The Pennsylvania State University

The Graduate School

The Mary Jean and Frank P. Smeal College of Business

BEING OPEN IN A CLOSED WORLD: ESSAYS ON INNOVATION IN OPEN SOURCE NETWORKS

A Thesis in

Business Administration

by

Girish Mallapragada

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2008

The thesis of Girish Mallapragada was reviewed and approved* by the following:

Gary L. Lilien
Distinguished Research Professor of Management Science
Thesis Co-Advisor
Co-Chair of Committee

Rajdeep Grewal
Dean's Faculty Fellow and Professor of Marketing
Thesis Co-Advisor
Co-Chair of Committee

William T. Ross
Professor of Marketing

John C. Liechty
Associate Professor of Marketing and Statistics

Reka Albert
Associate Professor of Physics and Biology

Johann Baumgartner
Professor of Marketing
Charles and Lillian Binder Faculty Fellow
Head of the Department of Marketing

*Signatures are on file in the Graduate School

ABSTRACT

My thesis consists of a brief introduction, two essays, and a brief conclusion. The two essays that form the core of my thesis investigate innovation coordination in the open source community model of product development. For this purpose, a social network approach is used to understand the effects of network structure that arises due to relationships between open source projects and developers.

In the first essay we build on the notion of the criticality of social networks for the new product development process. We rely on organizational founding literature to suggest that the network structure at the inception of a new product development project would have a lingering effect on the time to development of the new product. We test this assertion in the context of open source software development projects, where the open source community has emerged as a viable alternative to traditional firm-based software development initiatives. The open source community builds products through online collaboration, products whose source code is in the public domain. In this open source environment, we conceptualize a product development process consisting of two stages: (1) some projects transition from ideas to some development activity and (2) for the projects that make the transition at the first stage, some projects mature into 'marketable' products. Specifically, we hypothesize how network structure that surrounds a project at inception will play a role in determining which projects will stay inactive as against being active (stage 1), and influence the time to market for that the projects that become active (stage 2). We test the hypotheses with data from 817 new open source projects from SourceForge.net, the largest forum of open source projects on the Internet. Results from simulated maximum likelihood estimation of a Type II Tobit model specification provide

support for the hypotheses that network structure at inception of a project impacts whether the project becomes active and the time to first release of the project.

In the second essay we address issues relating to the participation of large firms in the open source software development community. The network structure that exists among open source projects is changing continuously due to the nature of collaboration in such communities. As the value and quality of resources available to the open source projects depends on the network configurations that surround the projects, it becomes important to understand what impacts these changes and what the consequences will be. Relying on social capital theory, we develop a model in which projects change their network connections by constantly evaluating the utility they derive from their current location in the network structure. We hypothesize that two important network characteristics namely, embeddedness and structural holes affect the network change process. We test the hypotheses with data from 43 open source projects that are being sponsored by a large firm on SourceForge.net, the largest forum of open source projects on the Internet. We build a stochastic actor-oriented model to specify the effects of social capital on the dynamics of network evolution. We estimate our model using a method of moments approach and find support for the hypotheses that network driven social capital plays a key role in determining evolution of network structure and success of open source projects.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to convey my deep gratitude to my mentors, Gary Lilien and Rajdeep Grewal. They have provided me the necessary support and guidance to become an able researcher in a competitive academic setting. I consider my dissertation as the most important outcome of my association with my mentors and therefore feel that no acknowledgement can fully capture my gratitude to them.

I would also like to thank professors William Ross, John Liechty and Reka Albert for serving on my committee and for making valuable suggestions and providing timely and quick feedback when required.

Besides my mentors and my committee members, I would also like to acknowledge the entire faculty of the marketing department and most importantly my colleagues, peers and seniors in the PhD program. I would like to particularly thank Sunil Thomas for putting up with my junior year rants and providing valuable advice at important junctures during my program. I would like to specially thank ISBM for providing me with the necessary financial support during my program, and its staff for putting up with my frequent requests and needs. I would also like to acknowledge that the data for my dissertation has been obtained from the SourceForge.net warehouse hosted at the University of Notre Dame. I would also like to acknowledge the support of my family during the course of my doctoral work. I thank my mother Savithri Raju, my aunt Alivelu, and my sisters Lavanya and Nagamani and their families. Finally, I thank Radhika, my best friend and my wife, who has lived an almost impossible dream with me for the last five years.

“Everyone is a new door opening into other worlds. Six degrees of separation between us and everyone else on this planet. But, to find the right six people...”

- Ouisa Kittredge in *Six Degrees of Separation* by John Guare

“The desire to be rewarded for one’s creativity does not justify depriving the world in general of all or part of that creativity.”

- Richard Stallman

Chapter 1

INTRODUCTION

It would not be hyperbole if I mention that the Internet would almost completely "vanish", if not for open source software (OSS) and following are some reasons why that might happen. The Google juggernaut is powered by thousands of servers that run Linux - an OSS, Yahoo is powered by the FreeBSD operating system - an OSS, and BIND, an OSS helps match IP addresses such as 72.14.207.99 to www.google.com. Clearly, it is tough to get around the Internet without these open source products. 70% of all web-severs that host content that makes up the Internet are run by Apache, an OSS and embedded Linux runs on routers that plumb the Internet pipelines. Without any or some of these OSS products, the Internet would not be as ubiquitous as it is today and the world would definitely be a different, less exciting place.

OSS is important, necessary and makes tremendous contribution to everyday life. Firms rely on OSS to run their operations, individuals use OSS for their daily needs and the importance of OSS is growing by leaps and bounds in our modern world. All the more surprising is the reality that many OSS products are not developed by firms, but by users and volunteers. Somehow, innovation is being coordinated by the crowds, with little or no central co-ordination. An intriguing context for sure, and yet, as an academic community, we know very little about how these OSS products are built, sustained and delivered. My thesis is an effort to develop insights on the coordination of product development in the OSS world.

My dissertation consists of two essays through which I study important aspects concerning innovation in the open source community model. The two essays are titled:

1. Born to Win: The Effects of Network Structure at Project Founding on Time-to-Market for Open Source Development Projects.
2. Evolutionary Dynamics of Open Source Projects: A Study of a Large. Firm Sponsored Open Source Projects.

The two essays deal with different yet related aspects of new product development in the open source model of collaborative innovation. I primarily rely on social network analysis to develop insights on aspects relating to success and coordination of OSS products. In this brief introduction to my dissertation I provide a 1) brief overview to the open source community model of software development, 2) discuss how social networks play an important role in this community driven product development context, and 3) briefly discuss my motivation in doing this research and the main contribution of my thesis.

1.1 Open source community model of product development

OSS development is beginning to emerge as a dominant alternative product development model to the traditional firm-driven model in the information technology world (Lerner and Tirole 2005; von Hippel and von Krogh 2003). Products such as the Linux operating system, Mozilla Firefox web browser, and Apache web server among others, that are developed by individuals who only have virtual contact through online communities, have taken the commercial world by storm (Lerner and Tirole 2002). On the

B2B front, major hardware vendors such as IBM, HP, Dell and Sun Microsystems have announced long-term commitments to bundle their hardware with software from the open source world to reduce the total cost of ownership for their enterprise clients (Pallatto and Spooner 2005; Weiss 2004). Dell, the largest PC manufacturer has recently announced that it will start shipping desktops and laptops loaded with Ubuntu, the most popular Linux distribution for desktops (Menchaca 2007). Sources project that the sales of Linux based shipments would be close to $35 Billion by 2008 and the increase in success stories of open source software is increasing the adoption momentum (Cloer 2005). Oracle has already started providing support to its customers for their Linux installations (Whitaker 2007) and even Microsoft, a bastion of all that is non-open source, has announced a partnership with Novell to collaborate on open source software (Lowry 2006).

The “open source” movement has its origins in the Free Software movement pioneered by Richard Stallman, a former researcher at the MIT Artificial Intelligence lab. Mr. Stallman embarked on a grand project for building an operating system from scratch, after the MIT administration disbanded a widely prevalent hacker culture in the group. Mr. Stallman’s efforts, however, only resulted in the development of key components that were required to build a complete OS (e.g., the GCC and Emacs), but fell short of the kernel, the core of an operating system. Meanwhile, far from all the eccentric spirited work of Mr. Stallman, a young Finnish grad student by the name of Linus Torvalds was looking for technical help with porting a Unix-like OS called POSIX onto his home machine. Frustrated with the unavailability of information, he decided to develop an operating system from scratch that was “like POSIX”. Finally after a solo-effort, he posted an email about his new operating system on the *usenet* groups, and the rest as they say is history.

Individuals who were waiting for a free and "open" OS such as the one developed from scratch by Linus Torvalds, joined the movement. The project which started out in a young grad student's home is currently one of the largest infrastructure software products ever to be built; running well into several million lines of code. The most fascinating thing about this development effort is that the product was built by groups of volunteering users who were not shackled by organizational boundaries. The Linux kernel project (hosted at http://www.kernel.org) was a huge step towards a free and open intellectual environment in the digital world.

Open source collaborative activities are often organized as software consortia on the Internet, (for e. g., The Apache Software Foundation at http://www.apache.org and the Linux Kernel project http://www.kernel.org). These consortia provide a common platform for developers, working across countries all over the world, to collaborate and develop world-class software products. Collaborative movements are gaining widespread popularity and gaining legitimacy as is evident from some European governments decision to adopt open source offerings for their IT architecture, IBM donating major chunks of software code to open source software consortia, and more mainstream products such as the Linux operation systems and the Firefox web browser finding their way into household computers (Marson 2004).

Clearly, a new paradigm of product development has begun to emerge in the high-technology software world and is giving rise to products that have wide commercial potential and can change the very competitive structure of high-technology markets (Lessig 2001; von Hippel 2001). Given the widespread appeal of collaborative innovativeness and its implications to intellectual property rights, the movement has started to influence not just new product development but the very notion of innovation in a number of other fields (Lerner 2005). Insiders from big technology firms warn that open source is here to stay and it is important that firms develop an IT strategy that acknowledges the importance of the phenomenon, lest they lose out strategically in the long run (Fink 2003; Tiemann 1999). The importance of the movement itself cannot be undermined, as we stand in the threshold of a paradigmatic change, that is about to redefine how innovation and knowledge development would occur in the years to come (Lessig 2001; von Krogh and von Hippel 2006).

The software industry is in a state of continuous evolution and faces complex issues such as product bundling, competition among standards and dynamic pricing mechanisms. There has been an increasing interest in academia on the open source movement in the recent years (Lee and Cole 2003; von Hippel and von Krogh 2003). MIT's initiative on sharing open source research papers, special issues on open source research in reputed academic journals such as Management Science, have set the stage for future research on this novel substantive context.

Currently, very little insight is available from academic research in new product development literature as to how the innovation process plays out in the open-source world (von Hippel and von Krogh 2003). There is some anecdotal evidence as to why some projects succeed and others fail, and some researchers have started to look at motivations of developers and their participation in the movement (e.g., Bagozzi and Dholakia 2006; Lakhani and Von Hippel 2002; Shah 2006), effects of social network structure (e.g., Grewal, Lilien, and Mallapragada 2006) on open source project success, and theoretical models addressing competitive strategies (e.g., Casadesus-Masanell and Ghemawat 2006).

However, much needs to be done before a comprehensive understanding of this complex and interesting phenomenon can be developed. Most of the work concerning the open source revolution has concerned itself in explaining the organizational structure of the collaboration process (Lee and Cole 2003) or in encapsulating the ramifications of the movement (Lerner and Tirole 2002; von Hippel and von Krogh 2003). There are a number of experiments in the real world that aim to appropriate rents from the open source world. To limit one's research inquiry into developing models of appropriating rents by using the open source movement would be scratching the surface of the open source revolution. At the heart of the open source context are very fundamental issues to economic organization – intellectual property rights, innovation governance and extraction of commercial rents. Clearly, an effective debate among the camps is necessary to reach a broader consensus on important issues concerning economic organization in the open source context.

In my dissertation I have made a serious attempt to understand some of the important determinants of new product success in the open source world and how relationships among various open source projects evolve over time. I use social network analysis as the primary tool to understand this complex real world phenomenon and in the next section I describe the importance of social networks as a tool in the context of open source software.

1.2 Social networks

The birth of online communities and the popularity of sites such as Youtube.com, Facebook.com, Sourceforge.net, and Wikipedia indicate new implications for a more connected world. The recent acquisition of Youtube.com by Google for 1.65 billion USD indicates that interest in new customer-interactive technology models is on the rise. As these social networking sites gain prominence, we, as an academic community need to develop theories and test them to develop a deeper understanding of this interesting and important context. Social network analysis provides a useful framework to study such communities. Earlier literature in marketing and management has applied a social network analysis lens to address many important issues at the heart of managerial theory (e.g., Van den Bulte and Moenaert 1998; Gulati and Garguilo 1999). More importantly, social network analysis facilitates a systemic approach to understanding coordination processes within and between organizations.

