9-18-2006

Research Agenda for Studying Open Source II: View Through the Lens of Referent Discipline Theories

Fred Niederman
St. Louis University, niederfa@slu.edu

Ashley Davis
University of Georgia

Martina E. Greiner
University of Georgia

Donald Wynn
University of Georgia

Paul T. York
University of Georgia

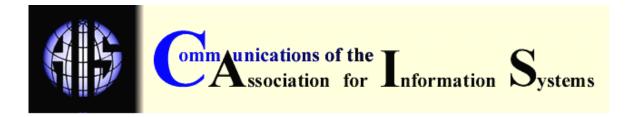Follow this and additional works at: https://aisel.aisnet.org/cais

RESEARCH AGENDA FOR STUDYING OPEN SOURCE II: VIEW THROUGH THE LENS OF REFERENT DISCIPLINE THEORIES

Fred Niederman
Saint Louis University
niederfa@slu.edu

Ashley Davis
Martina E. Greiner
Donald Wynn
Paul T. York
University of Georgia

ABSTRACT

In a companion paper [Niederman et al., 2006] we presented a multi-level research agenda for studying information systems using open source software. This paper examines open source in terms of MIS and referent discipline theories that are the base needed for rigorous study of the research agenda

Keywords: open-source software, adaptive structuration theory, agency theory, complexity theory, diffusion theory, game theory, social network theory, and transaction cost theory.

I. INTRODUCTION

Open source is an alternative to proprietary approaches to marketing software. Open source projects challenge aspects of organizational and societal thinking about work and software development as a result of their varied but distinctive philosophies about intellectual property. As a phenomenon that bridges the technical and the human, open source is a topic of interest to MIS scholars. Although this paper is aimed at MIS researchers, open source is a broad domain of study of interest to researchers in many academic disciplines. Existing research is published in areas as diverse as software engineering, sociology, economics, and public policy.

The goal of this paper is to suggest theoretical approaches drawn from management, behavioral sciences, and economics that apply to MIS research and that offer the opportunity to extend knowledge and principles to open source software development and use. Application of theoretical perspectives can:

- illuminate ways of thinking about open source software within MIS;

- illustrate areas where open source represents an example of general MIS principles and where it is a distinct study domain; and

- create the opportunity to test prevailing theory for its applicability to open source software development and use.

WHAT IS OPEN SOURCE[1]?

The central tenet of open source software is that the source code is available for anyone who wants to use or modify it. Beyond that broad definition, a continuum of "openness" exists. Variations in licensing agreements define categories with differing levels of restrictiveness on the use of "open source" software.

The classic scenario for open source software occurs when an individual wants others to share in a relatively large project (more than the individual wants to do alone) primarily because the individual wants to use the software created. The individual posts the project to a website and asks for contributions. If interest is sufficient, a core group of programmers and designers begins serious volunteer work to develop the software. A larger group reviews the output, adding significant patches, and a still larger group tests and finds weaknesses in the software that need repair [Mockus, et al., 2002]. For highly successful projects, such as GNU/Linux, Apache, and Mozilla, the stable software created is released to literally millions of users.

Particular social structures, including communities and a volunteer workforce, are generally viewed as part of open source software; however, the specific nature of the communities and work arrangements show more variation than the stereotypical image would suggest. Krishnamurthy [2002], for example, shows that in the majority of cases open source code was developed and continues to be managed by only a few or even a single developer.

More and more traditional proprietary software companies are releasing (fully or selectively) the source code for otherwise commercial products. Microsoft, for example, reportedly released source code for selected products to selected customers [Cukier, 2005]. However, the consensus among researchers seems to be to use the Open Source Initiative (OSI) [OSI, 2006] definition[2]. This definition effectively means that any software distributed under an OSI approved license is 'open source' and anything distributed under a non-OSI approved license is not open source. This definition would, for example, exclude Microsoft's shared source initiative from being considered a form of open source.

The term "free software" [Free Software Foundation, 2006] is frequently used in addition to "open source". The emphasis of the Free Software Foundation is on preserving a range of freedoms for the acquisition, use, distribution, and modification of software beyond simply allowing for direct access to source code.

In this paper, we use the term open source to include both philosophical positions.

RELATION TO RESEARCH AGENDA PAPER

This paper discusses MIS theories applicable in open source environments. It proposes that theories described here can be used as a basis for generating individual research projects and as a part of research streams. In a companion paper [Niederman, et al. 2006] which immediately precedes this paper, we discuss a five level research agenda for the study of open source. Readers are urged to read both papers to obtain a fuller understanding of the open source research proposed.

---

[1] This section is identical to the same section in the companion paper [Niederman et al. 2006].

[2] OSI defines open source on its website as: When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs. And this can happen at a speed that, if one is used to the slow pace of conventional software development, seems astonishing [OSI, 2006].

ORGANIZATION OF THIS PAPER

In Section II we present seven theories that we believe illuminate open source: adaptive structuration theory, agency theory, complexity theory, diffusion theory, game theory, social network theory, and transaction cost theory. For each theory we discuss how the theory is used in MIS and who it can be adapted to studying open source. We present the conclusions and the limitations of this paper in Section III.

Table 1. Seven Referent Discipline Theories Used in MIS Research.

	Theory	Basic Concept
1	Adaptive structuration theory	Actors create and are constrained by social structures that can be represented as rules or norms. This provides a lens for viewing the interaction of developers, users, and technology as it is put into practice.
2	Agency theory	When organizations employ agents to represent them, there is always a gap of some degree between the goals of the organization and agent. Minimizing these gaps represents an opportunity and cost to organizations.
3	Complexity theory	As systems grow more complex, they are likely to be explained through concepts such as decomposition, feedback looks, and non-linear relationships
4	Diffusion theory	New technologies spread in their adoption according to patterns that frequently resemble S-shaped curves; there are communication oriented factors that influences the rapidity and pattern of such diffusion
5	Game theory	In a transactional situation two or more "players" making moves will resolve into patterns suggesting best moves. When such situations are observed in practice, the best moves of the game can inform the actors in practice
6	Social network Theory	The positioning of an element within a network contributes to its value and to the kind of messages which pass through it between other network members
7	Transaction cost theory	Transactions themselves have a cost. The nature of the transaction will affect whether organizations will more likely want to conduct the transaction on the market or internally (through a hierarchy).

II. THEORY BASES

This section presents the seven theories used in MIS that we believe illuminate open source. The discussion is intended to illustrate the potential value of introducing referent discipline theory to issues within open source. These seven theories were selected based on their potential for examining open source issues and their existing base of application within the MIS literature. These theories were also selected to acknowledge the variation in the type of MIS research that they target. We aimed for an array of theories, while retaining a manageable number.   For each theory, we:

- briefly discuss the theory in general;

- address observations of research in which it is already applied;

- describe how it might be used in open source research; and

- suggest how it might be extended to additional areas within open source research.

www.manaraa.com

Examples are selected from relatively recent publications with the expectation that their lists of references will guide researchers to more comprehensive listings of MIS research based upon these particular theories. Examples are also selected to show the diversity in application of these theories in the literature. Because research questions posed in the examples are sometimes implicit, the questions were rephrased in the context of the present study.

Table 1 lists the basic concepts of the seven theories chosen.

Adaptive Structuration Theory

Adaptive structuration theory explains how social structures interact with technology in an adaptive fashion [DeSanctis and Poole, 1994]. An important consideration when this theory is applied to the introduction of a new technology into a social system is how the intentions of the designers are related to the actual manner of use. If the actual use is consistent with or faithful to the designer's intention, then the outcomes should match those intended by the designers. Both diversions from and adherence to the intentions of the designer in using the systems affect people's norms and assumptions, and thus influence future system use.

Table 2 compares and contrasts MIS and open systems use of adaptive structuration theory[3].

Table 2. Adaptive Structuration Theory in MIS and Open Source.

