Association for Information Systems
AIS Electronic Library (AISeL)

2008

Toward a Theory of Socialization in Open Source Software Communities: A Symbolic Interactionist Perspective

Jijesh Devan
Queen's University, jdevan@business.queensu.ca

Dany Di Tullio
Queen's University, dditullio@business.queensu.ca

Follow this and additional works at: http://aisel.aisnet.org/amcis2008

Toward a Theory of Socialization in Open Source Software Communities: A Symbolic Interactionist Perspective

Jijesh Devan

Queen's School of Business

Queen's University

jdevan@business.queensu.ca

Dany Di Tullio

Queen's School of Business

Queen's University

dditullio@business.queensu.ca

ABSTRACT

Recent literature reviews highlight a gap in our understanding of how members socialize into virtual communities. Motivated by this shortcoming, we seek to understand how members socialize into open source software (OSS) communities, which are both virtual and normative in nature. Using the symbolic interactionist perspective, we define socialization and explicate the socialization process in OSS communities. Specifically, we define socialization as the extent of correspondence between the personal meanings of individual members and the shared meanings of the community, which emerge from the negotiated experiences of members. In addition, we posit that the socialization process consists of the interplay between individuals' meaning-making process, consisting of both action and personal meaning, and the negotiated shared meanings within the community.

Keywords

Open source software, open source software communities, socialization, symbolic interactionism.

INTRODUCTION

Recent reviews recognize the lack of theoretical and empirical work on member socialization in virtual groups, and communities (Ahuja and Galvin 2003, Galvin and Ahuja 2001, Maas 2004) . Interestingly, there has been a similar call from Lois (1999) for more research on socialization in normative organizations, such as volunteer communities. To close these research gaps, we investigate the socialization of members into volunteer and virtual software development communities, with a specific focus on open source software (OSS) communities. The research question that motivates this paper is: How do individuals socialize into OSS communities? In this paper, we define OSS communities as virtual, normative, knowledge-creating communities and explicate member socialization using the symbolic interactionist perspective (Blumer 1969).

As such, by drawing on this approach, we offer two contributions to research: (1) we define socialization in OSS communities, and (2) we offer a theoretical explication of the socialization process. We define socialization as the extent of correspondence, resulting from shared and negotiated experience, between the personal meanings of individual members and the shared meanings of the community. We also posit that the socialization process community members go through in OSS communities can be best understood as the interdependent interplay between *action*, comprised of both knowledge seeking and knowledge providing activities, and *meaning-making,* consisting of both the individual meaning-making activities and the shared meanings that evolve in the community

The remainder of this paper is organized as follows. The next section reviews research that sheds light on the different stages of membership in OSS communities. This review helps inform our own theorizing on OSS member socialization. This is followed by a discussion of symbolic interactionism as the theoretical lens we use to define socialization and explicate the socialization process in OSS communities. Member socialization in OSS communities is then defined and the process explained by drawing on the symbolic interactionist approach. Concluding thoughts are provided last.

OPEN SOURCE SOFTWARE AND SOCIALIZATION

This section offers a review of OSS research that addresses the various stages of membership individuals transition through when participating in an OSS community. This review helps inform our own theorizing on OSS member socialization by setting the stage for the following section in which we present our definition of socialization and our explication of the socialization process based on the symbolic interactionist perspective.

The open source model of software production is becoming an increasingly popular alternative to proprietary software development, with Gartner predicting that by the end of 2008, 95% of Global 2000 organizations will have implemented open source acquisition and management strategies (Driver 2005). However, there is much that is still not known about how

OSS communities function. Indeed, despite some very successful and highly publicized OSS projects, the fact remains that many projects often struggle with shortages in skilled contributors and suffer from high developer turnover rates (von Krogh et al. 2003). This makes it all the more important to understand how members socialize and become an integral & reliable part of OSS communities since many beneficial outcomes of successful socialization into volunteer communities have been underscored in prior research. For example, successful socialization into volunteer organizations has been shown to reduce member turnover rates (Mignerey et al 1995). As such, a better understanding of socialization in this context could undoubtedly help in the success of open source projects that struggle with issues like developer turnover and a lack of reliable contributions on the part of community members (von Krogh et al. 2003).

Research has identified a general pattern of activity members follow when becoming part of OSS communities. At first, we offer a brief description of the various stages individuals go through in the process of becoming members (see Table 1). Subsequently, we offer an elaborate description of the activities members undertake as they transition through these stages.