Open systems theory suggests that an organization is a thriving entity that is in a continuous two-way exchange process with its environment (Katz and Kahn 1966). The environment is characterized by numerous other entities that engage in transactional processes with the focal organization (Achrol 1997). Therefore, every organization can be viewed as being embedded in a network of relationships that are part of its environment. These relationships form the conduit for the transfer of information and resources to and from the organization to its environmental constituents. The relationships in this network are a heterogeneous set that allow the organization to transact with various entities. For instance, firms have a different relationship with their suppliers than with their distributors and governmental agencies. The network connections between a firm and its suppliers are configured in a way that appropriately facilitates the flow of resources and information and these connections would differ from those that exist between the firm, and say, a governmental regulatory agency. This social network forms the basis of the firm's environment or its neighborhood. Earlier research in organizational theory has investigated the antecedents of such interorganizational networks (e.g., Gulati and Gargiulo 1999) and the impact of such networks on performance (e.g., Ahuja 2000a; Ahuja 2000b; Grewal et al. 2006). Clearly, social capital that accrues to individuals and firms through their location in a network structure plays an important role in determining their success.

The open source environment is uniquely characterized through its collaborative product development model in a networked environment. Unlike coordination processes within traditional organizations, the processes in the collaborative model have evolved due to continuous interactions between developers through their membership in open source projects. Moreover, the developers are free to join and leave open source projects based on their preferences and thereby create an ecosystem in which some open source projects perish due to lack of interest in the community whereas some projects thrive actively. This connectivity between open source projects and developers and its subsequent impact on the performance of open source projects forms the basis of my dissertation.

1.3 Motivation and contributions

I have always been fascinated by the dynamic nature of the information technology industry, particularly the sub-domain of software products. Open source is a passionate context involving proprietary firms on one end and free market thinkers on the other. The verdict on open source evolving to become a challenge to closed source alternatives is not yet out. There have been some important successes in the recent past as the participation of large firms has become a rule rather than an exception. IBM's greater involvement with the community, Oracle's renewed interest in supporting Linux installations, Dell's recent decision to ship Linux installed computers are examples of how strategic commitments are being made by large firms. It is now increasingly clear to these firms that they have to compete by adapting to changes occurring in the broader economic world.

The dynamic software industry is experiencing a fundamental change as to how products will be designed and distributed, and as to how firms will compete. The open source context seems to be the catalyst for ushering in that change. The open source world is like a microcosm with its own ecosystem where creative destruction is the rule of the game. Products compete for resources and those that are the fittest survive, but only till the next better product comes along. Open source product teams don't rest once their releases become stable, they just move on with further improvement. The open source world is characterized by this process of creative destruction and for that reason is surely the model for future growth in the software industry. The famous Austrian economist Schumpeter clearly drives home the point when he says (Schumpeter 1975, pg. 82),

> "Capitalism, then, by nature a form or method of economic change not only is never but can never be stationary. And this evolutionary characteristics of the capitalistic process is not merely due to the fact that economic life goes on in a social and natural environment which changes and by its change alters the data of economic action;.."

Clearly, change is inevitable in the economic world and change always plays out in a broader social context (Granovetter 1985). The important thing to note is that managing the social context sometimes becomes more important as economic outcomes get embedded deeper in social action.

Therefore, my dissertation applies a social network approach to innovation coordination processes in the open source world. By building on earlier research in social networks and open source collaborative models (e.g., Grewal et al. 2006), the first essay investigates the effects of social capital of the project founders on the performance of open source projects. The second essay extends the understanding of open source project networks by investigating social network effects on the evolutionary dynamics of open source project networks. My thesis provides a richer understanding of crucial processes inherent to the coordination of open source collaboration. The core of my contribution lies in the development of an in-depth understanding on the nature of network effects in innovation coordination.

In principle, my effort draws its inspiration from earlier research on user-driven innovation (e.g., von Hippel 1989) that investigated the benefits of involving users in innovation development. More importantly, my thesis introduces a new substantive and important area of research to the field of marketing. “Connecting Innovation and Growth” and “Connecting Customers with the Company” are stated as two of the most important research priorities by the Marketing Science Institute for the years 2006-2008, and both these issues are the foundations for my thesis. In addition, there is a resurging interest in online business models driven by user involvement and customer participation (popularly referred to as Web 2.0), and development of hybrid models of revenue sharing (e.g., Google’s adsense program). Moreover, firms are more willing to experiment with hybrid models of governance (e.g., IBM’s work with the Apache Software Foundation). These new developments and the road ahead are bound to raise a number of interesting questions to practitioners and researchers in diverse fields such as strategy, organization theory and learning, marketing, and innovation, among others. I believe my thesis is a significant step towards developing a deeper understanding of this new domain

Chapter 2

BORN TO WIN: THE EFFECTS OF NETWORK STRUCTURE AT FOUNDING ON OPEN SOURCE DEVELOPMENT PROJECTS

2.1 Introduction

The study of the effects of social networks is emerging as an important area in the field of marketing (e.g., Frenzen and Nakamoto 1993; Iacobucci and Hopkins 1992; Rindfleisch and Moorman 2001). Researchers in this area have studied the importance of social networks for the performance of new products (e.g., Grewal et al. 2006), new product introduction strategy (e.g., Mahajan, Muller, and Kerin 1984), word-of-mouth effects (e.g., Brown and Reingen 1987; Frenzen and Nakamoto 1993; Reingen, Foster, Brown, and Seidman 1984), communication among R&D, manufacturing, and marketing team members (e.g., Van den Bulte and Moenaert 1998), and social contagion effects in innovation diffusion (e.g., Van den Bulte and Lilien 2001). This study builds on this research to investigate the importance of social networks at the birth, i.e., inception of a new project in the open source community on the time to market – an important metric for assessing the effectiveness of new product development projects (e.g., Bayus, Jain, and Rao 2001; Griffin 1997).

In this study we seek to investigate the importance of social networks for new product development projects in the context of open source software development, where information on product development activities is readily available in the public domain –

unlike traditional product development that happens within firms and product development activities are proprietary. Open source software development that relies on coordination efforts of networks of volunteers is beginning to emerge as a dominant alternative product development model to the traditional firm-driven model in the information technology world (Lerner and Tirole 2005; von Hippel and von Krogh 2003). The open source product development model is gathering momentum as dominant firms such as IBM, HP, and Compaq begin to announce long-term commitments to bundle their hardware with software from the open source world, to reduce the total cost of ownership for their enterprise clients (Pallatto and Spooner 2005; Weiss 2004). Industry sources project that the sales of Linux based shipments would be close to $35 Billion by 2008 (IDC 2004) and an increase in the number of success stories is increasing the adoption momentum of Linux and other open source software (Cloer 2005). New product development projects in the open source community are launched by one or more individual participants from the open source community, where these individuals may work on various other projects in addition to the new projects they initiate with other contributors. Therefore, these individuals form a social network through the common projects that they work on – the so called "affiliation networks" in the social networks literature (Wasserman and Faust 1999).

We conceptualize the development of software in the open source environment as a two-stage process model. In the first stage, projects are initiated as ideas such that some ideas remain inactive and see no developmental activity, whereas other ideas become active and developmental activities are initiated. In the second stage, for those projects that become active, we observe the time to release if a working version is released or that the projects are still work in progress (censored). For example, *Azureus* – a Java based bit-torrent client, listed as the most active product under development on SourceForge.net is downloaded over 50,000 times a week and is quite successful in its product class, while *Wap Getmail*- a product that was launched 3 years ago is yet to release a working product.

To explain how network structure might influence both stages of the above described new product development project, I rely on research in organizational founding that suggests that the environmental conditions at founding of organizations have a persistent effect on the evolution and success of the organizations (e.g., Baum and Oliver 1991; Delacroix and Carroll 1983; Swaminathan 1996). We build on this notion to suggest that the network structure, i.e., the associations of the founders, at the time of initiation of the projects characterizes the ambient conditions at founding for open source projects and, therefore, this network structure should have a persistent effect on the outcomes associated with the new product. Specifically, we seek to understand the effect of network structure surrounding these new product initiatives (projects) on the likelihood of the new product initiatives being active or inactive, and subsequently on the time-to-market for those products that become active.

Research in social networks has suggested that the two issues of 1) coordinating *action* and 2) initiating *ideas* are fundamental to understanding the effects of network structure on organization (e.g., Obstfeld 2005). The action problem is effectively solved by networks which exhibit a high level of interconnectivity among its nodes (Granovetter 1985) and the idea problem is solved effectively by sparsely connected networks (Burt 1992). The concept of embeddedness effectively captures the extent to which a node is entrenched or connected in the network (e.g., Grewal et al. 2006; Gulati and Gargiulo 1999), and the notion of structural holes captures the degree to which sparseness is present around any given node (e.g., Burt 1997; Obstfeld 2005). Therefore, by relying on this rich stream of literature in social network capital we operationalize the effect of network structure through the two notions of embeddedness and structural holes. The "network embeddedness" or the degree to which various nodes are central in the network (in our context, both developers and products), and "structural holes" or the degree to which a node can form alternative links in the network, can be empirically determined by the pattern of relationships that exist among the founders of new projects (e.g., Grewal et al. 2006; Mizruchi and Potts 1998). Projects would derive social capital, or in other words, access to key resources through the networks of their founders and this social capital would be vital for the success of the projects (e.g., Grewal et al. 2006).

To test the proposed model for the effect of network structure at birth on the success of the projects, we collect data on 817 projects that were initiated during first ten days of January 2003 on SourceForge.net, a thriving ecosystem of open source initiatives with over 138,000 products and over 1.47 million users (e.g., von Hippel and von Krogh 2003). Following Terza (1998), Greene (1995), and Greene (2006) we formulate a model that captures the first stage (i.e., projects either stay inactive or become active) in our two-stage process model, through a selection mechanism and the second stage (i.e., either time to market is observed or projects remain as work in progress) through a censored duration model (the popular Type II Tobit model specification). Results from the simulated maximum likelihood estimation provide support for the hypothesized effects. Specifically, we find that embeddedness of the founders increases the probability of a new open source project becoming active and reduces the time-to-market for the project. Lack of structural holes in a project founder's network neighborhood increases the probability of the new project becoming active and shortens the time-to-market. There is also a synergistic effect between high levels of embeddedness and the lack of structural holes such that fewer the structural holes for the founders, greater is the effect of embeddedness on the probability of the project becoming active and shorter is the time-to-market.

The second chapter is organized in the following manner. We begin by providing information on (1) the open source movement, (2) the importance of social networks for new product development process, (3) the two-stage process model specification, and (4) the effect of network structure on time to market of new projects. We then elaborate on our research methodology where we describe the data collection procedure, our measures, and the model estimation framework. Next we present our results and we end by discussing the implications of this research for theory and practice. Our research enriches the theoretical understanding concerning the effects of social capital in affiliation networks by specifying the nature and effect of embeddedness and structural holes in coordinated economic action. Methodologically, we implement a generic modeling framework that addresses the issues of duration modeling in the presence of sample selection. Substantively, our results provide meaningful insights on the importance of network effects to managers and firms interested in participating in the collaborative open source community model.

2.2 Social networks and open source product development

In this section, we first provide a brief background on the open source movement and then detail the conceptual background, hypotheses and our research model.

2.2.1 The open source movement

Computer software products have been traditionally developed through a "closed source" model, meaning that customers do not have access to the "source" of the product and their rights are limited only to the usage within the customization constraints provided by the manufacturer. As the marginal costs of software production are close to zero, such a model of not revealing the "source" is unique to the software product world. Therefore, product development firms resort to a "closed source" model that does not adhere to technology standards to create customer lock-in and thereby seek to inhibit competition (e.g., Economides 1996). The open source movement, however, champions the idea of opening up the source code of products through a more liberal "copyleft", an open licensing model, that allows users to alter the functionality of the software to fit their needs (von Hippel and von Krogh 2003).

The roots of the open source ideology can be traced back to the events surrounding the exit of Dr. Richard Stallman from MIT's AI lab back in the 70's. Dr. Stallman, then a researcher at the MIT labs, was disgruntled by the dismantling of the free hacker culture at the lab, decided to create a "free" operating system, that would not be tied down by any proprietary technology or licensing constraints. His efforts snowballed into a world-wide movement and resulted in a wide range of products being developed by the community of volunteers who believed in his ideology. As of today, open source products dominate their target markets such as the Linux OS in the enterprise server market segments, while others such as the Firefox web-browser and JBoss application server, have emerged as credible threats to their closed source counter parts.

Firms such as IBM and HP have recognized the impact of the community driven product development model on innovation and have begun to experiment with hybrid models of product development. For example, IBM has donated chunks of various application level code and patents to foundations such as the Apache Software Foundation that organize and coordinate new product development (Galli 2005b), and HP has followed suit by releasing the entire suite of its printer drivers under the General Public License (GPL), the major licensing mechanism for open source software. Firms such as IBM, Samsung, and Sun among others have come together to form a patent pool that can help fuel the software product development in the open source world (Cover 2005). Major IT hardware vendors such as Dell, IBM and HP have announced plans of supporting open source products on their platforms (Barker 2006).

Apart from these firms that do not completely rely on the community for developing new products, there are firms such as RedHat Inc. and JBoss that completely rely on the open source community for software product development and maintenance. Moreover, the venture-capital inflows into firms that have a product strategy accommodating the presence of the open source community are climbing (LaMonica 2005). The flagship product of the open source community, the Linux operating system, continues to gain greater momentum in the enterprise-server space and with more applications, such as SugarCRM and Compiere (an enterprise resource planning software),

are gaining widespread acceptance. It seems that customers have begun to see the benefits of this alternative model of intellectual property management.