What Research Questions are Addressed by these Theories in Previous MIS Research?	Examples of Use in MIS Research	What Research Questions for OS Research are we Suggesting?	Example of OS Research
Does the fit between the task and GSS structures affect GSS use? Do the specific types of appropriation support affect GSS use? What are the types of enactment by which people create structures of social practice in the "ongoing use and change of technologies in the workplace"?	Dennis, Wixom, & Vandenberg, [2001] Orlikowski, W. J., [2000]	How do the types of appropriation support affect the use of open source software (Are these different from the relationship of support and use for proprietary software?) How do established patterns of enactment (e.g. team focused versus individualistic versus hierarchical) affect the likelihood that a particular end user company will select and adopt open source software? How does the use of open source rather than proprietary software change the personnel, design, and activities of an MIS department among end user firms? How do the intentions of the designers of open source software differ from the intentions of proprietary software designers? In turn, how does that difference affect the culture, activities, and outcomes among software users?	No observed instances

GSS = group support systems

---

[3] Note that open source is abbreviated OS in this table and in subsequent tables

Application of Adaptive Structuration Theory in MIS Research

Adaptive structuration theory has been applied in two major ways in MIS research. The first is a significant stream of Group Support Systems (GSS) research. DeSanctis and Poole (1994) proposed a detailed model for using a structurational framework for conceptualizing variables in explaining inputs, processes, and outputs about applying information technology to support group decision meetings. The structurational framework adds a dynamic feedback loop component to the description of issues involving the adjustment of groups to new technology and changes in technology that affect group processes. For the most part GSS research was not intended to test the structurational framework. In a larger sense, the structuration approach is not intended as a testable theory [Poole and DeSanctis, 2004], but rather to provide a conceptualization within which particular variables would be defined, measured, and tested. Dennis et al. [2001] exemplifies the evolving sophistication with which the structurational approach is applied in GSS research.

The second major approach to structuration focuses on organizational change. Orlikowski [2000] observed a range of mechanisms by which people enact particular use and change of technology in organizations. Broad categories would include inertia, change, and application. These categories are further distinguished by interest in using the technology. They result in differences in interpretive, technical, and institutional conditions and in process, technology, and structural consequences.

Adaptive Structuration Theory in Open Source Research

No examples of the use of adaptive structuration theory were found in the existing open source literature.

Extending Adaptive Structuration Theory to Additional Areas within Open Source Research

A major theme in the original formulation of adaptive structuration theory by DeSanctis and Poole (1994) pertains to the tension between intentions of designers and system users. They called large differences in these intentions low faithfulness and small differences high degree of faithfulness. Such a concept can be used to test whether there is a difference in intention of open source artifact designers and proprietary open source artifact designers. If such a difference is found, what is its nature, how strong is it and does it make a difference in the nature or quality of the artifact? Does it make a difference in the experience of the artifact from the user's point of view?

On the one hand, it can be argued that at least among individuals playing both designer and user roles (e.g. core community members) there should be little difference between designer intentions and actual appropriation patterns. Members of an open source community have no reason to appropriate the system unfaithfully. On the other hand, it is an open question whether members of the community would differ in their intentions from users in the broader community. For example, volunteer developers may find little incentive to develop useful documentation and may, therefore, either try to develop "self-documenting" software or simply release software with no documentation and "let the user beware".

From the perspective of organizational change, adaptive structuration theory raises questions about the effects (if any) that the open source philosophy may have on its organizational adoption. At the operating system level open source versus proprietary artifacts may result in no noticeable difference on organizations. However, strong distinctions could exist at the enterprise level where, even if only to stay abreast of version changes, there may be explicit or implicit requirements to participate in an open source community. In terms of Orlikowski's [2000] work, do organizations with stronger or weaker commitment to specific types of enactment react differently to open source in contrast to proprietary software, and, if so, are the differences significant and what is their nature?

www.manaraa.com

Another application of adaptive structuration theory would be to consider the adoption of both technical (e.g. development tools and version control tools) and social structures within communities. In other words, how does the introduction of new social structure such as mixed professional and volunteer labor affect the norms and values of the community? Madanmohan and Krishnamurthy [2005] discuss the importance for commercial firms of working within open system community norms. They discuss this problem in terms of "legitimating" projects, or providing motivation for the community value of the project and of the processes and roles to be played.   Given that in the typical community developers use only computer-mediated relationships and may never meet face-to-face, structuration theory may provide useful insight into the evolution of communities and points of change. Although the interaction of the work group has been studied somewhat in open source, the impact in an adaptive structuration context of this topic has not been studied at all.

AGENCY THEORY

Agency theory deals with the contractual relationship between a principal and agent whose goals and attitudes toward risk differ [Eisenhardt, 1989]. Since agency theory assumes that both parties are goal maximizers, the agent does not always act in the principal's best interest [Jensen and Meckling, 1976]. The focus of agency theory is thus to determine the most efficient contract governing the principal-agent relationship to make sure that the agent fulfills the principal's interest [Jensen and Meckling, 1976]. The basic human assumptions of agency theory are bounded rationality, self-interest toward fulfilling goals, and different level of risk aversion [Eisenhardt, 1989]. Contracts between principal and agent involve delegation of decision making responsibilities to the agent so that the agent can make decisions which may not be completely known to the principal. If the principal does not have complete information about the behavior of the agent, the agent may capitalize by behaving opportunistically to maximize his goals. The two main problems related to bounded rationality are moral hazard (i.e., agent is shirking) and adverse selection (i.e., the principal misinterprets the abilities of the agent). Risk aversion, (i.e., principal and agent may differ about how much risk to take) can lead to decisions that are not in the principal's best interest. The principal will therefore make efforts to deploy mechanisms to ensure that the agent behaves in the principal's best interest. Such efforts include monitoring of the agent, providing incentives to the agents, and requesting guarantees from the agent if a contract goal is not fulfilled. The two basic control options available to the principal are behavior-based contracts (e.g., salaries, efficient if principal is able to monitor the agent; hierarchical governance) and outcome-based contracts (i.e., provide incentives through e.g., stock options; market governance) [Eisenhardt, 1989].

Application of Agency Theory in MIS Research

Two very different studies are described to provide a flavor of the different types of application of agency theory in MIS research.

1. In the knowledge management MIS literature, agency costs were acknowledged in the organizational knowledge creation process [Chen and Edgington, 2005] in employee contracts. This particular knowledge management study was performed using simulations.

2. In a different type of study involving subsidiaries, the argument is made that there is an influence of organizational characteristics on agency costs in subsidiary situations [Mirchandani and Lederer, 2004]. In this case, the agent is the subsidiary and the principal is the parent firm. IS planning is studied in terms of agency theory [Mirchandani and Lederer, 2004] because IS planning is an example of a decision making that a parent (principal) may or may not delegate to the subsidiary (agent). The methodology for the subsidiary study was a field survey.

As can be seen from these two examples, agency theory is applied in MIS research to quite different problems.

Agency Theory in Open Source Research

No examples of the use of agency theory were found in the existing open source literature.

Extending Agency Theory to Open Source

Agency theory applied to open source is important because the motivation of the agent is in question. Agency theory is built around differing goals of the principal and the agent. In an open source community, the principal might be defined as the developers, or in some cases, an organization sponsoring open source development might be considered the principal. The agents would be the managers trying to control the development process or the developers. It is clear that there would be a conflict of goals; but it is not clear what would be the goals of the principal and agent. Research should explore these relationships.

Application of agency theory to open source requires first addressing three basic problems:

- Do the human and organizational assumptions of agency theory (i.e., self-interest, risk aversion, bounded rationality, goal conflict, efficiency, and information asymmetry) apply to open source relationships? One might argue that the voluntary and altruistic nature of open source communities does not match the self-interest assumption. However, open source literature shows that interests of contributors differ widely from reputation building to career concerns to purely monetary motivations [Hars and Ou, 2002; Lakhani and Wolf, 2003]. Thus, self-interest and goal conflicts are likely to occur.

- What are the principal-agent relationships in open source ecosystems? Relationships encompass those among developers, between adopting user (principal) and developer (agent), and between adopting firm (principal) and project/vendor/community (agent). Open source communities are embedded in a large ecosystem with many different entities such as professional open source companies, vendors, and consultants. These actors need to be taken into consideration.

