Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2006 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2006

An Absorptive Capacity Perspective of Open Source Software Development Projects

Sherae Daniel
University of Maryland

Follow this and additional works at: http://aisel.aisnet.org/amcis2006

Recommended Citation
Daniel, Sherae, "An Absorptive Capacity Perspective of Open Source Software Development Projects" (2006). *AMCIS 2006 Proceedings*. 24.
http://aisel.aisnet.org/amcis2006/24

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2006 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

An Absorptive Capacity Perspective of Open Source Software Development Projects

Sherae Daniel
University of Maryland
sdaniel@rhsmith.umd.edu

ABSTRACT

An organization's absorptive capacity is important for facilitating innovation that can lead to organizational performance. I examine the effects of absorptive capacity on open source software development group performance. Specifically, in addition to the commonly noted importance of OSS developers, the role of the Internet-based user community is introduced as fundamental in developing absorptive capacity for open source software development projects. I suggest that the Internet-based user community and the development group enhance the open source software project's absorptive capacity by strengthening two salient capabilities; knowledge acquisition and knowledge transfer. My dissertation research develops a model that relates knowledge acquisition and knowledge transfer to open source software development group performance and empirically test the model.

Keywords

Absorptive Capacity, Open Source Software, Software Development.

INTRODUCTION

The ubiquity and global reach of the Internet is fundamentally altering the way software is developed and marketed. The Internet reduces the restriction that developers must be geographically co-located and therefore increases the number and diversity of potential developers on a project. The Internet facilitates the collaboration of developers from different organizations, cultures and experience backgrounds on a software development project. Based on the diversity of backgrounds the developers are likely to offer different kinds of knowledge to the development process. Similarly, the Internet broadens the ways in which users can participate in the development process. Previously, users were often co-located with developers and had a shared organizational affiliation. The users were also recruited by the organization and as a set the number of users who could participate in the development process was limited. Internet-based user communities (IBUCs) represent a significant departure from traditional user participation. The Internet enables participation from a dynamic, diverse set of users, not recruited or selected by the organization. Also, the size of the IBUC may increase exponentially during the development process. Examples of such communities include those with a focus on proprietary software (http://forums.microsoft.com/) and open source software (OSS) ("SourceForge.net").

OSS projects are on the forefront of leveraging the Internet to develop software. There are many OSS projects that take advantage of diverse developer and user communities (see "SourceForge.net"). As an initial step in understanding the antecedents of success for Internet-based collaborations, this research focuses on OSS projects. Consistent with prior OSS research (Stewart, Ammeter and Maruping, 2005), this study identifies OSS by the definition provided by the Open Source Initiative.

Research to understand the antecedents of success in OSS projects has focused primarily on the developers. Research has focused on developer motivation (Hars and Ou, 2002), development team processes (Stewart and Gosain, 2006) and developer license choice (Stewart et al., 2005). Because software development is a knowledge-intensive activity, this research complements prior OSS research by taking a knowledge management perspective. In particular I draw on the absorptive capacity framework to understand the antecedents of OSS development group performance.

The main contributions of this study lie in using a fresh perspective to understand OSS development group performance. Specifically, the contribution is in the developing and testing of a theoretical model explaining how the characteristics of IBUCs, software developers, and their interaction facilitate knowledge acquisition, transfer and application to enhance

performance. Finally, this research will extend the current OSS development research by highlighting and explaining the key role that users play in the OSS development process.

The remainder of this paper proceeds as follows. The next section summarizes prior work on absorptive capacity and highlights salient dimensions of absorptive capacity for OSS projects. This is followed by development of a model and the presentation of hypotheses. Following the research model, an overview of the proposed methodology is presented and this paper finishes with concluding remarks.

ABSORPTIVE CAPACITY

Drawing on the seminal work by Cohen and Levinthal (1990), I define absorptive capacity as an organization's ability to recognize the value of new, external information, assimilate it, and apply it. Based on Cohen and Levinthal (1990), Zahra and George (2002) defined two key dimensions of absorptive capacity; potential and realized absorptive capacity. Potential absorptive capacity makes an organization receptive to acquiring and assimilating external knowledge (Lane and Lubatkin, 1998), while realized capacity is a function of the transformation and exploitation capabilities of an organization.