Von Krogh et al. (2003)	Mechanism	Ducheneaut (2005)	Mechanism	Adopted Typology
Lurker: Subscribes to the mailing lists and undertakes knowledge seeking activity by reading messages.	Opening script	Peripheral participant: Subscribe to mailing lists and seek knowledge by reading messages.	Opening script	**Lurker**
Participant: Participates in ongoing discussions, undertakes knowledge seeking and providing activities.	Joining script. They usually begin as debuggers and patchers.	Debugger and patcher: Participates in ongoing discussions	Joining script. They usually begin as debuggers and patchers. The level and type of activity gets them access to developer community.	**Participant**
Joiner: Has access to the developer community.	The level and type of activity gets them access to developer community.			**Joiner**
Newcomer: Has access to other developers and source code.	Specialization and contribution.	Developer: Has CVS access	Specialization, contribution, social interaction, allies.	**Developer**
Developer: Is contributing code to the project	Specialization, contribution, ability to garner allies, and maintain social interaction.			
Core developer / Administrator: Has commit privileges	N/A	Core developers / Administrators	N/A	**Core developers /Administrators**

Table 1. Stages of Membership in OSS Communities

Extant research on OSS communities demonstrates that new members start as lurkers. A lurker is someone who subscribes to the mailing list but is only observing the activity in the community (Ducheneaut 2005, von Krogh et al. 2003). The first act of posting, the opening post, signals the transition of a lurker to a participant (von Krogh et al. 2003). Participant members undertake various activities including debugging, testing, and code reviewing. When the participant makes a joining post, what von Krogh et al (2003) refer to as the "joining script", they transition to the joiner stage. Then, when joiners transition to become newcomers, they get access to the Concurrent Version System (CVS). The CVS is a popular configuration management tool for maintaining the source code of the software application. This privilege allows newcomers access to the source code, but they cannot commit changes to the final code. A newcomer becomes a core developer when the project administrator(s) give(s) him the responsibility of a full software module and the ability to commit (save) changes to the source code. The transition from a newcomer to a developer is a testimony to the ability of the member to become not just a software coder, but a craftsperson of relationships (Orr 1990).

The papers discussed above imply that progressing through the different membership stages consists of member socialization into OSS communities. However, they fail to explicate *how* the socialization process unfolds. Moreover, according to Ducheneaut (2005), current OSS research falls short when addressing socialization as its treats action and meaning-making discretely (e.g., von Krogh et al. (2003)). Therefore, we seek to address this omission using the symbolic interactionist perspective in the following sections.

SYMBOLIC INTERACTIONISM AS A THEORETICAL PERSPECTIVE

We believe symbolic interactionism is particularly appropriate to study socialization, especially as defined in this paper, because it emphasizes the dual role of both action and symbolic meaning-making in social situations. Therefore, we employ this perspective to understand the socialization process in open source software communities.

One of the underlying premises of symbolic interactionism is the symbolic nature of human interactions. This approach draws from the social constructionist position that views all social interaction as based on the meanings that individuals, groups, or communities attribute to objects and events in everyday life. By interacting with one another, people negotiate and assign meanings to different things. Both these processes of action and meaning-making, what we refer to as interactions, collectively shape the community's values, norms, and beliefs, or the shared symbolic meanings that emerge. Colton (1987, p. 346) provides a succinct, yet thorough definition of symbolic interactionism: "symbolic interactionism is based on the premise that human society is characterized by the use of symbols and meanings, and that the meanings of various social and nonsocial objects or symbols are derived though the interaction process." Therefore, one of the main goals of applying this perspective here is to understand how people interpret or make sense of their shared experience.

One of the strengths of the symbolic interactionism approach is its recognition that multiple meanings and realities pervade social life and its objective to search for and explicate these diverse interpretations rather than identify single shared ones. The importance of local meanings is also addressed. Indeed, as underscored by Prasad (1993, p. 1404): "symbolic interactionists recognize that people may symbolically sustain certain notions such as freedom, justice, and leadership at a broad societal level, but that these notions contain very different images and meanings for people within particular institutional contexts." Therefore, pervasive concepts such as socialization can carry very different connotation in different contexts and thus need to be understood as such.

The foundation for interaction among open source community members is a shared system of meaning, comprised of symbols that, over time, evolve and take on meaning. Throughout this process, negotiation and feedback are present as community members together create symbols and new communal meanings. The entire communication process in open source communities is based on perceptions, the use of text-based language and symbols, and the creation and development of shared meaning, therefore making this area a fertile ground for the application of symbolic interactionism. In the next section we build on the existing OSS literature and apply the symbolic interactionist perspective to define socialization and explicate the socialization process of OSS community members in an effort to move closer to a theory of socialization in this context.