- Do open source relationships mirror a contractual relationship between a principal and agent? If no contractual or pseudo-contractual relationship can be assumed, then agency theory cannot be applied since the principal cannot influence the agent.

Table 3 describes agency theory in MIS and open source.

After a solid case for these three basic questions is established, application of agency theory to open source can begin. One application would be to use agency theory as a lens for determining characteristics of the principal-agent relationship for open source (e.g., transaction relationships, collaboration relationships, co-development relationships). Defining these characteristics enables exploring agency problems that might arise.

Another application of agency theory includes explicating the monitoring, incentive, and bonding mechanisms in the principal-agent relationship. These mechanisms may differ from traditional mechanisms since many participants in an open source project are not paid. Therefore, maximizing salary is not an agent motivation in many cases. Likewise, maximizing profits may not be the principal's motivation. However, even if maximizing profits is the principal's motivation, the mechanisms through which profits are maximized are different for open source organizations. For example, an open source organization may sell consulting services that customize software for clients. In this case, the principal is not incentivized to build comprehensive software. However, the developers (agents) may see comprehensive software as the best goal for their efforts. In this case, what incentives would mitigate this agency problem?

Table 3. Agency Theory in MIS and Open Source

What Research Questions are Addressed by these Theories in Previous MIS Research?	Examples of Use in MIS Research	What Research Questions for OS Research are we Suggesting?	Example of OS Research
How do we strategically assess knowledge creation over time giving consideration to complex decision criteria in order to improve organizational value? How does agency theory explain the relationship between key organizational variables and the autonomy of IS planning?	Chen and Edgington, 2005 Mirchandani & Lederer, 2004	Do the human and organizational assumptions of agency theory (i.e., self-interest, risk aversion, bounded rationality, goal conflict, efficiency, and information asymmetry) apply to open source relationships? What are the possible and interesting principal-agent relationships in open source ecosystems? Do OS relationships mirror a contractual *relationship* between a principal and agent? What are the characteristics of the principal-agent relationship (e.g., transaction relationships, collaboration relationships, co-development relationships) and what problems may arise by these distinct characteristics? What are monitoring, incentive, and bonding mechanisms to mitigate the principal agency problems? How do they differ from traditional mechanisms? What business models develop through problems that arise through the principal-agent relationship?	No observed instances

Along the same lines, different business models are used in the open source industry. HP and IBM, for example, participate in open source communities such as Linux and Apache in order to increase their influence on project directions. Professional open source companies offer their customers written contracts and guarantees to help to reduce the uncertainty around open source. Do these business models serve to mitigate the principal-agent relationship?

COMPLEXITY THEORY

Complexity theory applies to dynamic systems, capable of changing over time, and the predictability of their behavior [Rosenhead, 2005]. Some systems are stable in that given specified inputs, predictable outputs are created. However, other systems are dynamic in that outputs cannot be predicted reliably from inputs due to internal positive and negative feedback loops, strong influence of initial states, and the interaction of potentially uncountable numbers of different inputs with multiple values. Applied to management, this theory would challenge ordinary views of rational behavior and systematic cycles of planning and action taking, suggesting a more experimental learning approach particularly in highly turbulent industries or

times. In developing a significant grounded theory approach to management during times of constant change, Brown and Eisenhardt [1997] presented three themes for distinguishing high from low levels of management success:

1. examining multiple new initiatives, assigning clear responsibilities, and extensive communication;

2. using low-cost probes into the future such as experimental products, futurists, and strategic partnerships, and

3. linking future and past actions through carefully timed transitions [Brown and Eisenhardt, 1997].

These findings broke ranks substantially with earlier management thinking. The researchers proposed that their observations were more consistent with complexity theory than with other possible theoretical explanations. Table 4 shows the role of complexity theory in IS and open source.

Table 4. Complexity Theory in MIS and Open Source

What Research Questions are Addressed by these Theories in Previous MIS Research?	Examples of Use in MIS Research	What Research Questions for OS Research are we Suggesting?	Example of OS Research
How can the difficulties of maintenance and cost that follow from complexity of large information systems be minimized? How can organizations find the balance minimizing complexity at the system level and the component levels?	Sarkar & Ramaswamy, 2000 Schneberger, S., McLean, E., 2003	Can complexity theory help explain variance in open source and proprietary software artifacts? Can complexity theory help explain the characteristics of open source communities and heir relationship to specific artifact characteristics?	No observed instances

Application of Complexity Theory in MIS Research

Complexity theory has been used in MIS to show the value of partitioning large systems to create and evaluate them more effectively (Sarkar and Ramaswamy, 2000). However a more nuanced view holds that complexity potentially exists at the system level and at the level of each component (Schneberger and McLean, 2003). As personal computers assumed increasingly large roles in organizations, the complexity of computing declined relative to centralized mainframe processing, but as PCs became networked, the system complexity increased. A reasonable goal would be to seek the equilibrium point where the complexity of the whole system, and its components, are allocated to the maximum benefit.

Complexity Theory in Open Source Research

No examples of the use of complexity theory were found in the existing open source literature.

www.manaraa.com

Extending Complexity Theory to Open Source

One application of complexity theory to open source research involves considering the complexity of artifacts themselves. To the extent that increased complexity makes maintenance more complex and, therefore, more costly, less complex software is preferable, assuming all else (functionality, throughput, and flexibility) is equal. The interaction of community development methods may increase artifact complexity. Rather than emphasizing methodical planning and step-by-step movement from one phase to another in the development process, (represented at the inflexible extreme by the traditional waterfall model) complexity theory would suggest structures typically found in open source projects.

Another application of complexity theory is the study of the organizational structure of open source communities themselves. Consistent with the Brown and Eisenhardt's [1999] description that successful managers within organizations face high levels of turbulence, the typical open source project relies little on formal planning, but greatly on responsibilities and high levels of communication. The actions of teams, projects, and communities may function in accord with complexity theory by displaying both positive and negative feedback loops (e.g. as work becomes more concentrated among fewer people, remaining workers will tend to leave, with the result that work becomes even more concentrated among even fewer people). Identification of specific positive and negative feedback loops that occur frequently in open source development communities can provide important information for those leading or participating in such communities.

DIFFUSION THEORY

As defined by Rogers [1995], innovation is communicated through particular channels over time among the members of a social system. The newness of the idea being communicated is the defining characteristic around which this body of research was formed. As the rate and extent of new technologies and processes continues to increase, the classic diffusion model proved to be a useful descriptive and diagnostic tool for researchers. The basic components of the classic model include:

- the innovation itself,
- the characteristics and roles of adopters,
- the process through which they attempt to adopt the innovation,
- the social context, and
- the communication channels through which the innovation is passed.

Research across many disciplines has been classified into eight basic types of innovation research [Rogers, 1995]. The most common type employs the innovativeness of members of a social system as a dependent variable and their characteristics as independent variables.

Diffusion Theory in MIS Research

Diffusion research within the MIS domain historically focused on the impact of the specific attributes of a given innovation on the rate at which that innovation is adopted for use by organizations. The five basic attributes of an innovation are:

- complexity,
- compatibility,
- relative advantage,
- observability, and

Research Agenda for Studying Open Source II: View Through the Lens of Referent Discipline Theories by F. Niederman, A. Davis, M.E. Greiner, D. Wynn, and P.T. York

- trialability [Rogers, 1995].

Over time, many researchers proposed additional attributes in the study of technology diffusion [Downs and Mohr, 1976; Tornatzky and Klein, 1982], including critical mass, cost, and social approval. MIS research typically revolved around relative advantage, compatibility, trialability, ease of use, image, visibility, and result demonstrability.

The classical diffusion model focuses on identifying variables that serve as precursors to successful adoptions[4]   As such, the theory appears to apply more readily to discrete, straightforward technologies than to ones that involve linked adoption decisions and complex organizational contexts [Fichman, 2000]. A wider range of factors have been investigated across a number of studies in IT, including the organization-innovation fit, firm, and IS unit characteristics, and the actions of institutions seeking to propagate the innovation [Fichman, 2000]. Table 5 shows the role of diffusion theory in MIS and open source.