Potential absorptive capacity is critical for organizations that operate in highly dynamic industries. For such organizations problems and solutions are continually changing and so the organizations must be able to continually acquire knowledge to create solutions to relevant problems. Because technology changes rapidly, the problems or the technical needs of users also change rapidly (Iansiti and MacCormack, 1997). As the technology and problems change the solutions change and organizations need to acquire knowledge about new solutions. Therefore, organizations seeking to develop software cannot rely on potentially outdated knowledge, suggesting a need for continuous knowledge acquisition and application. This motivates our focus on the aspect of potential absorptive capacity for OSS projects.

Researchers have suggested that the ability to acquire knowledge is enhanced when an organization has preexisting related knowledge (Cohen and Levinthal, 1990; Fichman and Kemerer, 1999). Preexisting related knowledge is the extent of abstract knowledge, know how, and skills within the organization in areas related to the focal innovation (Fichman and Kemerer, 1999). Preexisting related knowledge is important because it provides the foundation necessary for understanding new knowledge and how the new knowledge is relevant to new innovations. For example, in an OSS development project, knowledge about HTML within a project enables that project to acquire knowledge about XML and understand why XML may be relevant to the OSS project.

I argue that that the availability of pre-existing related knowledge is related to the participant's diversity of experience with SourceForge, a platform often used to develop OSS, and the number of relationships that participants have with other OSS projects. Experience with the SourceForge platform signals the experience with OSS and their experience with OSS development. Relationships with other projects may also provide knowledge that may lead to innovations. There are two primary types of knowledge from SourceForge and other projects that are important to organizations that seek to innovate; knowledge of a problem that needs a solution and knowledge of the solution. I argue that these two distinct types of knowledge are found primarily within two different types of participant in OSS projects.

OSS PROJECT PARTICIPANTS: THE DEVELOPERS AND THE IBUC MEMBERS

There are several ways to categorize the participants in OSS projects. Ye and Kishida (2003) describe eight types of participants from their study of the GIMP project, an OSS development project that processes images in Linux. These include the project leader, the core members, the active developers, the peripheral developers, bug fixers, bug reporters, readers and the passive users. Fine-grained categorizations of participants, developed from exploration of some of the largest OSS projects, while they provide more detail about these projects, may not apply for many of the smaller or less mature OSS projects. These categorizations also may not be important from an absorptive capacity perspective.

I therefore consider a simpler categorization scheme and partition participants in an OSS project into two roles; the developer and the IBUC member. The developers are those who actively develop source code and include Ye and Kishida's project leader, core member and active developer roles. Then, drawing from the growing literature around OSS participant roles (e.g. Hertel, Niedner and Herrmann, 2003), the IBUC for a given OSS development project is the set of participants who demonstrate an interest in the software developed by the OSS development project by participating in some computer-mediated communication related to the project, but do not regularly develop source code. The IBUC includes Ye and Kishida's peripheral developers, bug fixers, bug reporters, readers and the passive users.

The distinction between the OSS development group and the IBUC is important because these two participant types have different characteristics that enable them to facilitate knowledge acquisition in different ways. The developers, because they develop source code, have a specific set of knowledge based on their understanding of the technical details of the project. Developers are therefore able to acquire knowledge based on their understanding of the technical details about the project, or

because the technical details of the project suggest that the new knowledge is relevant. While the IBUC members are not expected to have knowledge of the technical details of the project, they may have other knowledge that facilitates OSS development group performance.