2.2.2 New product development and social networks

Starting with Bass (1969)'s influential paper on diffusion of durable goods, there is a rich body of research in marketing on issues dealing with new products (for a detailed review see Hauser, Tellis, and Griffin 2006; Mahajan, Muller, and Wind 2000). Besides analyzing the diffusion process, marketing researchers have also looked at new product issues such as sales takeoff of new products (e.g., Agarwal and Bayus 2002), entry-timing (e.g., Bayus, Jain, and Rao 1997), entry strategy (e.g., Mahajan et al. 1984), new product announcements (e.g., Bayus et al. 2001), WOM effects (e.g., Bayus 1985; Mahajan et al. 1984), willingness to cannibalize in radical product innovation(e.g., Chandy and Tellis 1998) and product life cycle(e.g., Bayus 1998; Bayus and Putsis 1999), among others. More importantly, research concerning network effects in new product development, both in terms of WOM effects (e.g., Frenzen and Nakamoto 1993), and in the development processes itself (e.g., Van den Bulte and Moenaert 1998), is beginning to emerge.

Unlike, traditional product development which requires enormous investments in time and resources, much of the initial investment in new product development in the open source world resides in the skills of individuals involved in the initiation of the new product. Products initiated by individuals occupying key locations in the network of volunteers (i.e., by working on crucial products), would be better positioned to take off than those that are initiated by individuals with limited access to social capital.

Ethnographic accounts of the open source movement highlights the importance of attracting and retaining a good army of volunteers in the initial stages of the product development process (e.g., Raymond 1999; Torvalds and Diamond 2001). In contrast, there are instances wherein the product was so revolutionary that it automatically attracted a community of volunteers. [1] However, the importance of the network of individuals who work on these products and ensure their growth and survival cannot be downplayed (e.g., Grewal et al. 2006). These volunteering individuals develop ties and share knowledge and resources all through Internet forums such as www.apache.org and www.sourceforge.net. These forums or online communities serve as platforms for developing ties to products and individuals listed on these forums, thereby ensuring transfer of information and resources on the network (e.g., von Hippel 2001).

Academic interest in the open source product development model is growing and research in diverse fields has looked at some preliminary aspects of the phenomenon (e.g., von Hippel and von Krogh 2003; von Krogh and von Hippel 2006). However, given the critical organizational implications of the open source model for new product development, marketing strategy and competitive dynamics in the information technology industry, it is highly crucial to develop testable theories and empirical models in the domain (von Hippel and von Krogh 2003; von Krogh and von Hippel 2006).

Literature in technology management has used an evolutionary approach to study the birth and death of organizations (e.g., Carroll and Delacroix 1982; Swaminathan 1996) and proposed that the environmental conditions at the time of founding (birth) will have an

[1] For example, the first version of the kernel that forms the core of the now ubiquitous Linux OS was developed alone by Linus Torvalds, a Finnish undergrad student at that time. The kernel later attracted a number of talented individuals who worked on its improvement.

enduring effect on organizational survival. Similarly, research in new product development has taken an evolutionary approach to understand the product life cycle model (e.g., Lambkin and Day 1989; Tellis and Crawford 1981). We draw on this extensive body of literature in technology management and new products that suggests that the network structure at the time of inception of an open source project signifies the ambient conditions of the project's habitat and, therefore, will have an impact on the development cycle of the open source project.

The presence of network effects in various economic contexts, including but not limited to interorganizational relationships and virtual communities is increasingly being acknowledge by academic research (e.g., Grewal et al. 2006; Gulati and Gargiulo 1999). More generally, research in sociology has been quite assertive on the role of network structure in the formation of social capital (e.g., Lin 2001; Zukin and DiMaggio 1990). Studies have linked finding new jobs (e.g., Granovetter 1973), access to financial capital (e.g., Uzzi 1999) and importance of employee referrals (e.g., Faust and Kimball 1985) to social networks. The social networks approach of modeling social phenomenon by incorporating the complex structural effects of networks, owes its origins to research in organizational sociology (e.g., Granovetter 1973).

The two issues of *coordinating action* and *initiating ideas* are crucial to understand the network structural effects (e.g., Obstfeld 2005). Networks in which the nodes are highly connected to each other are very efficient in solving the *action* problem as the high degree of interconnectivity allows a smooth dissemination of information and flow of resources, thereby facilitating coordination (Granovetter 1985). Networks that are sparsely connected have been found to be effective in solving the *idea* problem as they present a conducive opportunity for generation of new ideas through the isolation and presence of unique sets of resources in the network (e.g., Obstfeld 2005). The concept of embeddedness effectively captures the extent to which a node is entrenched or connected in the network (e.g., Grewal et al. 2006; Gulati and Gargiulo 1999), while the notion of structural holes captures the degree to which sparseness is present around any given node (e.g., Burt 1997; Obstfeld 2005).

Following the above mentioned stream of literature on the value of social capital and literature on organizational founding that suggests that the resources available to an organization at its founding will have lasting impact on its future survival and performance (e.g., Freeman, Carroll, and Hannan 1983; Swaminathan 1996), we propose that two aspects of network structure, namely embeddedness (e.g., Gulati and Gargiulo 1999; Uzzi 1996) and structural holes (e.g., Burt 1992; Obstfeld 2005) will have a lasting impact on important aspects of new product development in the open source community model. Embeddedness has been used in a variety of research contexts to measure the degree to which an entity is enmeshed or entrenched in a web of relationships. Although scholars have used various operationalizations of the embeddedness construct (e.g., Gulati and Gargiulo 1999; Uzzi 1999), the underlying notion of being densely connected in a web of relationships is common across all these operationalizations. Embeddedness measures the degree of connectivity of a node in the network, and greater the embeddedness, greater is the degree to which the node is enmeshed in relationships. The concept of structural holes captures the extent to which alternative opportunities are present for any given node in the network. Burt (1992) proposed that those nodes that are less constrained in their network neighborhood have more structural holes and therefore will be able to exploit the opportunities available by filling in such structural holes.

2.2.3 The two stage process – model specification

Based on our observation of open source development activities, we conceptualize a two-stage new product development process, the first stage wherein projects that all start as being ideas either remain inactive (no developmental activity), or move onto an activity track as contributors sustain the development trajectory through their participation. In the second stage, projects that are on development trajectory either experience releases of the working version of the product, or remain as work in progress during the observation window. Clearly, a two-stage process conceptualization of the transition of open source products from being ideas to working versions provides us an opportunity to understand the effects of network structure in open source product development. We illustrate this selection process conceptually in Figure 2.1.

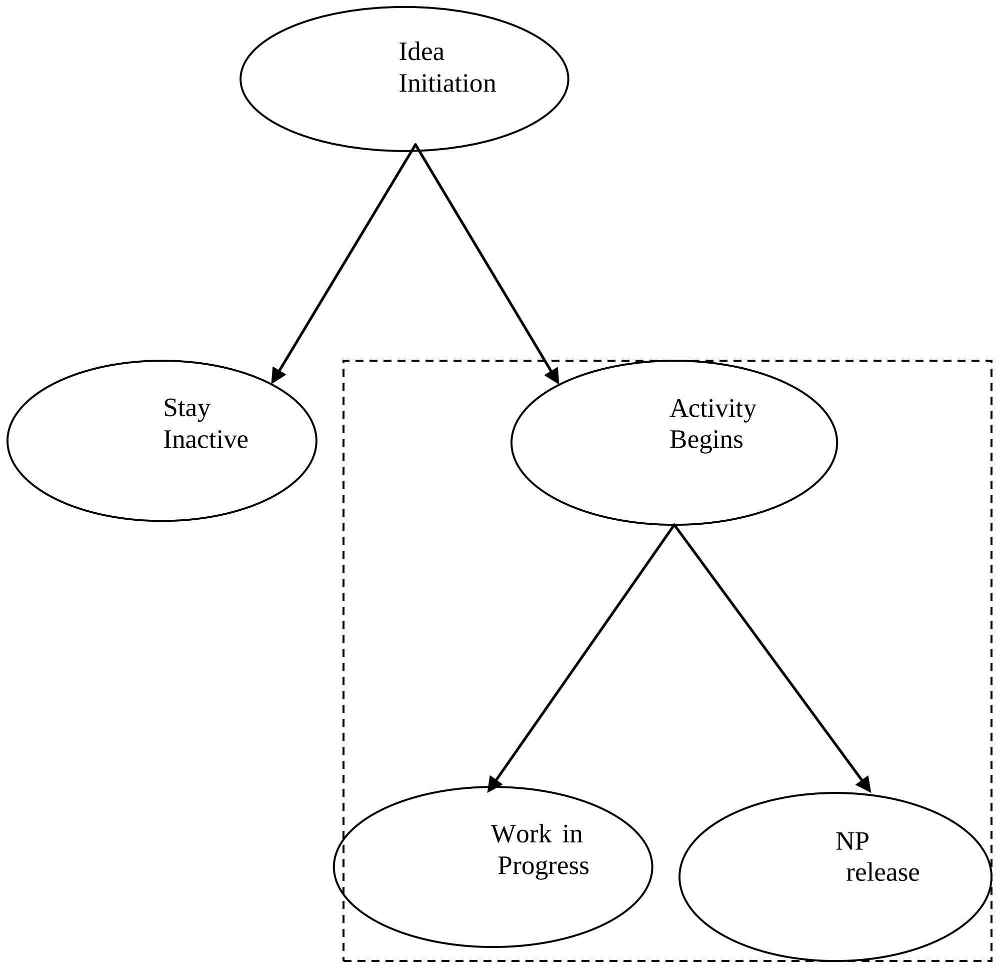

Figure 2.1: Two stage process model for software development in the open source environment

The run-away success of the open source model in developing a highly robust and scalable product such as the Linux operating system resulted in the birth of thousands of product development efforts driven by virtual communities[2]. These products vary on their usage contexts and product complexity (from simple applications e.g., *spamassassin* to complex software e.g., *SugarCRM* for customer relationship management). However, apart from the highly successful products such as the Mozilla web browser, OpenOffice suite, and Azureus (a bit-torrent client), a number of these product initiatives are abandoned. We propose that the network structure surrounding the birth of the project will impact the likelihood of the project becoming active, and this effect can be represented as,

$$y_i^{s^*} = \alpha' x_i^s + u_i \qquad 2.1$$

and,

$$u_i \sim N[0,1] \qquad 2.2$$

where, x_i^s is the vector of explanatory variables that have an effect on the selection process (of whether projects become active or not), including the network structure and control variables, α' is the vector representing the effect of explanatory variables on the latent variable, $y_i^{s^*}$ such that, $y_i^s = 1$ when $y_i^{s^*} > 0$ implying that the project became active and $y_i^s = 0$ when $y_i^{s^*} < 0$ implying that the projects is inactive (where y_i^s is the observed outcome variable of whether a project is active or not).

---

[2] The Apache Software Foundation, SourceForge.net, Mozilla Foundation, Free Software Foundation are some of the more widely known efforts.

The development initiatives that do become active vary considerably in terms of the time required to release the first working version of the product, depending on the participation of the community in the development process. The community can contribute directly by writing programming code, or indirectly by downloading, using the software and reporting bugs, or by posting reviews and participating on the internet discussion forums. It is possible that an initiative may become active remain as work in progress without releasing a working version for a considerable duration. In contrast, an initiative can also reach completion by releasing a working version. We observe a range of outcome characteristics that includes time-to-market, for those products that become active in the first stage. We illustrate this process conceptually in Panel B of Figure 1. We propose that the network structure surrounding the birth of these new products will have a persistent effect on the outcomes associated with product development, specifically, on the time-to-market for these new products and this effect can be represented as,

$$\mu_i^d \mid \varepsilon_i = \beta^{'} x_i^d + \sigma\varepsilon_i \qquad 2.3$$

$$y_i^d \mid x_i^d, \varepsilon_i, c_i \sim f(y_i^d \mid x_i^d, \varepsilon_i, c_i) \qquad 2.4$$

$$\varepsilon_i \sim N[0, \sigma] \qquad 2.5$$

where x_i^d is the vector of explanatory variables that have an effect on the time-to-market, including network structure and control variables, μ_i^d is the mean function with heterogeneity, y_i^d is the observed time-to-market, $f(.)$ is the pdf of the time-to-market variable, and $\beta^{'}$ is a vector representing the effect of explanatory variables on the outcome variable.