Diffusion Theory in Open Source Research

Within open source research, we found no studies derived from the classical diffusion literature. Instead, the focus of innovation research is largely based on the communication network and social structures within open source communities. Unlike most conventional software, the communication of innovations and ideas in open source software is often bottom-up from users to developers as opposed to the more typical top-down approach [von Hippel, 2001; Franke and Shah, 2003]. The community that evolves around an open source product is often the primary source of innovation, especially where the users of the product become involved as co-developers [Raymond, 1998]. Other research papers discuss the critical mass in users/developers required for open source software diffusion [Bonaccorsi and Rossi, 2003; von Hippel and von Krogh, 2003], organizational adoption factors [Wang and Wang, 2001; Dedrick and West, 2003], and the roles of individuals in the innovation process [Ye, Kishida et al., 2002; von Krogh, Spaeth et al., 2003].

Table 5. Diffusion Theory in MIS and Open Source

What Research Questions are Addressed by these Theories in Previous MIS Research?	Examples of Use in MIS Research	What Research Questions for OS Research are we Suggesting?	Example of OS Research
At what rate do new technologies diffuse among user? What are the characteristics of users that facilitate adoption? What are the characteristics of technologies that facilitate adoption? What are the technical and organizational factors that influence innovation?	Karahanna, Straub, and Chervany, 1999 Fichman, 2004 Hardgrave, Davis, and Riemenschneider, 2003	Are there differences in the rate or influences on diffusion of OS versus proprietary SW? Does OS introduce new factors influencing diffusion of SW?	No observed instances

---

[4] In some research it is assumed that adoption is, by its nature, a measure of success.

A number of papers address a different, more technical set of requirements for the adoption of open source technologies. For example, Wang and Wang [2001] list technical requirements (availability of technical support, future upgradability, open-standard compatibility, customizability, extensibility, and reliability) and management requirements (budgetary, development team expertise, licensing, project scope, and long-term maintainability). Their paper then assesses a wide range of operating systems, application environment, development library, and application open source products in terms of these criteria. The issues listed in papers of this nature almost never overlap with constructs developed by the diffusion literature (e.g. Rogers) such as trialability.

Extending Diffusion Theory to Additional Areas within Open Source Research

Using Rogers' [1995] typology of diffusion research, a framework for future research in open source can be developed. Organizations are not uniform in their adoption of open source artifacts. Whereas some organizations have been extremely proactive in employing open source artifacts, others have not developed policies and procedures for doing so in the future. Further studies of the specific aspects of these organizations that predispose them toward leading or lagging the adoption curve would enrich general understanding of influences on adoption of complex socio-technical systems. Also, research into the impact of features and complexity of specific open source software applications on the ultimate consequences of implementing the software in various contexts would benefit researchers and practitioners attempting to understand the necessary preconditions for successful adoption of open source software. Other research in open source (e.g. development models, social networks, and motivation of participants) will inform research on the diffusion and adoption across social contexts. It is particularly appealing to consider Fichman's (2004) call for alternative approaches to diffusion theory (in contrast to the classical approach) coupled with the domain of open source development. Some of the alternatives suggested, including social contagion and management fashion, would be of particular appeal as approaches to investigate the shift from proprietary to open source software use.

GAME THEORY

Game theory refers to a loose collection of single-person (vs. environment), two-person, and multi-player/group strategic games that are used primarily to model decision behavior. Game theory deals with strategic games, which are distinct from games of pure chance (e.g., gambling) and those of pure skill (e.g., 100-meter dash), however the games often involve some degree of both chance and skill in addition to pure strategy [Dixit & Skeath, 2004]. Game theory also operates under a number of assumptions, including the existence of a measurable payoff for the winner, players operating under the norms of rationality, a common knowledge of the rules of the game, and the existence of an equilibrium towards which the game will naturally progress given the prior assumptions (Dixit & Skeath, 2004). The games are broadly categorized as allowing for sequential or simultaneous moves, or actions. In other words, players either know the other player's action prior to acting (e.g., chess) or both players must anticipate the other player's actions prior to deciding on an immediate strategy (e.g., American football). Games become increasingly complex with the addition of multiple players, the introduction of dynamic or evolving rules and payoffs across multiple rounds of play, and the existence of incomplete or asymmetric information.

Table 6 shows the role of game theory in MIS and open source.

Application of Game Theory in MIS Research

Game theory was used in MIS research to model strategies for information technology outsourcing (Elitzur & Wensley, 1997). Their approach synthesizes the nature of actions taken by each side in an outsourcing relationship by modeling the essential characteristics of the transactions as a kind of game. Such an approach can be used not only to characterize the

essence of the relationship between say a vendor and customer, but also to assess specific details of their arrangements such as fee structures, risk sharing, relationship building, and renegotiation. A different application in MIS research was modeling strategies for entry into the information technology and telecommunication marketplace (Nault and Vandenbosch, 2000). This approach considers the incentives and risks for new companies to enter into particular product markets, and for companies to invest in innovative technologies that may replace their own successful offerings. Such an approach is used to consider the relationships between competitors absorbing market forces.

Table 6. Game Theory in MIS and Open Source

What Research Questions are Addressed by these Theories in Previous MIS Research?	Examples of Use in MIS Research	What Research Questions for OS Research are we Suggesting?	Example of OS Research
What are the essential characteristics of relationships between actors in the IT marketplace?	Elitzur, R., & Wensley, A., 1997 Nault, B. R., & Vandenbosch, M. B. , 2000 Orlikowski, W. J., 2002	What are the essential characteristics of individual designers working in the open source domain (and in contrast to those in the proprietary domain)? What are the essential characteristics of organization and vendor/service provider in the open source domain?	Johnson, 2002 von Hippel and von Krogh, 2003 O'Mahony, 2003 Stenborg, 2004 Bitzer and Schroder, 2005

Game Theory in Open Source Research

The type of game most commonly studied in connection with open-source software development is the *collective action* game. Collective action [Olsen, 1971] examines the dilemma of producing pure public goods, or goods that are produced for the nonexclusive (i.e., available to all without exception) and non-rival (i.e., one person's use doesn't diminish its value to others) use of the public at large. Ideally, all who benefit from public good use would also be involved in creating and maintaining them; however, in reality such goods are most often produced by a few with the remainder of users, known as *free riders*, benefiting without cost.

Traditional open source software development fits the collective action model quite well. Developers produce a software product that is subsequently made available for public use that is both nonexclusive and non-rival. Numerous researchers used this lens to analyze open source development. For example:

- Johnson [2002] creates an economic model to describe open source software development that is based entirely on the collective action model.

- Von Hippel and von Krogh [2003] posit that open source development is not a true collective action dilemma, but rather that it should be seen as a "private-collective" from which developers gain certain private returns from their contribution to the projects.

- O'Mahony [2003] argues that while open source developers allow free use of their products, they maintain a number of private rights to the software by leveraging one of several licensing options.

Stenborg [2004] and Bitzer and Schroder [2005] both incorporate a second game, a *War of Attrition*, to help explain how the dilemma of collective action is overcome in the domain of open source.

Nault and Vandenbosch [2000] could potentially be applied to relationships between different types of open source creation communities, firms selling or buying open source services, and choices of individual designers and community leaders about incentives and responses for starting and continuing work on open source projects.

Extending Game Theory to Open Source

While many papers already apply game theory to open source, other aspects of the open source phenomenon have yet to be analyzed using this lens. First is the effect that organizations are having on the "publicness" of open source software. As profit-generating firms seek to generate revenue from open source software, payoff structures change significantly. Developers of many modern open source projects (e.g., Apple's Darwin and Sun's Open Office) must forfeit their rights to their contributions to the commercial owners of these projects. Firms such as JBoss, Inc. do not require forfeiture of rights, but, by nature of their market position, are de facto primary sources of paid support services for their product suite. In these cases, while the software remains open, much of the profit potential (payoff structure) for these projects becomes both exclusive and rival. Are developers' motivations towards development of such impure public goods different than those seen in the development of pure public goods? General collective action research provides for the study of impure public goods, and should be further incorporated into the current research base on open source. Further, to date the use of game theory in open source focused heavily on developer motivations. However, numerous other skills are involved in this marketplace. In addition to the corporations participating directly in the open source community, traditional software providers are affected directly by open source competition. Their actions and reactions to this form of competition, as well as an eventual market equilibrium, should also be able to be modeled using one or more dynamic and evolutionary games. Open-source is a significant challenge to traditional economics in the software industry, and game theory should help provide useful insight into the dynamics of this industry over the next few years.