The IBUC may be important for knowledge acquisition because they bring a distinct background to the project. Because the IBUC members may not share a programming background their background is different from developers and therefore may enable IBUC members to acquire knowledge that is different from the developers. For example, IBUC members are more likely than developers to acquire knowledge about problems that users who are not technically proficient have because IBUC members may not be technically proficient. As a second example, while developers are often experienced in software development (Zhao and Elbaum, 2003), IBUC members may be experienced in other domains. IBUC member experiences in domains other than software development allows IBUC members to be better prepared, compared to developers, to acquire knowledge from business contexts other than software development. Knowledge from other business contexts can lead to feature requests or bug reports that potentially are easily implemented by developers but that the developers may not have thought of because they do not have the business context background necessary to understand the need for the feature. Consistent with Nambisan, Agarwal and Tanniru (1999), I argue that innovation for OSS projects occurs at the "confluence of business expertise and technical mastery."

Because IBUC members may have greater access to know-what and developers specialize in know-how, a strong knowledge transfer capability is likely to improve the effect of the IBUC knowledge acquisition on development group performance. When one subunit within an organization has better access to knowledge, as I suggest is the case with the IBUC in the OSS project, the ability of subunits to transfer knowledge is an important predictor of organizational performance (Argote and Beckman, 1990; Hansen, 2002). Knowledge transfer is defined as the process through which one unit is affected by the experience of another (Argote and Ingram, 2000). For these reasons I propose that a strong knowledge transfer capability is a critical dimension of absorptive capacity for OSS projects. In summary, I argue that the salient dimensions of absorptive capacity for OSS projects are IBUC and development group knowledge acquisition and project knowledge transfer. Hypotheses relating OSS knowledge acquisition and knowledge transfer to development group performance will be developed and tested. The research model that suggests these relationships is below.

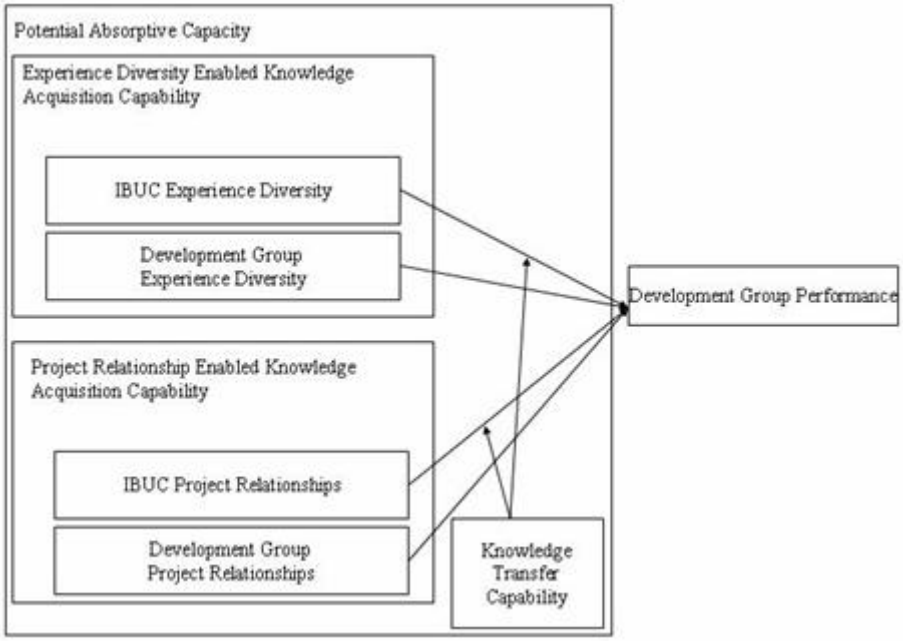

Figure 1. Research Model.

METHODOLOGY

The development and effects of absorptive capacity for OSS projects will be explored using data from a sample of OSS projects hosted by SourceForge. SourceForge is one of the largest repositories of OSS projects. Projects for the sample will be chosen based on three criteria; the project's registration date with SourceForge, the project's activity percentile and the project's use of the Gnu General Public License. By choosing projects that have similar registration dates, I limit the project variation that may be due to the time that the project was registered on SourceForge. Because SourceForge may be used as a place to distribute software and not to develop software, screening for projects with a minimum activity percentile will limit

the sample to projects that were actually under development during the time frame considered. Finally, the GPL is one of the most popular OSS licenses, therefore this restriction ensures that the projects are OSS projects according the OSI initiative. By selecting projects that use the same license, variation in projects that that prior literature suggests may be related to the license used is limited (Stewart et al., 2005).