Such a two-stage process implies that there might be an underlying selection bias, as the explanatory variables that influence the selection process of whether projects stay inactive or active might not be independent of the outcome variable of interest, that is time-to-market (e.g., Greene 2006). To accommodate this selectivity, we combine the two stages by allowing the error term in the first stage to be correlated with the error term in the second stage. Therefore, the model specification for the two-stage process can now be represented as,

$$\left.\begin{aligned} & y_i^{s^*} = \alpha' x_i^s + u_i \\ & \mu_i^d \mid \varepsilon_i = \beta' x_i^d + \sigma\varepsilon_i \\ & y_i^d \mid x_i^d, \varepsilon_i, c_i \sim f(y_i^d \mid x_i^d, \varepsilon_i, c_i) \\ & [u_i, \varepsilon_i] \sim N[0,0,1,\sigma,\rho] \end{aligned}\right\} \qquad 2.6$$

2.2.4 Effects of network structure on project activity and time-to-market

2.2.4.1 Effect of embeddedness

As projects are initiated by individuals in the network, they would gain access to the social capital of their founders (individuals who started the project) through the associations of the founders in the networked community. These associations are the founders' shared relationships on projects that were already in the system when the new projects were being initiated. These established relationships signify embeddedness of the founders in the network, and provide means of information, exposure, and support to the new projects. The ubiquitous effect of embeddedness in networks is increasingly being recognized (e.g., Kilduff, Tsai, and Hanke 2006) and clearly projects that do not have access to the network, would not be able to gather the attention of the community and the initial resources that are so crucial to their survival (e.g., Lerner and Tirole 2002; Raymond 1999). A higher degree of connectivity eases the flow of information and resources over the network, facilitates transfer of resources and capital and thus solves the *action problem* (e.g., Obstfeld 2005). Embeddedness has been found to have a benign effect on organizational outcomes associated with bio-technology start-ups and ability to raise financial capital, thereby indicating that the degree of connectivity in the network is a suitable proxy to social capital (e.g., Powell, Koput, and Smith-Doerr 1996; Uzzi 1996). Therefore, embeddedness of the founders of open source projects can provide the necessary access to valuable resources and information in the network, and thus increase the likelihood of the project becoming active, and reduce the time-to-completion.

H1a: Higher the embeddedness of a project's founder, higher is the probability of the project becoming active.

H1b: Higher the embeddedness of a project's founder, shorter is the time-to-market for the project.

2.2.4.2 Effect of structural holes

Burt (1992) proposed the concept of structural holes in networks and that their presence in a network indicates the availability of entrepreneurial opportunities in the network. The presence of structural holes also facilitates gaps in the network and thereby solves the *idea problem* (Obstfeld 2005). The presence of structural holes in the network neighborhood of a node indicates the sparseness of connections, thereby providing an opportunity for an entity to extract rent by stepping in to fill gaps in the network. The missing connections allow new ideas to originate and blossom in the network without the interference from denser neighborhoods in the network. Earlier research in the theory of organizational learning suggests that organizations need to balance their search behavior (for opportunities and solutions to problems) by focusing on new unexplored territories (e.g., Cyert and March 1963; Fiol and Lyles 1985) and an environment rich in structural holes provides the ambience for this search. However, for any given node in the network, greater the presence of structural holes in their neighborhood, lesser is the incentive to be involved on any given initiative, as the holes provide an opportunity to focus resources on alternative opportunities (e.g., Burt 1997; Granovetter 1973). If the founders are limited to working on the projects that they have initiated, or stated differently, if the founders are constrained by fewer structural holes in the network neighborhood, it would benefit the projects that they initiated (e.g., Obstfeld 2005).

H2a: The fewer the structural holes in the founder's network neighborhood, the higher is the probability of the project becoming active.

H2b: The fewer the structural holes in the founder's network neighborhood, the shorter is the time-to-market for the project.

2.2.4.3 Interaction effect between embeddedness and structural holes

Obstfeld (2005) acknowledges the presence of two important issues when analyzing the effects of network structure, namely "the idea problem" and "the action problem". Networks with high connectivity among its constituents are efficient in dissemination of information and ease of coordination due to the free flow of resources and information across the connecting links. Such networks where the nodes exhibit high embeddedness are similar to tightly coupled systems – systems where the constituents are linked to each other and are therefore very efficient in coordination and dissemination of information and resources (e.g., Weick 1976). These networks face the "idea problem" due to redundancy of information and equivalence that operates due to the connectivity. In contrast, sparse networks that are filled with structural holes have been empirically found to solve "the idea problem", due to the entrepreneurial and arbitrage opportunities they provide as a result of the gaps in the network (e.g., Burt 1992; Granovetter 1973). These networks face "the action problem" as it is difficult to coordinate in such sparsely connected networks filled with holes. These two forces, i.e., the need to be densely connected to solve the action problem and the need to have enough sparseness to solve the idea problem, are simultaneously operating in all networks.

These arguments highlight that although these two structural characteristics of networks are opposing in nature, they induce benign effects on the outcomes associated with coordination through different mechanisms. New projects gain valuable resources through the diverse sets of and extensive social capital associated with the high embeddedness of the founders (e.g., Grewal et al. 2006; Uzzi 1996). However, when there are a number of structural holes present in the network neighborhood of the founder, the value that can accrue to any given new idea or initiate is diluted as the resources gained through embeddedness get dispersed (e.g., Obstfeld 2005). However, if there are fewer structural holes in the network, the resources and capital gained through embeddedness would be channeled effectively to the new projects that are initiated. Therefore, the positive effect of embeddedness should be more pronounced when structural holes are fewer than when there are more in the project founder's network neighborhood.

H3a: The fewer the structural holes in the network neighborhood of the projects founders, the higher is the positive effect of the project founders' embeddedness on the probability of an open source project becoming active.

H3b: The fewer the structural holes in the network neighborhood of the projects founders, the higher is the negative effect of the project founders' embeddedness on the time-to-market of an open source project.

2.3 Method

2.3.1 Data source: SourceForge.net

The SourceForge.net site hosts around 138,000 open source product development efforts and has over 1.47 million registered users (as of March 24, 2007). It is one of the largest collections of open source software on the Internet, and is essentially a collaborative product development platform that allows individuals to co-ordinate their open source product development efforts. Products that are being developed on this forum belong to a wide range of domains, including database tools, application software, games, text and programming editors, and utility tools, among others. The site offers a rich opportunity to conduct empirical research involving open source product development efforts, and academicians have recommended and used this forum for empirical research (e.g., Grewal et al. 2006; von Hippel and von Krogh 2003).

We collected network structure data on projects initiated on the SourceForge.net site (http://www.sourceforge.net). The data begins in the month of January 2003 and we selected all the projects that were initiated on the first ten days of January 2003 that gave us a sample size of 817 projects. We gained access to the data through the SourceForge.net data warehouse (Madey 2005) and constructed the network structure required for our data analysis in two-stages. The first stage involved accessing the data used in the current study from the relational database underlying the SourceForge.net website. We wrote SQL queries to download data from the research database. The second stage involved transforming the relational data downloaded from the database to network structure data i.e., the sociomatrix (Wasserman and Faust 1999). A sociomatrix for an affiliation network is a matrix with the actors i.e., developers as rows and the events i.e., projects as columns and each of the matrix cells takes either value of 0 if a developer does not work on a project and 1 if he/she does. Figure 2.2 is an illustrative example of an affiliation network and the corresponding sociomatrix.

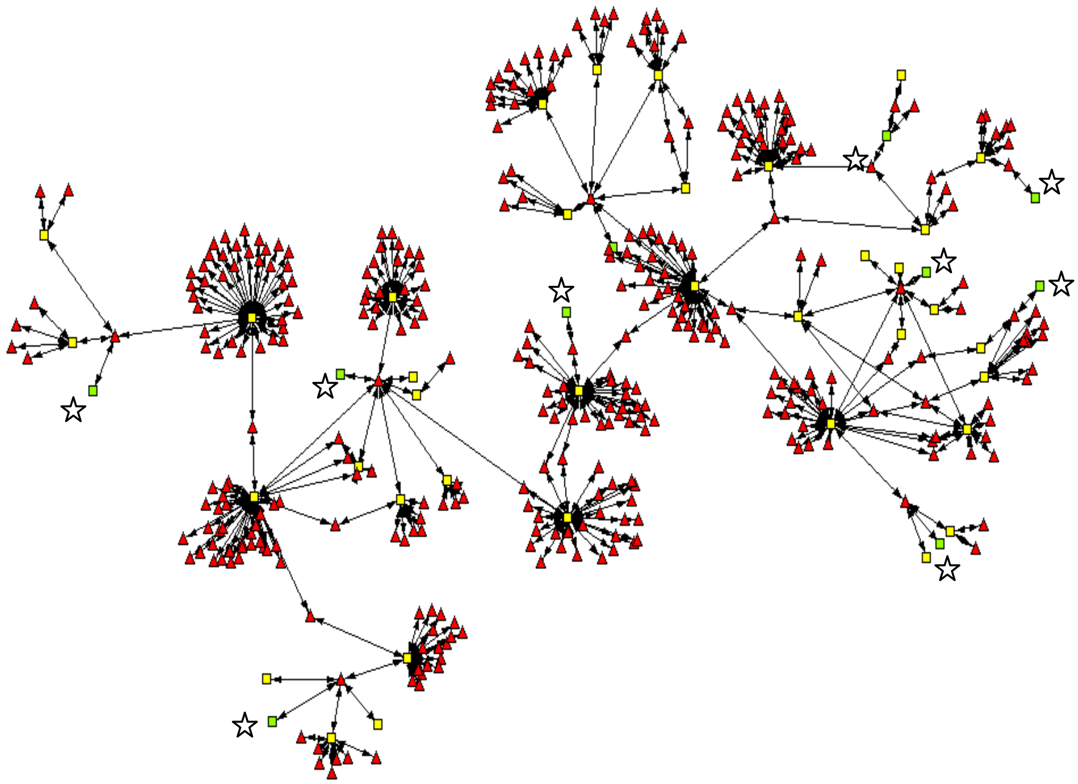

Figure 2.2: Visual representation of the largest continuously connected component of the network

Note:

1. The triangles represent the developers and the squares represent the open source projects.
2. The stars point to projects that have been newly founded.

We first listed all the products whose development efforts were formally initiated in the first ten days of January 2003; we then listed all the developers working on these initiatives which gave us a list of 966 developers. We then constructed a list of other (already existing) projects that these 966 developers were working on excluding the first set of 817 products and this gave us a set of 624 additional projects, following which we listed all the developers working on these 624 projects, which resulted in 1723 additional developers. This two-step procedure resulted in a network of 1441 projects and 2689 developers working on these projects. We transformed this relational data to a sociomatrix with 0's and 1s as elements; where 1 indicates the presence of a network tie between a project and a developer and 0 indicates its absence. Such a chain method or snowballing sampling procedure has been earlier reviewed in the social networks literature (e.g., Wasserman and Faust 1999) and has been used in earlier network research (e.g., Grewal et al. 2006). Our method is conceptually nominalist in its approach and is generally driven by our theoretical concerns (e.g., Laumann, Marsden, and Prensky 1989). Apart from collecting the network structure data at the time of birth, we also collected data on some key variables of interest.

2.3.2 Measures

2.3.2.1 Dependent variables

2.3.2.1.1 Inactive-Active projects

Projects that are initiated on the SourceForge system are classified as either being inactive or active based on whether product development procedures have been initiated on the project. Using this classification we used a binary classification (0-1) and coded all projects that were inactive as “0”s and all projects that were active as “1”s.

2.3.2.1.2 Time to market

A rich tradition of research in new product development has investigated a number of success measures, including probability of success (e.g., Bollen 1999), technical and commercial success (e.g., Mansfield and Wagner 1975), entry timing (e.g., Bayus et al. 1997), and product development cycle time (e.g., Griffin 1997) among others. In a real world dominated by shortening product cycles and rapid technological innovation (e.g., Bayus 1998), it is very important to conceptualize and develop products that meet customer needs in rapid time. Clearly, the relevance of new product release timing is of great importance to the marketing of new products. The technology landscape is strewn with examples of failed products, those that were too early to market (e.g., Apple’s Newton in the PDA market) and those that were me-too followers (e.g., Sony’s Clie in the PDA market). Moreover, due to high dynamism in such product markets, it is very crucial to delineate the factors that drive development time required for new products (e.g., Eisenhardt and Behnam 1995). We measured time to market as the time from the initiation or registration of the product on the SourceForge.net system to the time when the actual release of the product occurs. We used hours as unit of measurement and then log-transformed the data, following earlier suggestions by research in duration modeling (e.g., Kiefer 1988).

2.3.2.2 Independent variables

2.3.2.2.1 Network embeddedness

Consistent with extant research on social capital measures, we operationalize network embeddedness using two measures of centrality: degree centrality and betweenness centrality (e.g., Grewal et al. 2006; Gulati and Gargiulo 1999; Uzzi 1996).

Degree Centrality: We use degree centrality (e.g., Faust 1997) as one measure of network embeddedness, following similar conceptualizations in earlier research (e.g., Grewal et al. 2006, Gulati and Gargiulo 1999). Conceptually, *degree centrality* captures the extant to which the project founder is entrenched in the local network neighborhood, and is an indication of degree of connectivity in the first-order network of the founder. Essentially it measures the number of other projects that the particular individual is associated with in the network. Methodologically, it is calculated using the following expression (e.g., Faust 1997; Grewal et al. 2006),

$$C_D(X_i^A) = X_{ii}^A \qquad 2.7$$

where X^A is the affiliation matrix of the developers and X_{ii}^A is the *i*th diagonal element of the affiliation matrix.

Betweenness Centrality: We use betweenness centrality (e.g., Faust 1997; Freeman 1977) as the second measure of embeddedness (e.g., Grewal et al. 2006). Conceptually, *betweenness centrality* measures the extant to which the project founder acts as the connecting link between other projects on the network. It captures the degree of connectivity in the extended network (indirect connections) of the founder. It can be calculated by counting the number of times the project founder lies on the shortest path between any two projects on the network. Methodologically, it is calculated using the following expression,

$$C_B\left(X_i^A\right) = \sum_{j<k} \frac{g_{jk}(i)}{g_{jk}} \qquad 2.8$$

where $g_{jk}(i)$ is the number of geodesic paths (shortest paths) between any two actors j and k that pass through the node *i*, and g_{jk} is the number of all possible geodesic paths between nodes *j* and *k*. Whenever there were more than two project founders, we used an average measure of the measures to develop a combined measure.