A SYMBOLIC INTERACTIONIST APPROACH TO OSS SOCIALIZATION

The preceding section informed our view of symbolic interactionism. Therefore, in this section, we offer an explication of the socialization process (see Figure 1), along with a definition of the concept.
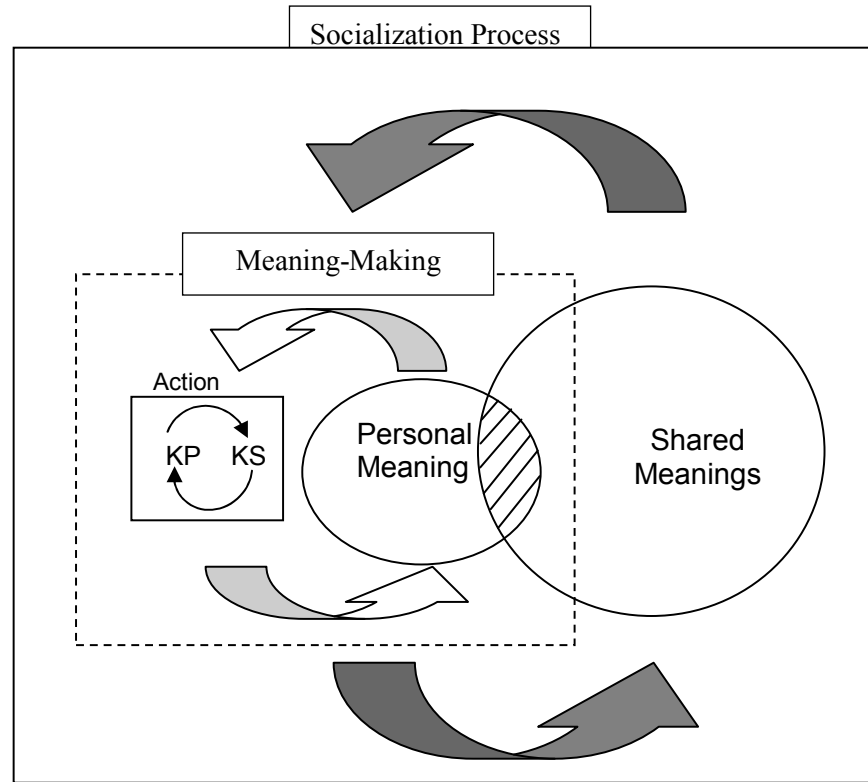
Legend: KP = knowledge providing, KS = knowledge sharing, hatched area = extent of socialization

Figure 1. A Symbolic Interactionist Perspective of OSS Socialization

Specifically, this section seeks to explicate the black box of socialization in open source software communities through the lens of the symbolic interactionist approach. Our main thesis is that the socialization process community members go through in open source communities can be best understood as the interdependent interplay between (1) meaning-making, which consists of action – comprised of both knowledge seeking and knowledge providing activities – and personal meaning, and (2) the shared meanings that evolve in the community (see Figure 1). It is important to note that given the interdependent nature of these concepts, the figure must not be interpreted as a defined linear sequence of events. Rather, the concepts are enmeshed with one another and take place simultaneously. Action and the development of personal meaning feed back into one another, with action shaping the personal meanings that individuals attribute to symbolic artifacts, and these meanings in turn influencing the future actions that are undertaken. This entangled process between action and personal meaning consists of the personal meaning-making process that individuals experience when interacting with the community. Furthermore, this personal meaning-making process does not take place in isolation, but instead is constantly influenced and shaped by one's interactions with other community members. The meanings that emerge within the community tend to be negotiated and mutually agreed upon – what we term here as "shared meanings." Therefore, we further contend that these shared meanings influence the personal meanings each community member is constantly deriving from the community. In short, the process of socialization is the integrated result of the interaction of individuals, consisting of action and personal meaning-making, leading to the emergence of shared meanings within the community. The above described interplay between action and personal meaning-making through a negotiation process with other individuals and the community allows members to form shared meanings from these shared experiences (Nonaka 1994). As such, we define socialization as the extent of overlap or correspondence between an individual member's personal meaning and the shared meanings of the community (see hatched area in Figure 1).