CONCLUSION

Despite the emergence of IBUCs in OSS development and other Internet-based organizations, research to understand how IBUCs contribute to these organizations' performance is limited. This research begins to bridge the gap between theory and practice for Internet-based organizations. Results are expected to provide empirical evidence of how an IBUC can facilitate an organization's innovative abilities. Further, this research contributes to the growing literature that seeks to understand the factors that lead to success for OSS projects.

ACKNOWLEDGMENTS

I would like to thank my advisors, Ritu Agarwal and Katherine Stewart, for their contribution to my development as a researcher. It is my pleasure and privilege to be their student.

REFERENCES

1. Argote, L. and Ingram, P. "Knowledge Transfer in Organizations: A Basis for Competitive Advantage in Firms," *Organizational Behavior and Human Decision Processes* (82:1), 2000, pp. 150-169.Ajzen, I. (1988) Attitudes, personality, and behavior, The Dorsey Press, Chicago.
2. Argote, L., Beckman,S., and Epple, D. "The persistence and transfer of learning in industrial settings." *Management Science* (36:2), 1990, pp. 140-154.
3. Cohen, W. and Levinthal, D. "Absorptive Capacity: A New Perspective on Learning and Innovation." *Administrative Science Quarterly* (35), 1990, pp.128-152.
4. Fichman, R. and C. Kemerer. "The Illusory Diffusion of Innovation: An Examination of Assimilation Gaps," *Information Systems Research* (10:3), 1999, pp. 255-275.
5. Hansen, M. T. "Knowledge networks: Explaining effective knowledge sharing in multiunit companies," *Organization Science* (13:3), 2002, pp. 232-248.
6. Hars, A., and Ou, S. "Working for Free? Motivations for Participating in Open-Source Projects," *International Journal of Electronic Commerce* (6:3), 2002, pp. 25-39.
7. Hertel, G., Niedner,S., and Herrmann, S. "Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel," *Research Policy* (32:7), 2003, pp. 1159-1177.
8. Iansiti, M. and MacCormack, A. "Developing Products on Internet Time," *Harvard Business Review* (75:5), 1997, pp. 108-117.
9. von Krogh, G., Spaeth,S., and Lakhani, K. "Community, joining, and specialization in open source software innovation: a case study," *Research Policy* (32), 2003, pp. 1217-1241.
10. Lane, P., and Lubatkin, M. "Relative absorptive capacity and interorganizational learning," *Strategic Management Journal*, (19:5), 1988, pp. 461-477.
11. Mockus, A., Fielding, R., and Herbsleb, J. "A Case Study of Open Source Software Development: The Apache Server," in *Proceedings of the 22nd International Conference on Software Engineering*, Limerick, Ireland, 2000.
12. Nambisan, S., Agarwal, R., and Tanniru, M."Organizational Mechanisms for Enhancing User Innovation in Information Technology," *Management Information Systems Quarterly* (23:3), 1999, pp. 365-395.
13. Stewart, K. and Gosain, S. "The Impact of Ideology on Effectiveness in Open Source Software Development Teams," *Management Information Systems Quarterly*, 2006.
14. Stewart, K., Ammeter, T., and Maruping, L."A Preliminary Analysis of the Influences of Licensing and Organizational Sponsorship on Success in Open Source Projects," in *Thirty-eighth Hawaii International Conference on System Sciences*, 2005.
15. Ye, Y. and K. Kishida. "Toward an Understanding of the Motivations of Open Source Software Developers," *Twenty-Fifth International Conference On Software Engineering*, Portland, Oregon, 2003.
16. Zhao, L. and S. Elbaum. "Quality assurance under the open source development model," *Journal of Systems and Software* (66), 2003, pp. 65-75.