2.3.2.2.2 Structural holes

The presence or absence of a structural hole is a theoretical notion and can only be inferred based on the pattern of relationships associated with any node in the network (Burt 1992). Following earlier work in social networks (e.g., Burt 1997; Obstfeld 2005), we adopted the measure of constraint for structural holes as proposed by Burt (1992). Burt's measure of constraint captures the degree to which a node is dependent on the nodes it is connected to in the network and thereby the access to unique resources. The measure represents the aggregate constraint that is imposed on the project founder due to the pattern of connections that are present in its neighborhood. Higher the measure of constraint for the founder, fewer is the degree of structural holes present in the founder's neighborhood. Burt's measure of constraint can be calculated from network structure data that we obtained from the SourceForge data warehouse using the following expression (Burt 1992, p. 66),

$$C_S\left(X_i^A\right) = \sum_j (p_{ij} + \sum_k q_{ik} r_{kj})^2 \qquad 2.9$$

where p_{ij} represents the proportional investment of node *i* in node *j*, *k* is a node that is linked to both the nodes *i* and *j*, q_{ik} represents the proportional investment of node *i* in node *k*, and r_{kj} represents the proportional investment of node *k* in node *j*. To calculate the proportional investment of a node *i* in node *j* we first obtain the one-mode affiliation network of projects from the two-mode network. Proportional investment is a node *i* in node *j* is calculated as the strength of the relationship between *i* and *j* (number of developers shared between them), divided by the number of Whenever a project had more than one founder, we used an average measure to develop a composite measure for the project.

2.3.2.3 Control variables

2.3.2.3.1 Type of product

Analogous to leisure vs. utility classification used in earlier research on new product development (e.g., Golder and Tellis 1997), we also tabulated products on usage orientation, i.e., consumer vs. developer. Specifically, by relying on the classification provided in the data warehouse and by examining the project descriptions provided at the time of project registration, we coded the projects in our sample as being either consumer oriented or developer oriented. Consumer oriented projects are those that are geared towards non-technical users for e.g., mp3editor is a product in the sample that enables users to edit mp3 songs. Developer oriented products are those that are specifically geared towards technical users, for e.g., openQRM is a systems management platform that integrates with enterprise data centers.

2.3.2.3.2 Prior expertise of founders

Users who have a greater experience participating in the coordination of activities on SourceForge.net should be able to influence the future of new projects on the systems, specifically those that they initiate. We control this effect of expertise of the founders in determining the probability of the project becoming active as well as time-to-market for new products. We measured the age of the founder in the system as the elapsed time between the founder's registration on the SourceForge system as a volunteer and the exact time at which the project was initiated. We used a log-transformation of this explanatory variable (e.g., Kiefer 1988).

2.3.2.3.3 Degree of activity

Projects that experience a higher level of participation from the community might experience a shorter time for their first product release. We control for this effect by introducing a control variable in the duration stage of our model. We measure the degree of activity using the number of checkouts for the project. A checkout is when a piece of code pertaining to the source code of the project is downloaded by a community user. As the degree of activity only exists for those projects that become active, this control variable enters the model only in the duration stage of our two-stage model. The measures are summarized in Table 2.1.

Table 2.1: Descriptive statistics and bivariate correlation coefficients

N=817	Degree Centrality	Betweenness Centrality	Strutural Holes	Experience	Product Type	Development Activity	Activity	Time-to-market
Betweenness Centrality	.08**	-						
Structural Holes	-.24**	-.03	-					
Experience	.29**	.10**	-.49**	-				
Product Type	.07**	.06*	.06*	.02	-			
Development Activity	.12**	.11**	-.02	.16**	.08**	-		
Active-Inactive	.22**	.02	.06*	.16**	.08**	.10**	-	
Time-to-Market	-.16**	.01	.04	.06	.002	-..01	.77**	-
Mean	2.82**	125.52	.73	15.41	.50	7.68	.86	8.35
SD	5	1550.84	.33	2.36	.50	25.63	.35	2.59

2.3.3 Likelihood function specification and model estimation

Our model development follows standard frameworks used in econometrics textbooks (e.g., Amemiya 1985; Greene 2003). Similar model development was done in earlier applications such as Greene (1995) and Terza (1998). Based on the model specified in Equation 7, we can specify the following:

$$y_i^s \mid x_i^s, u_i \sim \phi(y_i^s \mid x_i^s, u_i) \qquad 2.10$$

where y_i^s is the observed outcome whether a project is active or not, x_i^s is the vector of explanatory variables that affect the selection process, u_i is s normally distributed error term with an expected value of 0, and $\phi(.)$ is the standard normal pdf.

Let $f(.)$ be the pdf of the outcome variable in the second stage, therefore,

$$y_i^d \mid x_i^d, \varepsilon_i, c_i \sim f(y_i^d \mid x_i^d, \varepsilon_i, c_i) \qquad 2.11$$

where y_i^d is the observed duration time, x_i^d is the vector of explanatory variables affecting the duration model, ε_i is a normally distributed error term with zero mean and variance σ^2, and c_i is the right censoring indicator for observations that get selected at stage one

The probability of projects being inactive has the following form,

$$\text{Prob}(y_i^s = 0 \mid x_i^s) = \int_{-\infty}^{\infty} \Phi\left(-\left[\alpha' x_i^s + \rho\varepsilon_i\right] / \sqrt{1-\rho^2}\right) \phi(\varepsilon_i) d\varepsilon_i \qquad 2.12$$

where $\Phi(.)$ is the standard normal cdf, $\rho\varepsilon_i = u_i$ (from equation).

For projects that become active, we observe the duration time if the project releases a working version or that they are still work-in-progress (that the observation is censored),

$$f(y_i^d, y_i^s = 1 \mid x_i^s, x_i^d, \varepsilon_i, c_i) = \int_{-\infty}^{\infty} f(y_i^d \mid x_i^d, \varepsilon_i, c_i)\,\Phi\left(\left[\alpha' x_i^s + \rho\varepsilon_i\right] / \sqrt{1-\rho^2}\right)\phi(\varepsilon_i)d\varepsilon_i \qquad 2.13$$

Combining equations (12) and (13), we can specify the joint pdf of the outcome variables in the two stage process as follows,

$$f(y_i^d, y_i^s \mid x_i^s, x_i^d) = \int_{-\infty}^{\infty} \left[\left(1-y_i^s\right) + y_i^s f(y_i^d \mid x_i^d, \varepsilon_i, c_i)\right]\Phi\left((2y_i^s-1)\left[\alpha' x_i^s + \rho\varepsilon_i\right] / \sqrt{1-\rho^2}\right)\phi(\varepsilon_i)d\varepsilon_i \qquad 2.14$$

The full log-likelihood of the model can now be written down as,

$$LL = \sum_{i=1}^{N} \log \int_{-\infty}^{\infty} \left[\left(1-y_i^s\right) + y_i^s f(y_i^d \mid x_i^d, \varepsilon_i, c_i)\right]\Phi\left((2y_i^s-1)\left[\alpha' x_i^s + \rho\varepsilon_i\right] / \sqrt{1-\rho^2}\right)\phi(\varepsilon_i)d\varepsilon_i \qquad 2.15$$

To specify a functional form for the pdf of the duration outcome variable, we first investigated the form of the hazard function using Kaplan-Meier estimates. The resulting graph indicated that the hazard function was non-monotonic, and therefore we ruled out the exponential and Weibull specifications for the hazard function. Further, we estimated three stand-alone survival regression models (stage two of our consolidated model) by specifying a log-normal, log-logistic and log-gamma forms for the duration times. A log-likelihood comparison indicated that the best fit was achieved with the log-normal specification[3]. Therefore, we choose the lognormal distribution as the appropriate specification for the duration outcome variable in the second stage of our two-stage model.

---

[3] The log-likelihood for the models were as follows: 1) -1139.675 for the log-normal, 2) -1140.694 for the log-logistic, and 3) -1153.824, -1147.684 and -1141.142 for the gamma with shape parameters 0.75, 1.25 and 4.25 respectively.

After log-transforming the duration data, the pdf of the outcome variable in the second stage, i.e., $f(y_i^d \mid x_i^d, \varepsilon_i, c_i)$ can now be represented as

$$f(y_i^d \mid x_i^d, \varepsilon_i, c_i) = \left[\sqrt{2\pi} \exp\left\{ -\left(\frac{y_i^d - \beta' x_i^d - \sigma\varepsilon_i}{\sqrt{2}\sigma^d} \right)^2 \right\} \right]^{c_i} \left[\Phi\left(-\left(y_i^d - \beta' x_i^d - \sigma\varepsilon_i \right) \right) \right]^{1-c_i} \qquad 2.16$$

The final log-likelihood expression can be obtained by substituting the expression for $f(.)$ from equation (2.16) in equation (2.15).

$$LL = \sum_{i=1}^{N} \log \int_{-\infty}^{\infty} \left[\left(1 - y_i^s\right) + y_i^s \left[\left[\sqrt{2\pi} \exp\left\{ -\left(\frac{y_i^d - \beta' x_i^d - \sigma\varepsilon_i}{\sqrt{2}\sigma^d} \right)^2 \right\} \right]^{c_i} \left[\Phi\left(-\left(y_i^d - \beta' x_i^d - \sigma\varepsilon_i \right) \right) \right]^{1-c_i} \right] \right] * \Phi\left(\frac{(2y_i^s - 1)\left[\alpha' x_i^s + \rho\varepsilon_i \right]}{\sqrt{1-\rho^2}} \right) \phi(\varepsilon_i) d\varepsilon_i \qquad 2.17$$

Our model is an implementation of a general framework for incorporating selectivity in econometric models proposed by Greene (2006).

The general class of Tobit models (e.g., Amemiya 1984), provides a good framework to understand data structures involving truncated distributions and selection models. In our context involving the two stage process model, the first stage is a selection process, i.e., whether projects become active or not and in the second stage for those projects that are active we observe time-to-market. Our context fits the Type II Tobit model involving a selection mechanism in the first stage and a regression in the second (Amemiya 1984). A number of estimators have been proposed for such models, including the widely used Heckman’s two-step estimator (e.g., Amemiya 1984). Heckman suggested the two-step estimator for such models and the details of this procedure can be found in a standard econometric textbook such as Greene (1994). The Heckman 2-step estimation involves adding the inverse-mills ratio to the regression in the second stage to account for selection bias. However, in our context, the outcome variable model in the second stage is a hazard regression and, as a result the inverse-mills ratio is not defined and cannot be used to correct for selectivity bias. We believe, our application is therefore unique in applying a more general framework suggested for incorporating selectivity in regression models (Greene 2006).

The likelihood function as presented in equation (2.12) does not have a closed form and therefore following suggestions by Greene (1994), and Terza (1998), we use the simulated maximum likelihood approach to estimate our parameters (e.g., Greene 2003; Train 2003). Our model development is unique in its application as we develop a duration model with right censoring in the presence of selection effects and estimate it using a framework that is flexible enough to accommodate a broad class of research problems.

By transforming the variables in the likelihood equation 2.16, through the substitutions,

$$\upsilon = \varepsilon / \sqrt{2}, \quad \theta = \sigma\sqrt{2}, \quad \tau = \sqrt{2}[\rho / \sqrt{1-\rho^2}], \quad and \ \gamma = [1/\sqrt{1-\rho^2}]\alpha$$

The simulated log-likelihood takes the form as expressed in equation 2.18,

$$LL_s = \sum_{i=1}^{N} \log \frac{1}{R} \sum_{r=1}^{R} \left[\left(1-y_i^s\right) + y_i^s f(y_i^d \mid x_i^d, \sigma\varepsilon_{ir}, c_i)\right] \Phi\left((2y_i^s - 1)\left[\gamma' x_i^s + \tau\varepsilon_{ir}\right]\right) \qquad 2.18$$

where ε_{ir} are a set of r random draws.

Recent developments in simulation have suggested that with large data samples computation can become very cumbersome and that the use of Halton draws results in dramatic speed gains with little effect on simulation performance (Train 2003). Therefore, we estimated our model using a sequence of Halton draws (we used 30 Halton draws) instead of the standard normal random draws. After we obtained the parameter estimates from the simulation, we calculated the variance-covariance matrix of the parameters using the delta method (e.g., Greene 2003; Oehlert 1992). The estimation procedure was carried out using the econometric software package LIMDEP.

2.4 Results

To verify whether the social network variables explain variance above and beyond those by the control variables, we estimated two models: one with only the control variables as explanatory variables and the other with both the social network and control variables as explanatory variables. A likelihood ratio test showed that the model with the social network variables outperformed the model with only the control variables ($\chi^2_{10} = 179.2$, $p < .01$). In Table 2.2 we present the results for the selection and the duration parts of the hypothesized model. The column labeled *Selection* model represents the effects of network structure and control variables on the likelihood of projects becoming active, and the column labeled *Duration model* represents the effects of network structure and control variables on time-to-market.