Finally, the Nault and Vandenbosch [2000] approach described earlier in this subsection could potentially be applied to relationships between different types of open source creation communities, firms selling or buying open source services, and choices of individual designers and community leaders about incentives and responses for starting and continuing work on open source projects.

SOCIAL NETWORK THEORIES

The social capital construct is defined in a number of ways that are consistent with one another. One of the original definitions of social capital describes the network of strong, interpersonal ties that provide a basis for trust, cooperation, and collective action [Jacobs, 1965]. Social capital is a resource derived from the interactions of members of an organization. It consists of the close, personal ties that members in an organization possess. It is both a resource that individuals within an organization possess and a valuable resource to the organization. Social capital refers to networks, norms, trust, and mutual understanding among members of an organization that enables these members to act together more effectively to pursue shared objectives. Many of the conceptualizations of social capital are applicable to the open source movement, and some have been used in open source research.

For example, one conceptualization of social capital includes three dimensions: structural, cognitive, and relational dimensions [Nahapiet and Ghoshal, 1998]. Another definition argues that social capital is comprised of the three elements of opportunity, motivation, and ability [Adler, 2001]. In the open source literature, some exploration uses these theories as foundations [Wang, 2005]. However, these theories must be pushed to their limits to understand the open source community.

Nahapiet and Ghoshal [1988] conceptualized the structural dimension as composed of network ties, network configuration, and appropriable organization. Network ties include the interaction or

networking part of social capital. The network ties or the relationships that the actor possesses together with the location of these relationships in the social structure of the organization are represented by structural capital. A rich literature describes social networks in organizations [Burt, 1997; Gabbay and Zuckerman, 1998; Burt, 2000; Inkpen and Tsang, 2005]. Network ties, network configuration, and the impact of networks are most commonly studied in the MIS field.

Application of Social Network Theory in MIS Research

Social capital was studied in virtual communities by Wasko and Fara (2005). Their assessment of structural, cognitive, and relational capital in the context of knowledge contribution to a national legal professional association provides a starting point for addressing social capital issues in MIS research.

Social Network Theory in Open Source Research

Existing studies explored networks in open source communities:

- Madey et al. [2002] examine collaborative networks in terms of clusters of networks and perhaps a power-law relationship.

- Ghosh [2003] looks at source code authorship and dependencies between projects. Both Lopez et al. [2004] and Gonzalez-Barahona et al. [2004] build on those articles and study network characterization beyond the distance between the actors, into the strength of the relationship [López et al,. 2004] and how the nodes interact to form groups [González-Barahona,et al., 2004].

Further, conceptualizations of the importance of networks and their impacts on organizations, in terms of network holes are discussed in Burt [1997 and 2000]. Boundary spanners may give insight into the open source community.

An important first step toward understanding a more complete view of network social capital and social structure in open source communities was taken by Crowston and Howison [2005] who investigate open source communities during the bug-fixing process. Their findings suggest that open source projects are not consistent in their social structure of communications. Although not theory driven, Krishnamurthy [2002] observed similar findings of diversity and largely individual efforts in creating code. It would be of interest to determine the characteristics of the network dimension of social capital that are generalizable across open source projects.

Research in open source also examines network governance [Jones et al., 1997] in works that explore project success [Sagers, 2004]. These insights are important in the open source community because of the gift culture [Bergquist and Ljungberg, 2001] discussed previously.

However, a common limitation to most of these open source studies is that they focus on the structural dimension of social capital but do not incorporate other dimensions. Broader studies of social capital and its effects in open source communities are needed. The cognitive and relational aspects of social capital, when fully integrated into the network understanding of structural capital of open source communities should better predict success of open source projects.

Table 7 shows the role of social network theory in MIS and open source.

Extending Social Network Theory to Open Source

Social capital is a multi-level construct that can be analyzed at several different levels. The micro-macro conceptualization recognizes that social capital is an individual level attribute generating outcomes at the organizational and group as well as individual levels, and thus functions as a multi-level concept [Fukuyama, 1995; Oh, et al., 2004]. Social capital benefits the individual who possess it and also, at the group level, benefits the group or community [Kostova and Roth, 2003]. In terms of understanding the open source community, individuals are

motivated by both individual level and by project-based outcomes. Social capital theory should help us understand these motivations. Beyond just the structural component of social capital [Nahapiet and Ghoshal, 1988], the open source domain can be expanded by exploring the relational and cognitive components of social capital.

Table 7. Social Network Theory in MIS and Open Source

What research questions are addressed by these theories in previous MIS research?	Examples of use in MIS research	What research questions for OS research are we suggesting?	Example of OS research
Why do people voluntarily contribute to knowledge and help others through electronic networks? How do individual motivations and social capital foster knowledge contribution? What are the implications of using IT to interfirm relations?	Wasko and Faraj, 2005 Schultze and Orlikowski, 2004	How is Social Capital in Open Source communities different from social capital in other virtual communities or from social capital within traditional organizations? How do venture capital firms' networks affect the open source organizations in which they invest?	Crowston and Howiston, 2005 Madey, Freeh, and Tynan, 2002 Ghosh (2003) Lopez et al., 2004 Gonzalez-Barahona et al. 2003

TRANSACTION COST THEORY

Transaction cost economics (TCE) is used in organization theory, marketing, and information systems, among others, to understand, select, and design the governance structures regulating economic transactions between partners. TCE focuses on the most efficient governance structure for a specific transaction. TCE is applied in organizational research to answer questions about the boundaries and existence of firms. TCE focuses on transactions as the basic unit of economic activity and stresses that costs occur when undertaking a transaction [Williamson, 1981]. Transaction costs include contractual ex ante costs (such as those related to searching, information gathering, bargaining, and negotiation) and ex post costs (such as those related to monitoring and contract enforcement) [Coase, 1960; Rindfleisch and Heide, 1997].

The TCE framework builds on the two basic human assumptions of bounded rationality and opportunism that influence transaction costs. Since not all information is available and people may behave opportunistically, costs occur because contracts cannot be completely ex ante determined, and safeguarding mechanisms against opportunism need to be implemented [Williamson, 1981]. Since it is assumed that efficiency is the basic criterion for designing transactions, an organization will economize on the sum of the production expenses (i.e., the costs for organizing a transaction within a firm such as administrative and coordination costs) and transaction costs by choosing the governance structure that is able to minimize those costs [Williamson, 1981]. Depending on transaction characteristics (i.e., asset specificity, uncertainty, and frequencies), different governance structures can be expected to lead to higher or lower transaction and production costs. The goal is to align the governance structure to the attributes of a transaction [Williamson, 1981]. Originally, only two distinct governance structures, markets and hierarchies, were included in the TCE analysis. Since then, the framework was extended to include other mixed governance structures such as franchising [Coase, 1937; Williamson, 1981].

Table 8 shows the role of transaction cost economics in MIS and open source.

Application of Transaction Cost Theory in MIS Research

In information systems, TCE is used as a theory base to explain and predict appropriate governance structures for outsourcing decisions [Riordan and Williamson, 1985; Aubert, Rivard et al., 1996; Ngwenyama and Bryson, 1999; Wang and Wang, 2001; Aubert, Rivard et al., 2004; Carmel and Nicholson, 2005]. TCE is also used to examine the relationships among collaborators and the use of technology in managing supply chain interactions [Subramani, 2004].

Table 8. Transaction Cost Economics in MIS and Open Source.