The remainder of this section covers each of the main components of the model. First, an explication of the two actions, knowledge seeking and knowledge providing, is offered. Each is addressed separately, followed by an explanation of how their interplay creates shared action and experience. Second, the interplay between action and personal meaning is explored, followed by the interaction between the shared meanings of the community and the individual meanings of its members. All of these interdependent components, taken together, form the socialization *process* of individuals into open source software communities, with socialization defined as the extent of overlap or correspondence between an individual member's personal meaning and the shared meanings of the community.

Knowledge Seeking

Knowledge seeking is the first side of the action coin. It involves the various activities undertaken by community members to acquire knowledge of the community. Since open source projects have publicly accessible project web sites, potential community members can easily access information on projects, including the identity of project administrators, the project's development stage, its level of activity, and the number of software downloads. They can also take note of project's mailing lists in order to familiarize themselves with the community and decide on whether or not to join it.

Research suggests that lurkers appear to follow a set of activities, commonly referred to as the "opening script" (von Hippel et al., 2003) in order to participate in the community after lurking for a certain length of time on the mailing list. These types of activities include seeking information about the group and its norms (Miller and Jablin 1991). A number of studies of open source projects indicate that opening activities have many dimensions, some of which include time spent lurking on the project's web site, frequency of emails submitted to the site, and quality of communications posted (Hertel et al. 2003; Mockus et al. 2002, Pitt et al. 2006). Many opening posts by lurkers consist of introducing themselves to the community, and/or submitting questions about the project in order to start learning about the community (von Hippel et al., 2003). In addition to seeking information on the symbolic meanings of the community's activities, technical knowledge can also be acquired by familiarizing oneself with the source code of the project's software product. This can not only provide newcomers with technical insights with regard to the developed software, but also provide them with indications of the expected contributions and level of expertise required of them.

All of this suggests that new members of OSS communities undertake knowledge seeking activities to make sense of the OSS community they are joining. However, as we have implied throughout, knowledge seeking does not take place in isolation, but rather hand-in-hand with knowledge providing activities.

Knowledge Providing

This section elaborates on the influence of knowledge providing activities on OSS community members' socialization process. As suggested by recent studies, it appears that during the lurking period, potential participants interact with posted messages to understand the community (von Krogh et al. 2003). The act of posting transforms implicit knowledge of the poster to explicit knowledge, also known as externalization (Nonaka 1994, von Krogh et al. 2003). Readers of these messages interact with the message, akin to interacting with the implicit knowledge of the poster. By doing so, they are interacting with the tacit knowledge – the norms, values, and beliefs – of the existing members of the community. In other words, message postings, or knowledge providing activities, by the existing members influence the participants' tacit knowledge.

The lurking period prepares newcomers to make a meaningful knowledge providing activity in the form of an opening post (Ducheneaut 2005). The replies from the core group to newcomers' opening posts offer newcomers cues about the expectations of the community. Thus, the knowledge providing activities of both newcomers and entrenched members influence the actions of each. In many cases, experienced members of the community suggest that new participants find some part of the software architecture to work on, but provide no encouragement, nor guidance (Von Krong et al 2003, Ducheneaut 2005). The kind of feedback existing developers and members give to a joiner helps the individual develop a sense of the project and the project team. This underscores the crucial role that knowledge providing activities play in developing shared meanings between members of OSS communities.

This transition of becoming a core developer comes with the privilege of creating and adding modules to the source code. The evolution of a joiner to a developer involves creating a persona that the community identifies with and recognizes (Ducheneaut 2005). Core members of the community are gatekeepers who make certain that developers show the merit of new ideas that they suggest, especially when proposing new modules, by garnering support from the community. As developers attempt to build support for their module using separate mailing lists, they become more active, engaging in, and initiating new discussions, and developing greater shared meanings with the community. This implies that developers are given the privilege of transitioning to core developers when they actively engage beyond coding; they have to build allies in the community.

To summarize, knowledge providing is one of the two categories of action. When members post to the community's shared knowledge space (the mailing list) they are "externalizing" their mental schemas (knowledge) and contributing to the process of creating shared meaning for the community. Similarly, when members undertake knowledge seeking activities they could be either "internalizing" or "externalizing" knowledge. During the internalization, members are actively deciphering the explicit knowledge of the community and this influences their own personal meanings about the community. The act of continuously engaging with the community through knowledge seeking and knowledge providing allows individual members to actively negotiate and develop a shared meaning system with the community.

