/issues/issue5_11/moon/.
- Nakokoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K., & Ye, Y. (2002). *Evolution patterns of open-source software systems and communities*. Paper presented at the International Workshop on Principles of Software Evolution, Orlando, FL.
- O’Leary, M., Orlikowski, W. J., & Yates, J. (2002). Distributed work over the centuries: Trust and control in the Hudson’s Bay Company, 1670–1826. In P. J. Hinds & S. Kiesler (Eds.), *Distributed work* (pp. 27–54). Cambridge, MA: MIT Press.
- O’Mahony, S., & Ferraro, F. (forthcoming). Managing the boundary of an ‘open’ project. In J. Padgett & W. Powell (Eds.), *Market emergence and transformation*. Cambridge, MA: MIT Press.
- O’Mahony, S., & West, J. (2005, August). *What makes a project open source? Migrating from organic to synthetic communities*. Paper presented at the Paper presented at Academy of Management Annual Meeting, Honolulu, HI.
- O’Mahony, S. (2003). Guarding the commons: How community managed software projects protect their work. *Research Policy*, 32(7), 1179–1198.
- O’Mahony, S. (2005). Nonprofit foundations and their role in community-firm software collaboration. In J. Feller, B. Fitzgerald, S. A. Hissan & K. R. Lakhani (Eds.), *Perspectives on free and open source software* (pp. 393–413). Cambridge, MA: The MIT Press.
- Oliver, C. (1990). Determinants of interorganizational relationships: Integration and future directions. *Academy of Management Review*, 15(2), 241–265.
- Ostrom, E. (1986). An agenda for the study of institutions. *Public Choice*, 48(1), 3–25.
- Ostrom, E. (1999). Coping with tragedies of the commons. *Annual Review of Political Science*, 2, 493–535.
- Ostrom, E. (2005). *Understanding institutional diversity*. Princeton, NJ: Princeton University Press.
- Ragin, C. C. (1987). *The comparative method: Moving beyond qualitative and quantitative strategies*. Berkeley, CA: University of California Press.
- Raymond, E. S. (1998). *The cathedral and the bazaar*. June, 1997. Retrieved May 9, 2006, from <http://www.catb.org/~esr/writings/cathedral-bazaar/>.
- Roberts, J. A., Hann, I.-H., & Slaughter, S. A. (2006). Understanding the motivations, participation and performance of open source software developers: A longitudinal study of the Apache projects. *Management Science*, 52(7), 984–999.
- Rota, S., von Wartburg, M., & Osterloh, M. (2002). *Trust and commerce in open source - A contradiction?*. Zurich, CH: Institute for Research in Business Administration.
- Sagers, G. W., Wasko, M. M., & Dickey, M. H. (2004). *Coordinating efforts in virtual communities: Examining network governance in open source*. Paper presented at the Tenth Americas Conference on Information Systems, New York, NY.
- Scacchi, W. (2002). Understanding the requirements for developing open source software systems. *IEEE Proceedings–Software*, 149(1), 24–39.
- Scacchi, W. (2004). Free and open source development practices in the game community. *IEEE Software*(January/February), 59–66.
- Shah, S. K. (2006). Motivation, governance and the viability of hybrid forms in open source software development. *Management Science*, 52(7), 1000–1014.
- Shaikh, M., & Cornford, T. (2003a). *Version control software for knowledge sharing, innovation and learning in OS*. Paper presented at the Workshop on Open Source Software Movements and Communities at the International Conference on Communities and Technologies, Amsterdam, NL.
- Shaikh, M., & Cornford, T. (2003b). *Version management tools: CVS to BK in the Linux kernel*. Paper presented at the Taking Stock of the Bazaar: Proceedings of the 3rd Workshop on Open Source Software Engineering at the International Conference on Software Engineering (ICSE), Portland, OR.

- Sobrero, M., & Schrader, S. (1998). Structural inter-firm relationships: A meta-analytic approach. *Organization Studies*, 19(4), 585–615.
- Vincent, C., & Camp, J. (2004). Looking to the Internet for models of governance. *Ethics and Information Technology*, 6, 161–173.
- von Hippel, E., & von Krogh, G. (2003). Open source software and the “private-collective” innovation model: Issues for organization science. *Organization Science*, 14(2), 209–223.
- von Krogh, G., Haefliger, S., & Spaeth, S. (2003). *Collective action and communal resources in open source software development: The case of Freenet*. CH: University of St. Gallen.
- von Krogh, G., Spaeth, S., & Lakhani, K. R. (2003). Community, joining, and specialization in open source software innovation: A case study. *Research Policy*, 32(7), 1217–1241.
- West, J. (2007). The economic realities of open standards: Black, white and many shades of gray. In S. Greenstein & V. Stango (Eds.), *Standards and public policy* (pp. 87–122). Cambridge, UK: Cambridge University Press.
- Xu, B., Xu, Y., & Lin, Z. (2005). *Control in open source software development*. Paper presented at the Eleventh Americans Conference on Information Systems, Omaha, NE.
- Yamauchi, Y., Yokozawa, M., Shinohara, T., & Ishida, T. (2000). *Collaboration with lean media: How open-source software succeeds*. Paper presented at the Conference on Computer Supported Cooperative Work, Philadelphia, PA.
- Yin, R. K. (1999). *Case study research: Design and method* (3rd ed.). Thousand Oaks: Sage.

Author Biography

M. Lynne Markus is the John W. Poduska, Sr. Professor of Information and Process Management at Bentley College. Professor Markus’s three primary research areas are enterprise and inter-enterprise systems, IT and organization change, and knowledge management. She was named Fellow of the Association for Information Systems in 2004.