What Research Questions are Addressed by these Theories in Previous MIS Research?	Examples of Use in MIS Research	What Research Questions for OS Research are we Suggesting?	Example of OS Research
What sort of governance structures are used in managing relationships between outsourcing and service providing companies? How can investments by suppliers in supply chain relationships be understood?	Carmel & Nicholson, 2005 Subramani, 2004	How does the organization of software development work fit into transaction cost models? How are the transactions costs for users of open source software different from those of users of proprietary software or users of both?	Kauffman, & Mohtadi, 2004

Transaction Cost Theory in Open Source

The open source literature argues that virtual communities producing public goods (such as open source communities) are becoming a viable and competing form of organizational governance alongside hierarchies and markets [Benkler, 2002; Demil and Lecocq, 2003; Glaeser, 2003; Watson et al., 2005]. Table 9 shows dimensions along which an open source community can be distinguished from markets and hierarchies.

Table 9. of Governance Structure Comparison

	Hierarchies	Markets	Communities
Contract law regime:	Employment contract	Classical contract	Open license
Definition of task is:	Centralized	Decentralized	Decentralized
Primary adjustment of actions by:	Formal rules	Price	Common subject matter of work (i.e. product)
Membership determined by:	Formal rules	Exchange offer	Perception of being a member
Nature of incentives:	Career advancement, status concerns	Competition	Reputational concerns, signaling
Intensity of Incentives:	Low	High	Low
Control:	High	Low	Low

Sources: [Demil and Lecocq, 2003; Glaeser, 2003; Watson et al., 2005]

A key characteristic of communities is that transaction exchanges are not coordinated either by formal rules (hierarchies) or price (markets), but are coordinated in a decentralized manner by each developer acting autonomously according to his interest and common subject matter of work [Glaeser, 2003].

Eric Raymond, a founder of the open source movement, compares communities to a bazaar where software development appears to be a chaotic process like a "babbling bazaar of differing agendas and approaches" that is distinct from hierarchies, which he compares to "cathedrals, carefully crafted by individual wizards" [Raymond, 1998].

Extending Transaction Cost Theory to Open Source

Research on open source through a TCE lens is still in its infancy, especially empirical research. Some of the research opportunities in this area include:

- Under what conditions are open source communities a superior mode to coordinate economic transactions compared to alternative governance structures? Glaeser [2003] for example argues that communities are the most efficient governance mode under conditions of extreme uncertainty. Greiner et al. [2005] apply the TCE framework to assess make-or-buy decisions of software among the alternative governance structures (communities, markets, and hierarchies) depending on asset specificity.

- What are the main characteristics of a community making it a potentially superior form of governance structure? Demil et al. [2003], for example, proposes that communities potentially reduce transaction costs because of (among other things) reduced information gathering and negotiation costs.

- What mechanisms, such as quality insurance mechanisms, can an open source community implement to ensure that the potential advantages over markets and hierarchies last?

- How can TCE be used to explain the development of different business models in open source such as professional open source?

III. CONCLUSION

THE TWO PAPERS

The relatively small number of papers in our literature on open source demonstrates that the MIS implications of open source software are insufficiently studied. This paper and its companion [Niederman et al., 2006] provide a research agenda to jump start the work needed.

The first paper [Niederman et al., 2006], subtitled "A Multi-Level Framework", presents a multi-level research model that describes five discrete levels of analysis: (1) the artifact; (2) the individual; (3) the team, project, and community; (4) the organization; and (5) society. Specific issues within each of these five levels can be studied individually. As the research evidence accumulates it will be possible to address issues at several levels of analysis simultaneously. By viewing the field this way, individual studies can be compared, and their findings collected to broaden the overall understanding even if their areas of focus overlap only partially.

This second paper, subtitled "View Through the Lens of Referent Theories" presents seven intuitively appealing theories already familiar to IS researchers that we show can be applied to open source. These theories, from reference disciplines, discussed in alphabetical order, are (1) adaptive structuration theory, (2) agency theory, (3) complexity theory, (4) diffusion theory, (5) game theory, (6) social network theory, and (7) transaction cost theory. We discuss each theory, its previous use in MIS studies, and present examples of the way it can be applied to study open

source issues. We, therefore, believe that this paper will be of use to colleagues who seek to study open source.

NEW THEORY BASES

Note that the use of referent discipline theory does not in any way preclude the development or discovery of new theory that pertains only to open source phenomena or that may generalize from open source to other realms. Although we did not find new open source theory in the existing literature, new theory may well be generated. Such new theory would inevitably also suggest new ways of viewing technical and socio-technical systems in general.

ANTICIPATED IMPLEMENTATION DIFFICULTIES

We recognize that developing a fully realized body of open source research presents significant difficulties. These include:

- The richness of the open source environment may be difficult to capture. However, some issues may be amenable to experimentation (e.g., interface ease of use, preferences among license types), case study (e.g., for a specific development or the decision process for accepting/rejecting a piece of software) or action research.

- The range of development settings and circumstances for open source make findings difficult to generalize.

- With open source continual evolving, widely used techniques such as interviewing and observation used in qualitative studies may be difficult to apply.

- The on-line presence of developers potentially drifting in and out of projects may be difficult to capture with research-oriented precision.

- Case study and qualitative approaches always present difficulties in negotiating with site hosts, gathering and analyzing data, and hoping that discernable patterns will be observed.

We anticipate that future research will profit from a broad mix of research methods.

INITIAL RESEARCH RECOMMENDATION

As organizations increasingly adopt open source, we recommend that research initially focus on organizations as users of open source artifacts. Such research would center on the issues such organizations face entering into the open source world:

- integrating open source into their portfolio;

- deciding on levels of open source community participation;

- assessing the economic, organizational, and technical impacts of open source on operations and strategic business practices.

LIMITATIONS

This paper discusses seven theories and their application to studying open source. These theories are among the most popular used in MIS studies. However, they are not the only theories that can be used in open source research. For descriptions of other MIS theories, go to http://www.istheory.yorku.ca/

The same limitations that were discussed in the companion paper apply here. As stated in Niederman et al. [2006], these limitations are:

www.manaraa.com

"The method used for developing this paper is based on the discussions and thinking primarily among the authors and colleagues. In the end we focused on the presentation of a multi-level view of the open source domain. Although a wide range and large number of open source related papers were identified and reviewed, there can be no guarantee that coverage across the range of studies is comprehensive. We focused our attention on the content of findings in the various studies considered rather than on details of their methodology".

REFERENCES

Editor's Note: The following reference list contains hyperlinks to World Wide Web pages. Readers who have the ability to access the Web directly from their word processor or are reading the paper on the Web, can gain direct access to these linked references. Readers are warned, however, that

> 1. these links existed as of the date of publication but are not guaranteed to be working thereafter.
>
> 2. the contents of Web pages may change over time. Where version information is provided in the References, different versions may not contain the information or the conclusions referenced.
>
> 3. the author(s) of the Web pages, not AIS, is (are) responsible for the accuracy of their content.
>
> 4. the author(s) of this article, not AIS, is (are) responsible for the accuracy of the URL and version information.

Adler, P. S. and S.-W. Kwon (2002) "Social capital: prospects for a new concept", *Academy of Management Review* (27)1, pp. 17-40.

Aubert, B. A., S. Rivard, et al. (1996) "A transaction cost approach to outsourcing behavior: Some empirical evidence", *Information and Management* (30), pp. 51-64.

Aubert, B. A., S. Rivard, et al. (2004) "A transaction cost model of IT outsourcing", *Information and Management* (41), pp. 921-932.

Benkler, Y. (2002) "Coase's penguin, or, Linux and the nature of the firm", *The Yale Law Journal* (112)3, pp. 369-446.

Bergquist, M. and J. Ljungberg (2001) "The Power of Gifts: Organising Social Relationships in Open Source Communities", *Information Systems Journal* (11)4, pp. 305-320.

Bitzer, J. and P. J. H. Schroder (2005) "Bug-fixing and code-writing: The private provision of open source software", *Information Economics & Policy* (17)3, pp. 389-406.

Bonaccorsi, A. and C. Rossi (2003) "Why Open Source software can succeed", *Research Policy* 32(7), pp. 243-1258.