Action

The symbolic interactionist perspective views action and meaning-making as two sides of the negotiated coin of socialization (Blumer 1969). Based on the symbolic interactionist perspective, we had mentioned earlier that the basic building blocks of socialization are action and personal meaning-making as depicted in Figure 1. We had further suggested grouping all actions by individual members in OSS communities into two categories: knowledge seeking and knowledge providing. Knowledge seeking actions consist of activities in which the individual member is soliciting knowledge from others to inform his personal understanding of the social situation, here the OSS community. Similarly, we grouped all activities by which members supply knowledge to those seeking it as knowledge providing activities. Knowledge seeking and knowledge providing do not take place in isolation.

Meaning-Making

The above discussion posits that the knowledge providing and knowledge seeking categories of action are mutually interdependent and do not exist in isolation. Moreover, these two categories of action are inextricably intertwined to the shaping of personal meaning. This inseparable coupling between action and personal meaning is itself implied in the term symbolic interaction (Gopal and Prasad 2000). Indeed, in the course of one's interaction with an open source community, personal meanings are assigned to objects and events based on actions that are undertaken by the individual. Personal meanings with regard to the community's objects and events stem from the action that community members undertake through their interactions with the community they are a part of. These actions, conceptualized here as both knowledge seeking and knowledge providing activities, shape the personal meaning community members develop with respect to the community. These interpretations of community events, situations, and objects also in turn influence the future actions of individuals – actions that depend on one's interpretation of the community at a given point in time. Symbolic interactionist scholars have always supported the notion that the relationship between meaning and action is dialectical and dynamic (Prasad, 1993). This interdependent and continuous interplay between action and personal meaning explicates the meaning-making process of individual community members as illustrated in Figure 1.

The Socialization Process

As previously indicated, we define socialization as the extent of correspondence between the personal meanings of individual members and the shared meanings of the community. This consonance between personal and shared meanings stems from the negotiated interactions members undertake with each other and/or the artifacts of the community.

While personal meaning-making is certainly an important component of the socialization process in open source software communities, it is important to recognize that the meanings and symbols that emerge from this process are not static objects or artifacts, but instead are constantly produced and reproduced through social interactions between community members. In fact, the process whereby symbolism influences and shapes everyday practices is a key notion of the symbolic interactionist position (Blumer, 1969). This notion of enactment is the process through which symbolism influences the actions of individuals in given contexts.

This symbolism is what we refer to as the shared meanings of the community. These shared meanings, which form and influence the norms, values, and behaviors in the community, are shaped by the prior and current interactions between the individual actions and personal meanings of community members. This happens through a negotiation process which consists of the interplay between knowledge sharing and knowledge providing activities. These shared meanings affect the personal meaning-making process that each community member experiences. Therefore, we argue that the socialization process consists of the interplay between individuals' meaning-making process, consisting of action and personal meaning, and the negotiated shared meanings within the community. The above described interplay between action and personal meaning takes place through a negotiation process with other individuals within the community, and allows members to form shared meanings. The extent of socialization of a community member is indicated by the extent of overlap between an individual member's personal meaning derived from the community and the shared meanings that are negotiated and established

between its members, which ties back to our initial definition. This is illustrated by the overlapping (hatched) area between personal meaning and shared meanings in Figure 1.

CONCLUSION

We started this paper by recognizing the lack of theoretical and empirical work on member socialization in virtual communities. Using the symbolic interactionist approach, we then attempted to explicate – or in other words, begin to move toward a theory of – socialization in virtual, normative, and knowledge-creating communities, with a specific focus on open source software communities. Although we focused on socialization in OSS communities, we believe that this first draft of the theory is also applicable to other types of virtual communities such as virtual brand communities (e.g., iPod user groups) since they are also virtual, normative, and knowledge-creating communities.

We defined socialization as the extent of correspondence, resulting from a shared and negotiated experience, between the personal meanings of individual members and the shared meanings of an OSS community, and we argued that the socialization process consists of the interplay between individuals' meaning-making process, consisting of action and personal meaning, and the negotiated shared meanings within the community. Although prior research has begun to identify the various stages of membership that individuals experience when taking part in OSS communities, the fact remains that there is a lack of theory when it comes to understanding socialization in this context. We believe that we have begun to address this gap in this paper.