Table 2.2: Effect of network structure on the two stages of product development

Variable Category	Variable Name			Selection Model	Duration Model
	Constant			.75** (.38)	12.12*** (1.10)
Network Variables	Network Embeddedness	Degree Centrality (D)		.24*** (.05)	- .42*** (.07)
		Betweenness Centrality (B)		.21 (.19)	.12 (.11)
	Structural Holes	Burt's Constraint (H)		.49*** (.23)	- 2.69*** (.70)
	D X H			.50*** (.09)	- .07 .31
	B X H			.01* (.007)	- .03*** (.006)
Control Variables	Experience			.07*** (.02)	- .26*** (.06)
	Product Type			.16 (.12)	- .96*** (.36)
	Activity Degree			NA	- .20*** (.003)
	Sigma			NA	4.07*** (.20)
	Selectivity Parameter			.09*** (.04)	
	Heterogeneity Parameter			.28 (.26)	

*** - $p < .01$
** - $p < .05$
* - $p < .1$

2.4.1 Effect of embeddedness

From Table 2.2, column labeled '*selection model*', degree centrality has a positive effect on the likelihood of a project becoming active (b = .25, p <.01), and the effect of betweenness centrality is statistically not significant. From Table 2.2, column labeled '*duration model*', degree centrality reduces the time-to-market for new open source projects (b = - .42, p < .01), and the effect of betweenness centrality is statistically not significant. These findings lend partial support to hypotheses H1a and H1b that network embeddedness will have a positive effect on the probability of the project becoming active and will reduce the time-to-market. The results imply that when the project founders have greater embeddedness, as measured through degree centrality, i.e., they were involved in the coordination and organization of a number of open source projects at the time of founding of a project the project has a higher probability of becoming active and is likely to lead to a shorter time-to-market.

2.4.2 Effect of structural holes

From Table 2.2, column labeled '*selection model'*, the lack of structural holes in the project founder's network neighborhood has a statistically significant positive effect on the likelihood of the project becoming active (b =.50, p < .01) and from Table 2.2, column labeled '*duration model*', the lack of structural holes in the project founder's network neighborhood decreases the time-to-market, (b = -2.68, p < .01). These results lend support to hypotheses H2a and H2b respectively. The lack of structural holes, measured through the constraint on the founders' location in the network, measures the degree to which there are alternative connecting roles the founder can play in the network neighborhood. A higher measure of constraint implies a smaller degree of structural holes, thus implying fewer alternative opportunities for the founder. Therefore our results imply that when a founder is forced to focus his/her resources on the new initiative, these projects tent to benefit from such attention from the founder.

2.4.3 Interaction effect between embeddedness and structural holes

Hypothesis H3 suggests that fewer the structural holes, higher is the positive effect of network embeddedness on the probability of a project becoming active (H3a). We find the interaction effects to be statistically significant ($b = .51$, $p < .01$ and $b =.01$, $p < .01$ respectively) and therefore we can conclude that H3 is supported. The results suggest that there is a synergistic effect between network embeddedness and structural holes on the probability of a project becoming active, such that the projects whose founders were not only involved in the coordination of more number of projects, but had fewer alternative opportunities in the network, have a higher probability of becoming active than those projects whose founders had more opportunities in the network.

In the duration stage of our model, we find that the interaction between degree centrality and structural holes does not have a statistically significant effect and that the interaction between betweenness centrality and structural holes has a statistically significant and negative effect on time-to-market ($b = - .25$, $p < .01$). This result lends partial support to hypothesis H3b, which states that fewer the structural holes, higher is the negative effect of embeddedness on time-to-market. Thus, our findings partially support the hypothesized effect, between founders' embeddedness and structural holes, on time-to-market.

2.4.4 Effects of control variables

Our results indicate that when the founders are more experienced there is a positive effect on the likelihood of the project becoming active ($b = .07$, $p < .01$), and projects that were customer oriented (i.e., non technical in usage context) had a greater likelihood of becoming active ($b = .13$, $p < .01$). The experience of the founders and degree of development activity reduce time-to-market as one would expect ($b = -.26$, $p < .01$ and $b = -.20$, $p < .01$ respectively). We also find that products that are customer-oriented also have a lower time-to-market than products that are purely technical in their usage context ($b = -.95$, $p < .01$).

2.5 Discussion

With success stories such as Linux operating systems and Firefox Internet browser, the open source software movement represents a viable alternative to traditional firm-based models for software development. In this model of software development, a complex web of interrelationships between volunteers facilitates the flow of skills and information needed to develop technically sophisticated and commercially viable software (e.g., Grewal et al. 2006; von Hippel and von Krogh 2003). Theoretically, in this research, we build on an emerging stream of research on open source software (e.g., Shah 2006; von

Krogh and von Hippel 2006) along with research in sociology on social capital (e.g., Gulati and Gargiulo 1999; Lin 2001; Uzzi 1997) and social networks (e.g., Burt 1997; Freeman 1979; Obstfeld 2005) and research in organizational theory on organizational founding (e.g., Carroll and Delacroix 1982; Delacroix and Carroll 1983; Swaminathan 1996). The core argument stemming from the amalgamation of these diverse yet related research streams is that projects that are born in certain network positions (as accessed through embeddedness and structural holes) would be more likely to succeed. With data from 817 projects and results from a Type II Tobit model, we find that projects born in appropriate network positions not only are more likely to see start of development activity, but are also likely to have a shorter time-to-market. This dual effect of network position at birth on project activity and time-to-market lends support to our theoretical conjectures concerning the criticality of network position of the founder of a new project.

To capture the new product development process in the open source environment, we conceptualized the process as consisting of two stages: (1) transition from idea to developmental activity and (2) maturing to marketable products given developmental activity. In building our theory we recognized the seemingly contradictory notions of *coordinating actions* and *generating ideas* as being critical in determining the value of the network structure that surrounds a new project (e.g., Obstfeld 2005). Specifically, we proposed that a central position (as indicated by measures of embeddedness – coordinating actions) of the founder and opportunity available to the founder to broker relationships among other developers (as indicated by the value of structural holes – generating ideas) have additive (as seen by the main effects) and multiplicative (as shown by the interaction terms) effects on the two stages of the new product development process.

The results provide support for this dual effect such that embeddedness and structural holes have additive as well as multiplicative effects on the probability of new projects becoming active and on the time-to-first release for new projects that become active. Our results imply that embeddedness of the project founders plays a key role in open source projects becoming active and achieving shorter time-to-market. However, our results also indicate that if there are too many options available to the founders in their network neighborhood, it does not bode well for the new projects that they initiate. The open source movement is known for its free spirited participants and continuous innovation (Raymond 1999). However, the social network also seems to impose constraints on the entrepreneurial spirit of volunteers through the demands of coordination of existing projects.

2.5.1 Theoretical implications

Our research contributes to the theoretical development of the notion of social capital by considering the joint role of embeddedness and structural holes, both important notions of social capital, in innovation coordination (e.g., Burt 1997). There is considerable debate in the social networks literature on the tradeoffs between dense and sparse networks with regards to knowledge related activities (Obstfeld 2005), and the verdict on this debate has been described as still being mixed (Rodan and Galunic 2004). We contribute to the theory in this domain by specifying the nature of network effects (characterized through embeddedness and structural holes) at a micro-level (at the node level) and thereby lay the foundations for a richer understanding of the dynamics involved in innovation processes in the presence of social networks. The ubiquitous role of embeddedness in social network processes and in information and resource flows in the network is receiving considerable importance in networks literature (e.g., Kilduff et al. 2006a). Our study enriches current theoretical understanding of such effects.

2.5.2 Managerial implications

The second essay contributes substantively by studying the open source product development process and by providing meaningful insights on the factors that impact important outcomes concerning the transition of projects from ideas to active initiatives and the subsequent time to release.

The open-source model of community driven product development has given rise to a number of technologically sophisticated and robust products that are gaining mainstream acceptance. The benefits of the open source ideology are more obvious with respect to information products for reasons primarily laid out in research on technology standards and lock-ins (e.g., Economides and Katsamakas 2006; Farrell and Saloner 1985). Opening up the source code provides customers better control over their technology investments and also provides them an opportunity to participate in the product design and development process. Research on lead-user innovation provides evidence on the benefits of involving and co-operating with customers in such hybrid initiatives (e.g., Lilien, Morrison, Searls, Sonnack, and von Hippel 2002). Our view of the innovation coordination through social networks is substantiated by the unique nature of information product development through online collaboration. The availability of data on such coordination activities and the subsequent product development metrics allowed us to develop a systems approach to innovation.

Specifically, increasing number of firms such as IBM and HP are showing interest in launching hybrid projects involving the open-source community. Our findings provide valuable insights concerning key individuals (or positions) in the network that should be coordinating such development initiatives at different stages of the product development process. Firms can identify individuals with high embeddedness and appropriate constraint in the network and using this information can seek their help to coordinate the product development activity. Such strategies increase the likelihood of projects moving from ideas to developmental versions that eventually result in technologically sophisticated and commercially viable products.

2.5.3 Future research

Due to the transient nature of project-developer relationships, coordination of open source projects in virtual communities is a dynamic process (e.g., von Krogh and von Hippel 2006). This dynamic nature of network structure also provides an opportunity to study the processes that determine the evolution of networks over time. Theoretical developments in stochastic actor-oriented models provide valuable frameworks to investigate the longitudinal nature of social networks in the coordination of open source projects (e.g., Holland and Leinhardt 1977; Snijders 1996; Snijders 2005).

The strategic interest of large firms such as IBM, HP and others in the open source product development model also provides an opportunity to understand the dynamics of competition and cooperation in this high-technology domain. Hybrid models of innovation, such as those being experimented on by IBM (e.g., Bosio 2004; Bosio 2005) provide an opportunity to understand the intricacies of innovation governance in the open source context.

The extension of the “openness” notion and the involvement of connected users in the innovation process to non-digital products is unexplored territory and therefore offer exciting opportunities for future research. The popularity of various social networking sites such as facebook.com, orkut.com, and video based Youtube.com (bought by Google for 1.65 Billion USD), the seemingly ubiquitous influence of Wikipedia, and epinions.com is an indication that this new revolution will play a major role in shaping the interaction between firms and consumers. The extension of the “openness” ideology to non-information product markets is not as straightforward and requires further research in diverse fields such as intellectual property management, patents, revenue sharing and law, among others (Lessig 2001).

The study of social networks in the open source realm can also provide an opportunity to understand aspects of team design and coordination in the new product development process (e.g., Van den Bulte and Moenaert 1998). The maturity of the Internet as a communication and commercial medium has facilitated the birth and evolution of virtual consumer communities that have begun to play an important role in the success of new product launches. A deep understanding, of how such communities are structured and how that impacts the flow of information and resources can be developed by subjecting such developments to a social network analysis. Alternative forms of social capital, such as those driven by resources and relational strength, might also provide useful perspectives on organizational aspects of networks (e.g., Lin 2001).

2.5.4 Conclusion

The ideology of involving users (including customers) in the process of product design and development is moving beyond the software product world – the launch of Linux-driven hardware phones, network switches and routers are a few examples of new ways of integrating the ideology into product development practices (Fink 2003). Large firms have begun to experiment with hybrid models of innovation governance by launching some of their new product development initiatives into the community. With industry giants such as IBM and HP reengineering their product market strategy to accommodate the open source community and with thousands of products being developed by such communities it is important to understand the factors that influence the success of product development in the open source world. Our research provides critical insights into ambient conditions at the inception of new open source projects that are critical for initial and eventual success of the projects. As this economically critical, technologically advanced, and networked open source environment gains in prominence, the social network perspective would go a long way in explaining the working of this environment.

Chapter 3

EVOLUTIONARY DYNAMICS OF OPEN SOURCE PROJECTS: A STUDY OF A LARGE FIRM SPONSORED OPEN SOURCE PROJECTS

3.1 Introduction

The software industry has evolved to play a crucial role in today's high-technology society. The industry has achieved tremendous growth in the past few years as cumulative sales of packaged software products have totaled 211 billion USD in 2006 (SIIA Report 2006). Recently, community driven open source product development, i.e., the development of software products by users (who are volunteers), has gathered steam in this industry and has become a viable alternative business model to the traditional firm-driven product development model (e.g., Cloer 2005; Fink 2003; Pallatto and Spooner 2005; Weiss 2004). Products such as the Linux operating system, developed by the open source community, have become strong alternatives to traditional products developed by a single firm. Industry sources estimate that Linux server shipments alone will account for 30% of the server market in 2008, that translates to about 10 billion USD (e.g., McMillan 2004). Venture capital flows into open source ventures have risen to about 400 million USD in 2005 (Lacy 2005). The growing importance of community driven innovation has generated interest among business and academic practitioners alike (Lerner and Tirole 2005; von Hippel and von Krogh 2003). Consecutively, researchers in diverse fields have investigated various aspects of the community driven model, such as the effects of network embeddedness on open source project success (e.g., Grewal, Lilien, and Mallapragada 2006), standards driven competition (e.g., Economides and Katsamakas 2006), the structure of complex product design (e.g., MacCormack, Rusnak, and Baldwin 2006), and motivation and governance issues (e.g., Lakhani and Wolf 2003; Shah 2006), among others.