Brown, S. L. and K. M. Eisenhardt (1997) "The Art of Continuous Change: Linking Complexity Theory and Time-paced Evolution in Relentlessly Shifting Organizations." *Administrative Science Quarterly* (42), pp. 1-34.

Burt, R. S. (1997) "A Note on Social Capital and Network Content" *Social Networks* (19)4, pp. 355-373.

Burt, R. S. (2000) "The network structure of social capital", *Research in Organizational Behavior (22),* pp. 345-423.

Carmel, E. and Nicholson, B. (2005). "Small Firms and Offshore Software Outsourcing: High Transaction Costs and Their Mitigation", *Journal of Global Information Managemen*t, 13(3): pp. 33.

Chen, A. N. K., and Edgington, T. M. (2005). Assessing value in organizational knowledge creation: Considerations for Knowledge Workers1. *MIS Quarterly*, 29(2), pp. 279-309.

Coase, R. H. (1937) "Nature of the Firm", *Economica* (4)16, pp. 386-405.

Coase, R. H. (1960) "The Problem of Social Cost", *Journal of Law and Economics (*3), pp. 1-44.

Crowston, K. and J. Howison (2005) "The Social Structure of Free and Open Source Software Development", http://www.firstmonday.org/issues/issue10_2/crowston/index.html.

Cukier, K. (2005) "A Market for Ideas: A survey of patents and technology", *The Economist*, pp. 1-16.

Dedrick, J. and J. West (2003) "Adoption of Open Source Platforms: An Exploratory Study", HBS - MIT Sloan Free/Open Source Software Conference: New Models of Software Development, Boston, MA.

Demil, B. and X. Lecocq (2003) "Neither market nor hierarchy or network: The emerging bazaar governance." available at http://opensource.mit.edu/online_papers.php, (current August 21, 2005).

Dennis, A. R., Wixom, B. H., & Vandenberg, R. J. (2001). Understanding fit and appropriation effects in group support systems via meta-analysis. *MIS Quarterly*, 25(2), pp. 167.

DeSanctis, G. and M. Poole (1994) "Capturing the Complexity in Advanced Technology Use: Adaptive Structuration Theory", *Organization Science* (5)5, pp. 121-147.

Dixit, A. and S. Skeath (2004) *Games of Strategy.* New York, W.W. Norton & Company Ltd.

Downs, G. W. and L. B. Mohr (1976) "Conceptual Issues in the Study of Innovation", *Administrative Science Quarterly* (21) December, pp. 700-714.

Eisenhardt, K. M. (1989) "Agency Theory: An Assessment and Review", *The Academy of Management Review* (14)1. pp. 57-75.

Elitzur, R., & Wensley, A. (1997). Game theory as a tool for understanding information services outsourcing. Journal of Information Technology (Routledge, Ltd.), 12(1), pp. 45-60.

Fichman, R. G. (2004). Going beyond the dominant paradigm for information technology innovation research: Emerging concepts and methods. *Journal of the Association for Information Systems*, 5(8), pp. 314-355.

Fichman, R. G. (2000) "The Diffusion and Assimilation of Information Technology Innovations" in R. W. Zmud. *Framing the Domains of IT Management: Projecting the Future Through the Past,* Cincinnati, OH: Pinnaflex Educational Resources.

Franke, N. and S. Shah (2003) "How Communities Support Innovative Activities: An Exploration of Assistance and Sharing Among End-Users", *Research Policy* (32)1, pp. 157-178.

Fukuyama, F. (1995) *Trust: the social virtues and the creation of prosperity.* New York, NY, Free Press.

Gabbay, S. M. and E. W. Zuckerman (1998) "Social Capital and Opportunity in Corporate R&D: The Contingent Effect of Contact Density on Mobility Expectations", *Social Science Research* (27), pp. 189-217.

Ghosh, R. A. (2003) "Clustering and dependencies in free/open source software development: Methodology and tools", *First Monday* 8(4).

Ghosh and Prakash (2000), "The Orbiten Free Software Survey", http://orbiten.org/ofss/01.html.

Glaeser, J. (2003) "A Highly Efficient Waste of Effort: Open Source Software Development as a Specific System of Collective Production", TASA 2003 conference, University of New England.

González-Barahona, J. M., L. López, et al. (2004) "Community structure of modules in the Apache project", GSyC Working Paper, Universidad Rey Juan Carlos (Mostoles).

Greiner, M. and D. Goodhue (2005) "Make-or-Buy in the Age of Open Source: A Transaction Cost Analysis" International Conference of Information Systems, Las Vegas (ICIS 2005), pp. 689-700.

Hardgrave, B. C., Davis, F. D., & Riemenschneider, C. K. (2003). Investigating determinants of software developers' intentions to follow methodologies. *Journal of Management Information System*s, 20(1), pp. 123-152.

Hars, A. and S. Ou (2002) "Working for Free? Motivations for Participating in Open-Source Projects", *International Journal of Electronic Commerce* (6)3, pp. 25-39.

Hertel, G., Niedener, 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, pp.1159-117.

Inkpen, A. C. and E. W. K. Tsang (2005) "Social Capital, Networks, and Knowledge Transfer", *Academy of Management Review* (30)1, pp. 146-165.

Jacobs, J. (1965) *The Death and Life of Great American Cities.* London: Penguin Books.

Jensen, M. C. and W. H. Meckling (1976) "Theory of the Firm: Managerial Behavior, Agency Costs, and Ownership Structure", *Journal of Financial Economics* (3), pp. 305-360.

Johnson, J. P. (2002) "Open Source: Private Provision of a Public Good." *Journal of Economics and Management Strategy* (11)4, pp. 637-662.

Jones, C., W. S. Hersterly, et al. (1997) "A general theory of network governance: exchange conditions and social mechanisms", *Academy of Management Review* (22)4, pp. 911-945.

Karahanna, E., Straub, D. W., & Chervany, N. L. (1999). Information technology adoption across time: A cross-sectional comparison of pre-adoption and post-adoption beliefs. *MIS Quarterly*, 23(2), pp. 183-213.

Karels, M.J. (2003). "Commercializing Open Source Software," *Queue*, July/August, pp. 46-55.

Kauffman, R.J., & Mohtadi, H. (2004), "Proprietary and Open Systems Adoption in E-Procurement: A Risk-Augmented Transaction Cost Perspective." *Journal of Management Information Systems*, 21(1), pp. 137-166.

Koch, S. and J. M. Gonzalez- Barahona (2005) "Open Source software engineering - The state of research", *First Monday*, http://www.firstmonday.org/issues/special10_10/koch/index.html (10)10.

Kostova, T. and K. Roth (2003) "Social capital in multinational corporations and a micro-macro model of its formation", *Academy of Management Review* 28(2), pp. 297-317.

Krishnamurthy, S. (2002) "Cave or Community? An Empirical Examination of 100 Mature Open Source Projects", *First Monday*, http://www.firstmonday.org/issues/issue7_6/krishnamurthy/index.html (7)6.

Lakhani, K. R. and R. G. Wolf (2003) "Why Hackers Do What They Do: Understanding Motivation Effort in Free/Open Source Software Projects", *MIT Sloan School of Management Working Paper 4425-03,* http://opensource.mit.edu/online_papers.php.

López, L., J. M. González-Barahona, et al. (2004) "Applying Social Network Analysis to the Information in CVS Repositories", Proceedings of the First International Workshop on Mining Software Repositories, http://msr.uwaterloo.ca/papers/Liu.pdf.

Madanmohan, T.R. and De, R. (2004) "Open Source Reuse in Commercial Firms", *IEEE Software*, (21) 6 pp. 62-69.

Madanmohan, T. R. and S. Krishnamurthy (2005) "Can the Cathedral Co-exist with the Bazaar? An Analysis of Open Source Software in Commercial Firms", *First Monday*, http://firstmonday.org/issues/special10_10/madanmohan/index.html Special Issue # 2.

Madey, G., V., Freeh, V, and Tynan, R. (2002) The Open Source Software Development Phenomenon: An Analysis based on Social Network Theory", Proceedings of the Americas Conference on Information Systems, Dallas, Texas, pp. 1806-1813.