REFERENCES

1. Ahuja, M. K. and J. E. Galvin (2003) Socialization in Virtual Groups, *Journal of Management*, 29 (2), 161-85.
2. Bell, B. S. and S. W. J. Kozlowski (2002) A typology of virtual teams: Implications for effective leadership, *Group & Organization Management*, 27 (1), 14-49.
3. Blumer, H. (1969) *Symbolic Interactionism: Perspective and Method*. Englewood Cliffs, New Jersey: Prentice-Hall.
4. Colton, C. W. (1987) Leisure, Recreation, Tourism: A Symbolic Interactionism View, *Annals of Tourism Research*, 14 (3), 345-57.
5. Driver, M. (2005) Positions 2005: Open-Source Solutions Will restructure the Software Industry, *Gartner Research Report*, I.D. Number G00126518, February 23, 2005, 1-11.
6. Ducheneaut, N. (2005) Socialization in an Open Source Software Community: A Socio-Technical Analysis, *Computer Supported Cooperative Work (CSCW)*, 14 (4), 323-68.
7. Galvin, J. and Ahuja, M. (2001) Am I Doing What's Expected? New Member Socialization in Virtual Groups, in Chidambaram, L., and Zigurs, I. (Eds.), *Our Virtual World: The Transformation of Work, Play and Life Via Technology*, Hershey, PA: Idea Group Publishing, 40-55.
8. Gibson, C. B. and S. G. Cohen (2003) *Virtual Teams That Work: Creating Conditions for Virtual Team Effectiveness*, Jossey-Bass.
9. Gopal, A. and P. Prasad (2000) Understanding GDSS in Symbolic Context: Shifting the Focus from Technology to Interaction, *MIS Quarterly*, 24 (3), 509-46.
10. Hertel, G., S. Niedner, and S. Herrmann (2003) Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel, *Research Policy*, 32 (7), 1159-77.
11. Lois, J. (1999) Socialization to heroism: Individualism and collectivism in a voluntary search and rescue group, *Social Psychology Quarterly*, 62 (2), 117-35.
12. Martins, L. L., L. L. Gilson, and M. T. Maynard (2004) Virtual Teams: What Do We Know and Where Do We Go From Here? *Journal of Management*, 30 (6), 805-35.
13. Maas, W. (2004) Inside an Open Source Software Community: Empirical Analysis on Individual and Group Level, *Proceedings of the 4th Workshop on Open Source Software Engineering*, Edinburgh Scotland, 64–70.
14. Maznevski, M. L. and K. M. Chudoba (2000) Bridging Space Over Time: Global Virtual Team Dynamics and Effectiveness, *Organization Science*, 11 (5), 473-92.
15. Mignerey, J. T., R. B. Rubin, and W. I. Gorden (1995) Organizational Entry: An Investigation of Newcomer Communication Behavior and Uncertainty, *Communication Research*, 22 (1), 54-54.

16. Miller, V. D. and F. M. Jablin (1991) Information Seeking during Organizational Entry: Influences, Tactics, and a Model of the Process, *The Academy of Management Review*, 16 (1), 92-120.

17. Mockus, A., Roy. T Fielding, and J. D Herbsleb (2002) Two Case Studies of Open Source Software Development: Apache and Mozilla, *ACM Transactions on Software Engineering and Methodology*, 11 (3), 309-46.

18. Nonaka, I. (1994) A Dynamic Theory of Organizational Knowledge Creation, *Organization Science*, 5 (1), 14-37.

19. Orr, J. (1990) Sharing Knowledge, Celebrating Identity: War Stories and Community Memory in a Service Culture, in *Collective Remembering: Memory in Society*, D.D Middleton and D. Edwards, Eds. Beverly Hills, CA: Sage Publications.

20. Pitt, Leyland, Richard Watson, and Pierre Berthon (2006) The Penguin's Window: Corporate Brands From an Open-Source Perspective*, Journal of the Academy of Marketing Science*, 34 (2), 115-27.

21. Powell, A., G. Piccoli, and B. Ives (2004) Virtual Teams: A Review of Current Literature and Directions for Future Research, *The DATA BASE for Advances in Information Systems*, 35 (1), 6-36.

22. Prasad, P. (1993) Symbolic Processes in the Implementation of Technological Change: A Symbolic Interactionist Study of Work Computerization, *The Academy of Management Journal*, 36 (6), 1400-29.

23. von Hippel, E. and G. von Krogh (2003) Open Source Software and the" Private-Collective" Innovation Model: Issues for Organization Science, *Organization Science*, 14 (2), 209-23.

24. von Krogh, G., S. Spaeth, and K. R. Lakhani (2003) Community, joining, and specialization in open source software innovation: a case study, *Research Policy*, 32 (7), 1217-41.