As a result of the success and potential of open source software products, prominent firms such as IBM, Sun Microsystems, and HP are showing interest, and are participating in, open source technology initiatives. Specifically, these firms have donated patents to software foundations (Bosio 2005), formed partnerships with open source firms (Galli 2005), and have started collaborative projects that involve the open source community. These firms are sharing the source code of software products with the community and developing new products, products that require know-how that is not available within their organizational boundary (Fink 2003). They are utilizing the technical know-how of the community, thus shortening their own product development cycle and reducing product development costs.

From an organizational perspective, such as that of IBM, it is important to effectively manage software product development in such open source initiatives. For this purpose Grewal et al. (2006), use a social network perspective to represent the relationships among projects and developers and to show how project location in this social space (of developers and projects) determines the success of the projects. The results from their cross-sectional study show that location in the social space is related to the success of the projects. What this finding means for firms such as IBM is that effective management of the social space of developers and projects is likely to drive the success of their open source initiatives. However, further meaningful analysis of new product management in the open source environment has to be dynamic (and not static) as projects in the open source world experience incessant changes to their ties with other projects and as a result, the attractiveness of the social space location. Examining such dynamics is the primary objective of this research.

The dynamics of network structure among open source projects is important to study because project managers and developers make decisions regarding participation in the community and this affects the relationships among projects. Their decisions impact how open source projects mature and perform over time. The incessant changes occurring to the project networks affect the availability of developer resources to the projects and therefore affect the performance of open source projects.

To understand such social space dynamics and project success, we rely on literature on social capital (e.g., Granovetter 1985; Lin 2001), and empirical studies that draw relationships between social capital and economic outcomes (e.g., Gulati and Gargiulo 1999). Specifically, we use the two notions of embeddedness and structural holes that have been used extensively in social capital research (Wasserman and Faust 1999) to capture important characteristics of network driven social capital (e.g., Burt 1997; Obstfeld 2005). These two notions imbue network actors with two different forms of social capital, namely the ability to provide efficient access in the case of embeddedness and the ability to provide access to unique resources for structural holes (e.g., Obstfeld 2005). In a connected network of open source projects, embeddedness captures the extent to which a project is intertwined in a web of relationships in the network (e.g., Grewal et al. 2006). Structural holes capture the degree to which there is opportunity for a project to broker the relationships between other open source projects (e.g., Burt 1997; Obstfeld 2005). Using these ideas of embeddedness and structural holes, we study how social capital within open source network of developers and projects affects network formation, growth and success of projects.

To understand the dynamics and implications of open source project network change we collect longitudinal network data concerning the change in network ties for 43 open source projects over a one year period. These projects are being sponsored by IBM and listed on SourceForge.net, a thriving ecosystem of open source initiatives with over 140,000 products and over 1.47 million users (e.g., von Hippel and von Krogh 2003). We propose a stochastic network evolution model (e.g., Snijders 1996), in which we model the two interdependent change processes, that of the changing network structure and project outcomes. We then model the dynamics behind the value a project will derive from 1) occupying a certain location in a given network structure and 2) achieving a certain level of success measure and estimate the model using a method of moments approach.

Briefly, we find that the network of open source projects is more likely to change when such a change 1) decreases, for a given project, the number of direct ties to other projects, 2) increases the number of connecting paths between other projects on which any given project lies, and 3) decreases a project's ability to broker tie among other projects. With respect to the success of open source projects, our results show that 1) projects experience higher success as the number of direct ties with other projects increases in the network and 2) the rate of success depends on level of community interest in the project. Implications for a firm such as IBM are that higher success in the community can be achieved by increasing interest about the project in the open source community and ensuring that the project shares ties with a large number of other projects. Overall the results seem to indicate that increasing the number of direct links to other projects lowers the likelihood of network change and therefore lowers rate of reorganization of network tie. Therefore, firms face a trade-off between achieving a desired level of success and keeping the network structure open to change so that there is a continuous flow of new information and resources vital for growth and survival.

The remainder of the chapter is organized in the following manner. We begin by providing information on 1) the importance of social capital in the formation and growth of networks, 2) the network evolution model, and 3) the effects of embeddedness and structural holes on the value associated with a particular network structure. We then elucidate the research methodology where we describe the data collection procedure, the measures, and the model estimation and validation framework. Next, we present our results and conclude by discussing the implications of this research for theory and practice. In the final section we discuss how the third chapter enriches the theoretical understanding concerning the dynamic effects of social capital in open source project networks. Methodologically, we contribute by estimating a stochastic network model that applies new developments in network modeling to understand new product innovation in the community driven open source context. Substantively, our results provide meaningful insights for managers and firms interested in participating in the open source community model on the importance of managing collaboration in community driven innovation.

3.2 Conceptual background

3.2.1 The context of collaborative innovation

The open source community has developed a number of successful and robust technology products such as the Linux operating system, MySql database, Apache web server, and the Mozilla web browser, among others. The unique collaboration between firms such as IBM and Sun Microsystems, and the open source community blends the organizational capabilities of these firms with the free-market spirit of the open source community. In the recent past, large firms, most notably IBM, have taken a greater interest in the open source movement and have adopted a strategy of active participation in the open source movement (e.g., Galli 2005; Marson 2004; Weiss 2004). Specifically, IBM has donated patents to the open source community, provided resources for new open source projects, and announced the portability of open source applications to its legacy platforms (e.g., Bosio 2005; Pallatto and Spooner 2005). As IBM and other firms adopt a service oriented architecture (SOA) with a greater focus on “configuring products”, the importance of product development by collaborative communities is likely to rise (e.g., Mann 2007).

The collaborative product development model being adopted by IBM and other firms involves sharing of source code of some of their software products with a networked community of open source volunteers. As this collaborative model gains prominence, it becomes crucial for these firms to develop the know-how required to manage product development so as to achieve their objectives. For this purpose, Grewal et al. (2006), utilize a social network approach to represent the relationships between open source projects and developers and to show how project location plays an important role in determining the success of open source projects. Similarly, other researchers have used a social network approach to represent the network structure between open source projects and developers (e.g., Xu, Christley, and Madey 2006). Although these cross-sectional studies provide valuable insights about open source collaboration through a network-centric approach, there are a number of questions concerning evolution and growth of coordination structures among open source projects that still need to be addressed. Longitudinal studies that account for changes occurring to project networks and performance can address the gap in this stream of research.

It is important to study the dynamic processes driving changes to project networks and project performance as it will allow us to develop richer insights concerning evolution and growth of governance in the open source community. Insights obtained from a dynamic network analysis of the open source community will subsequently facilitate better innovation management and new product success for firms interested in open source community participation.

3.2.2 Social capital effects on open source network dynamics

In the community model of software development, the associations shared by developers through their memberships in projects lead to informal collaborative structures that help coordinate product development processes (e.g., Shah 2006). In a cross-sectional study, Grewal et al. (2006) proposed that managing the location of projects in a social space filled with open source initiatives and developers is crucial to the success of these projects. Specifically, they relied on social capital theory (e.g., Burt 1992; Lin 2001; Uzzi 1997) to propose that network embeddedness has an effect on the technical and commercial success of open source projects. The unique approach of conceptualizing the ties among open source projects and developers as affiliation networks facilitated the application of theoretical concepts in social capital literature to the open source phenomenon. Other scholars have identified the importance of social structures that exist in the open source community and have used a sociological approach to develop an understanding of motivation, participation (e.g., Bagozzi and Dholakia 2006; Shah 2006) and structure of complex product design (e.g., MacCormack et al. 2006) in the open source product development context.

Open source projects benefit from their *location* in the network. Location refers to the specific place or position in the network in which the project is situated and is identified when there is information about all the connections that the project has with its network neighbors. Projects are connected to each other through the developers that they share with other projects in the network. The location of a project provides it with access to developer resources and knowledge that resides in other projects directly connected to or reachable through the project's location. Thus, as the location of the project changes due to formation of new connections and/or dissolution of existing ones, the project is exposed to different network configurations and thus varying bundles of resources over time. As the availability of resources changes over time, the open source projects experience variation in performance related outcomes. For example, an open source project that gains greater visibility in the network when a well connected developer joins the team might consequently experience a greater number of downloads. Moreover, the increasing popularity of the project as it garners more downloads also attracts more individuals to the project thus fueling a change to its network structure. Thus, the process of network change affects and is affected by a change in the outcomes associated with the projects. Clearly, these two change processes, that of the network connections and that of project outcomes influence each other thereby continuously altering the resources available to open source projects and the subsequent project performance.

Open source projects rely on the participation of volunteers who contribute code, debug software, report bugs, write patches, and discuss usage, among other activities (Torvalds and Diamond 2001). Project managers (often the project founders) play the gatekeeping role and decide which users get access to the source code of the project. Whenever a developer joins a project, the project gains access to new regions of the open source project network through the ties of the new developer. Similarly, when a developer leaves a project, the project loses access to parts of the network that it is uniquely connected to, through that particular developer. Moreover, when a project gains popularity in the community through its success it also garners the attention of users who would like to join the project's development team, thus affecting the project's network structure.

Before we develop a model to describe the evolution of an open source project's network connections and the substantive implications of such change we provide an illustrative example for the purpose of elucidating the phenomenon of network change. The example highlights some important issues in dynamic social networks and serves as a motivation for our model development. Let's suppose, for illustration, that Spamkill is an open source project whose developers also participate in two other projects – Deskpro and Screenpro at a certain moment in time *t1*. In this example, Actpro and Mailpro are two other projects in the community that share a tie between them. A *tie* or *link* between two projects indicates that they share at least one common developer as a team member. The figure 3.1 is a visual representation of the network of connections between the five projects at time *t1*. The presence of a line joining two projects in the figure indicates the presence of a tie between the projects. This network of projects is represented using a symmetrical matrix, $x(t1)$, wherein the element x_{ij} is binary and $x_{ij} = 1$ indicates the presence of a tie between projects *i* and *j*, and $x_{ij} = 0$ the tie's absence. At time *t1*, Spamkill has garnered a certain number of downloads- $y_1(t1)$, the subscript 1 indicates that Spamkill is the first project in the network.

Changes that occur to a social network such as the network of open source projects are not preordained, occur without a blueprint, and can be considered random. Similarly, the changes occurring to project characteristics such as downloads are unplanned and occur randomly as a result of users, developers and testers from the community downloading the project for usage. With the passage of time, let changes occur to the network configuration $x(t1)$ and the Spamkill's characteristics $y_1(t1)$. The figure 3.1 represents the network of connections between the five projects at time *t2* in a visual form as well as a matrix form. An inspection of these two figures indicates that the pattern of relationships at time *t1* i.e., $x(t1)$ has changed to $x(t2)$ due to the new relationships formed in the time interval *t1*-*t2*. Spamkill's number of downloads at time *t2* have also changed to a new value $y_1(t2)$ in this time period.

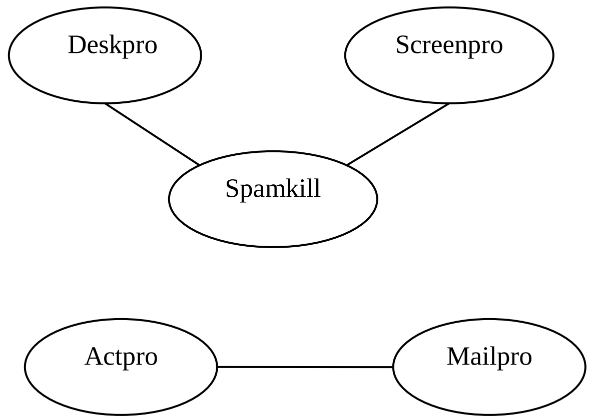

Figure 3.1: Illustrative affiliation network of open source projects at time *t1* – graphical representation.

Note:

1) Deskpro, Spamkill, Screenpro, Actpro, and Mailpro are five open source projects.
2) The line between any two projects indicates that the projects share at least one common developer between them.
3) In the above example, Spamkill shares developers with Screenpro and Deskpro. Actpro and Mailpro share at least one developer.

The above network can be represented in a matrix form as indicated below.

Project/project	Spamkill	Deskpro	Screenpro
Spamkill	0	1	1
Deskpro	1	0	0
Screenpro	1	0	0

Note:

1) As Spamkill has a tie with projects Deskpro and Screenpro, the corresponding cells in the matrix have a value of 1.
2) As Deskpro does not have tie with Screenpro, the corresponding cell has a value of zero.

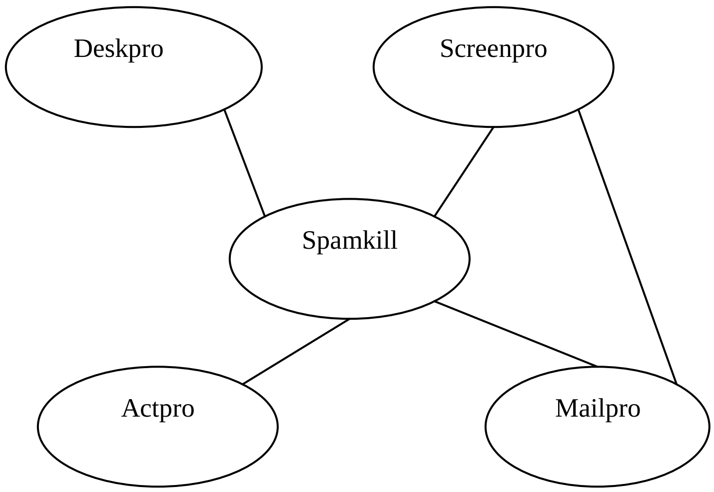

Figure 3.2: : Illustrative affiliation network of open source projects at time *t2*

Note:
1) Spamkill has two new ties – one with Actpro and one with Mailpro
2) Mailpro has a new tie with Screenpro
3) The tie between Actpro and Mailpro has been dissolved

The above network can be represented in matrix form as indicated below,

Project/project	Spamkill	Deskpro	Screenpro	Actpro	Mailpro
Spamkill	0	1	1	1	1
Deskpro	1	0	0	0	0
Screenpro	1	0	0	0	1
Actpro	1	0	0	0	0
Mailpro	1	0	1	0	0

Note:
1) There is a new entry of 1 in the cell corresponding to the relationship between Spamkill and Actpro. This indicates the new tie observed at time t2.
2) There is new entry of 1 in the cell corresponding to the relationship between Spamkill and Mailpro. This indicates the new tie observed at time t2.
3) There is a new entry of 1 in the cell corresponding to the relationship between Screenpro and Mailpro. This indicates the new tie observed at time t2.
4) There is a new entry of 0 in the cell corresponding to the relationship between Actpro and Mailpro. This indicates that the tie has been dissolved.