Mirchandani, D. A., & Lederer, A. L. (2004) IS planning autonomy in US subsidiaries of multinational firms. *Information & Management*, 41(8), pp. 1021-1036.

Mockus, A., R. T. Fielding, et al. (2002) "Two Case Studies of Open Source Software Development: Apache and Mozilla", *ACM Transactions on Software Engineering and Methodology* (11) 3, pp. 309-346.

Nahapiet, J. and S. Ghoshal (1998) "Social Capital, Intellectual Capital, and the Organizational Advantage", *Academy of Management Review* (23) 3, pp. 242-265.

Nault, B. R., & Vandenbosch, M. B. (2000). Research report: Disruptive technologies - explaining entry in next generation information technology markets. *Information Systems Research*, 11(3), pp. 304.

Ngwenyama, O. K. and N. Bryson (1999) "Making the Information Systems Outsourcing Decision: A Transaction Cost Approach to Analyzing Outsourcing Decision Problems." *European Journal of Operational Research* (115), pp. 351-367.

Niederman, F. A. Davis, M.E., Greiner, D. Wynn, & P.T. York. (2006) "A Research Agenda for Studying Open Source I: A Multilevel Framework" *Communications of the Association for Information Systems* (18) 7, pp. 129-149.

Oh, H., M.-H. Chung, et al. (2004) "Group Social Capital and Group Effectiveness: The Role of Informal Socializing Ties", *Academy of Management Journal*, (47) pp. 860-875.

Olsen, M. (1971) *Logic of Collective Action: Public Goods and the Theory of Groups,* Cambridge: Revised Edition Harvard University Press.

O'Mahony, S. (2003) "Guarding the Commons: How Community Managed Software Projects Protect Their Work", *Research Policy* (32)7, pp. 1179-1198.

Orlikowski, W. J. (2000). Using technology and constituting structures: A practice lens for studying technology in organizations. *Organization Science*, 11(4), pp. 404.

Orlikowski, W. J. (2002). Knowing in practice: Enacting a collective capability in distributed organizing. *Organization Science*, 13(3), pp. 249.

Opens Source Initiative (2006) Open Source Initiative (http://www.opensource.org/)

Poole, M. S., & DeSanctis, G. (2004). Structuration theory in information systems research: Methods and controversies. In M. E. Whitman & A. B. Woszczynski (Eds.) *Handbook of information systems research* (pp. 206-249). Hershey, PA: Idea Group.

Raymond, E. S. (1998) "The Cathedral and the Bazaar", http://www.firstmonday.dk/issues/issue3_3/raymond/index.html; (accessed August 21, 2005).

Rindfleisch, A. and J. B. Heide (1997) "Transaction Cost Analysis: Past, Present, and Future Applications", *Journal of Marketing* (61)4, pp. 30-54.

Riordan, M. H. and O. E. Williamson (1985) "Asset Specificity and Economic Organization", *International Journal of Industrial Organization* (3), pp. 365-378.

Rogers, E. M. (1995) *Diffusion of Innovation*, New York, New York, U.S.A.: The Free Press.

Rosenhead, J. (2005) "Complexity Theory and Management Practice",. Working Paper LSEOR 98.25; Department of Operational Research, London School of Economics; London 1998. ISBN 0 7530 1253 7. http://human-nature.com/science-as-culture/rosenhead.html.

Sagers, G. (2004) "The Influence of Network Governance Factors on Success in Open Source Software Development Projects", Twenty-Fifth International Conference on Information Systems, Washington DC.

Sarkar, Sumit, & Ramaswamy, Mysore. (2000). Knowledge base decomposition to facilitate verification. *Information Systems Research*, 11(3), pp. 260.

Schneberger, S., McLean, E. (2003) The Complexity Cross: Implications for Practice. *Communications of the ACM* 4(6:9), pp. 216-225.

Schultze, U. and Orlikowski, W.J. (2004) A Practice Perspective on Technology-Mediated Network Relations:The Use of Internet-Based Self-Serve Technologies. *Information Systems Research*, 15 (1), pp. 87-106.

Stenborg, M. (2004) "Waiting for F/OSS: Coordinating the Production of Free/Open Source Software", opensource.mit.edu.

Stewart, K. J. & Gosain, S. (forthcoming) "The Impact of ideology on Effectiveness in Open Source Software Development Teams", *MIS Quarterly*.

Subramani, M. (2004). "How Do Suppliers Benefit from Information Technology Use in Supply Chain Relationships?", *MIS Quarterly*, 28(1). pp. 45-73.

Tornatzky, L. G. and K. J. Klein (1982) "Innovation Characteristics and Innovation Adoption Implementation: A Meta-Analysis of Findings", *IEEE Transactions on Engineering Management (*29)1, pp. 28-44.

Von Hippel, E. (2001) "Innovation by User Communities: Learning from Open Source Software", *Sloan Management Review*, pp. 42(4) 82-86.

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, pp. 209.

von Krogh, G., S. Spaeth, et al. (2003) "Community, joining, and specialization in open source software innovation: a case study", *Research Policy* (32)7, pp. 1217-1241.

Wang, H. and C. Wang (2001). "Open Source Software Adoption: A Status Report." *IEEE Software* 18(2): pp. 90-95.

Wang, J. (2005). The Role of Social Capital in Open Source Software Communities. *Proceedings of the Eleventh Americas Conference on Information Systems,* Omaha, NE.

Wasko, M.M. and Faraj, S. (2005) Why Should I Share? Examining Social Capital And Knowledge Contribution In Electronic Networks Of Practice.   *MIS Quarterly* 29(1), pp. 35-57.

Watson, R.T., M-C. Boudreau, et al (2005). "Governance and global communities." *Journal of International Management*, 11(2): pp. 125-142.

Watson, R.T., Wynn, D.E., Boudreau, M.C. (2005). "JBOSS: The Evolution of Professional Open Source Software," *MIS Quarterly Executive* (4:3), September 2005, pp. 329-341.

Williamson, O.E. (1981). "The Economics of Organization: The Transaction Cost Approach." *The American Journal of Sociology* 87(3): pp. 548-577.

Williamson, O.E. (1981). "The Modern Corporation: Origins, Evolution, Attributes." *Journal of Economic Literature* 19(December): pp. 1537-1568.

Ye, Y., K. Kishida, et al. (2002). Creating and maintaining sustainable open source software communities. International Symposium on Future Software Technology, Wuhan, China.

ABOUT THE AUTHORS

Fred Niederman is the Shaughnessy Endowed Professor of MIS at Saint Louis University. His publications appear in such journals as *MIS Quarterly*, *Decision Sciences*, *DATABASE, Journal of Global Information Management*, *Communications of the ACM,* and *Communications of AIS*. He is an associate editor of the *Journal of Global Information Management.* . He is former chair of the ACM special interest group on Computer Personnel Research and has been active in supporting its annual conferences. His research interests include virtual teams, groups, and meetings; global information management; and IT personnel and workforce.

Ashley R. Davis is a PhD student in the MIS department of the Terry College of Business at the University of Georgia. Her research interests include open source business development, enterprise resource planning systems, information technology infrastructure decisions, and service oriented architecture.

Martina Greiner is a PhD student in the MIS department of the Terry College of Business, the University of Georgia. Her current research areas are open source software and open content communities. She has published in the *Journal of International Management* and in several IS conference proceedings.

Donald Wynn, Jr. is a PhD student in the MIS department of the Terry College of Business, the University of Georgia. His current research interests include open source software communities,

ecosystems, business models, technology innovation, and information systems security. His publications appear in the *Journal of International Management* and the *Journal of the Academy of Marketing Science (JAMS)*.

Paul T. York is a Ph.D. Student in the Management Information Systems department at the University of Georgia. His current research interests include open source business models, information technology outsourcing, and the interaction between organizational policies and the global political economy.

.

Research Agenda for Studying Open Source II: View Through the Lens of Referent Discipline Theories by F. Niederman, A. Davis, M.E. Greiner, D. Wynn, and P.T. York

www.manaraa.com

www.manaraa.com