A comparison of figures 3.1 and 3.2 reveals that the network of open source projects has changed due to three new ties being formed and one tie being dissolved during the time interval *t1-t2*. The project Spamkill has two new ties, one each with Actpro and Mailpro; Screenpro has a new tie with Mailpro, and finally, Actpro's tie with Mailpro has been severed. The changes to ties among projects have occurred as a result of new developers being shared across projects or a shared developer leaving one or more of the projects.

In our illustrative example, two changes have occurred to Spamkill's connections in the time interval *t1-t2* – it formed new ties with projects Actpro and Mailpro, and Spamkill also experienced a change to its number of downloads. The rate at which this change occurred is calculated as number of changes/time elapsed between changes, and could be thought of as the rate at which the changes occur to that particular project. Certain project characteristics influence the rate at which such changes occur. Similarly, the rate at which changes occur to the number of downloads is calculated as the number of changes/time elapsed between changes and is a function of project characteristics.

Of all the possible changes that have could have occurred to the connections among projects in our illustrative example, only a few have occurred. For example, in the time period between *t1* and *t2*, Spamkill experienced changes to its ties while Deskpro did not. Our example highlights certain issues that are important in dynamic project networks - 1) Why is it that only some projects experience changes to their network connections? 2) Does a project's location at *t1* affect future changes in its ties? 3) How does a project's success change affect its network change and vice versa. Answers to these questions will provide insights on how the project network will evolve and grow and thereby enable us to better manage product development in the open source community. For this purpose, we conceptualize a value function that is dependent on the network configuration and use it to specify a probability model (in a multinomial logistic regression framework) to explain the odds of a change occurring to a tie between projects. The core principle underlying the value framework is that those changes to a project's ties will be more likely that endow the project with a higher network value from the configuration that it is located in. The dependent variable is the probability of whether a change will occur to the current network tie between two projects. The value function is formulated as an additive function of project characteristics that are network dependent (e.g., embeddedness) and we estimate the weights of this function in our data analysis.

Our objective in the remainder of this section is to develop a model that describes the interdependent changes that occur to the network connections and outcomes of open source projects. For this purpose, we discuss some key features of our model that describes the process of open source project change, and then develop our model.

3.2.3 A dynamic model for open source project networks

In the open source community, the participation of developers in a project leaves a permanent impact on the project through the changes that these individuals make to source code, the messages they post, and the bugs they report, among other activities. As these individuals leave and join projects, they create and destroy ties across projects, and leave the projects with a mark of their participation. Therefore, the structure of connections among projects is a net result of all the past activities of the shared developers across these projects. Due to this, at any given point in time, the state of the network is a result of the sum of changes that have been occurring to the project network in its past.

Other transient network contexts exist such as the structure of ties in an email exchange network between acquaintances where the network is driven by random events that are not captured by the current state. In our data context, as the current state fully encapsulates the past information we use a markov process to represent the state transitions of the joint stochastic process. The markovian assumption has been used in earlier studies (e.g., Snijders, Steglich, and Schweinberger 2005b) to model longitudinal network change, and has been found to represent, quite accurately, the transitions occurring to social networks.

However, we only observe the changes occurring to the network structure and to project outcomes at discrete points in time due to the nature of data aggregation, and in reality the changes occur (unobserved) in continuous time. Therefore, we model the change process as a continuous time stochastic process such that in any infinitesimal interval of time Δt only one change (or event) occurs either to the network structure or to a project outcome. Our model development follows ideas developed by Snijders (1996), Snijders (2001) and Snijders (2005).

The changes occurring to the network structure and the project outcomes are observed only at discrete points in time and therefore, we allow for the possibility that the rate of change can vary across these time periods. However, as the changes occurring to the network and to the projects are unobserved between observations, a non-constant rate function would not be feasible in our context. Therefore, we assume that the rate of change remains constant within these time periods and accordingly we model the number of changes, or events, occurring to the network and the project outcomes using a homogenous Poisson process. The homogenous Poisson representation implies that the rate of the change process is constant within a time period and therefore, the time between changes follows an exponential distribution. As we model multiple time periods, we specify a base rate parameter for each of the time periods that absorbs the period dependant eccentricity in the rate. The memorylessness of the Poisson assumption facilitates the markov transitions occurring to the joint stochastic process of network change and project outcomes change. We describe the various terms used in our model and explain how they are operationalized in our model context in Table 3.1. The table can be used as a notational reference for the model development presented in the next section.

Table 3.1: Description of model terms

Term	Description	Observed/Estimated	Operational definition
$X_n(t)$	Stochastic matrix (nxn) representing ties between projects	NA	NA
$x_n(t1) = x(t1)$	Random realization of $X_n(t)$ at time t1.	Observed	Symmetric matrix representing relationships between projects (e.g., x(t1) in our example)
$Y_n(t)$	Stochastic vector representing outcomes of n project at time t	NA	NA
$y_n(t1) = y(t1)$	Random realization of $Y_n(t)$ At time t1	Observed	The variable *y* in the example – number of downloads.
Z(t)	Stochastic variable that jointly represents X and Y	NA	NA
ρ^m	Rate of poisson process used to model network change in period m	Estimated	Number of tie changes/ length of time period
ω^m	Rate of poisson process used to model project outcome change in period m	Estimated	Number of outcome changes/length of time period
V_h	Project covariate h	Observed	e.g., type of open source license that affects rate of change
$x_{ij}(t + \Delta t)$	Network configuration at time t+ Δt , obtained from x(t) after the tie between i and j changes.	Observed	Symmetric matrix representing relationships between projects (e.g., x(t2) in our example)
$f_i(x(t))$	Value derived by project *i* by virtue of its location in network configuration x(t)	Estimated	Additive function of social network effects and covariates
$g_i(y(t))$	Value derived by project *i* by achieving a certain outcome y(t)	Estimated	Additive function of social network effects and covariates
s_p	Network statistic p	Observed	e.g., embeddedness – calculated from network data
u_j	Stochastic component of the new value derived from network when the tie between *i* and *j* changes	NA	Follows a Type I extreme value distribution.

In our illustrative example, Spamkill experienced two changes to its connections in the time interval *(t1-t2)*. The rate at which such changes occur to connections of projects in a network is the rate parameter of the Poisson distribution that we use to model the network changes. The changes to the connections of a project *i* in our model occur with a rate parameter ρ_i (i.e., ρ_i is the mean of the Poisson distribution), that is modeled using equation 3.1,

$$\rho_i(\alpha, v) = \rho^m \exp\left(\sum_{h=1}^{H} \alpha_h * v_{hi}\right) \tag{3.1}$$

where ρ^m is the base rate of change in time period *m*, and α_h is the effect of project covariate V_h on the rate of change of connections being experienced by project i. The specificity associated with the time period is absorbed by the base period rate ρ^m and this is allowed to be constant within the time period *m*. Equation 3.1 tells us that the rate at which a project will experience changes to its network connections is a function of its characteristics (e.g., type of open source license).

In the illustrative example, Spamkill's downloads changed from $y_1(t1)$ to $y_1(t2)$ – implying that it experienced $y_1(t2)$ number of downloads during the time interval *(t1-t2)*. The rate at which such changes occur to the projects' outcomes is the rate parameter of the Poisson distribution that is used to model the project outcome changes. Therefore, the changes that occur to a project's outcome variable follow a Poisson process with a rate parameter ω_i, that is modeled using equation 3.2,

$$\omega_i(\psi, v) = \omega^m \exp\left(\sum_{h=1}^{H} \psi_h * v_{hi}\right) \tag{3.2}$$

where ω^m is the base rate of change in time period *m* and ψ_h is the effect of project covariate V_h on the rate of change occurring to the outcome variable. Equation 3.2 tells us that the rate at which an outcome variable of an open source project changes is a function of its characteristics (e.g., type of open source license).

In our illustrative example, four network tie changes occurred out of all possible tie changes between the five projects. We propose that this selectivity in the transitions of network connections is due to the differential preference attached to certain locations in the network (e.g., Faust 1997). Alternatively, it can be said that every location in the network has an intrinsic value attached to it due to the pattern of ties associated with that location. Clearly, as the location of an open source project changes over time, the amount and quality of resources that are available to the project change and thereby affect the outcomes associated with the project. Spamkill gained access to new regions of the network through the new ties it developed to Actpro and Mailpro. Moreover, the projects achieve a certain level of performance on outcomes such as number of downloads and thereby attract ties from other projects. The formation of new ties results in changes to the network configuration in which the projects are located. Thus, the two processes of network change and change in project outcomes feed into each other.

Let 1) $x_n(t)$, a nxn matrix represent the network configuration that *n* open source projects are located in at time *t* and 2) $y_n(t)$, a nx1 vector denote the outcome vector associated with the *n* projects at time *t* (a more general representation would be to use a nxm vector for m outcomes). Say $z(t)=(x_n(t), y_n(t))^4$ represent the random realizations of a continuous time stochastic process, say $Z(t)=(X_n(t), Y_n(t))$ at time *t*. As every open source project accrues benefits by occupying a location in a given network structure, let $f_i(x(t))$ represent the value obtained by the open source project *i* by being located in the network configuration *x* at time *t* and $g_i(t)$ be the value accrued by the project by achieving an outcome $y_i(t)$, say for example number of downloads.

The value function $f_i(.)$ can be thought of as the benefit gained by project *i* by being located in a particular network configuration $x(t)$, and $g_i(.)$ as the value accrued by achieving an outcome $y_i(t)$. Referring to our illustrative example, Spamkill gained certain value $f_1(x(.))$ by being located in a specific network configuration *x(t1)* at time *t1* and *x(t2)* at time *t2*. Similarly, it gained a specific value $g_i(y_1(.))$ by achieving an outcome $y_1(t1)$ at time *t1* and $y_1(t2)$ at time *t2*. We will specify the functional form of these two value functions at a later stage in this section.

As the resources accessible on the open source project network are scarce, locations that endow projects with the highest network value should be more valuable. Similarly, the outcome that endows the project with the highest value in the community should be more valuable. As developers leave and join projects, the network structure changes continuously, thereby altering the value of the network location for a particular project. As illustrated in the example, the ties between two open source projects change when either a current shared developer decides to leave one or both of the projects (so that the tie is

---

[4] For notational simplicity we use $x(t)$ for $x_n(t)$ and *y(t)* for $y_n(t)$ in the remainder of the essay.

severed or broken), or when a developer who belongs to one of those projects joins the other project (so that a new tie is formed). $x_{ij}(t+\Delta t)$ is the network configuration obtained by starting from the current configuration $x(t)$ after the tie between projects *i* and *j* changes in time Δt. If the tie to project *j* is changed the value of the new network configuration will be,

$$v_j = f_i(x_{ij}(t+\Delta t)) + u_j \qquad 3.3$$

where u_j is the unexplained stochastic variance in the value associated with the new configuration $x_{ij}(t+\Delta t)$.

We specify the total value associated with a particular network configuration as an additive function of network capital components. Therefore, the value function $f_i(.)$ is modeled as function comprising of network structural effects (such as embeddedness and structural holes), and project level covariates (e.g., type of license) for the open source project. Therefore, the deterministic part of the value function can be written as,

$$f_i(\beta, x, \gamma, v) = \sum_{p=1}^{P}(\beta_p * s_{pi}(x)) + \sum_{h=1}^{H}\gamma_h v_{hi} \qquad 3.4$$

where β_p represent the effect of network capital statistic s_p, where s_p is a function of network data (such as embeddedness), on the network value function and γ_h represents the effect of the project covariate V_h on the value function. Similarly for the project outcome change, as an open source project accrues a certain amount of value by achieving a certain degree of success, we model the value function for the change in actor dependent variable as a function of network effects and project level covariates as,

$$g_i(\kappa, x, \omega, v) = \sum_{p=1}^{P} \kappa_p s_{pi}(x) + \sum_h \omega_h v_{hi} \qquad 3.5$$

where κ_p represents the effect of network capital statistic s_p on the value function for the outcome variable and ω_h represents the effect of the project covariate V_h on the value function. Referring to our example, this is the value gained by Spamkill by achieving a certain level of outcome $y_1(t1)$ and by being located in a network configuration *x(t1).*

In summary, we suggest that changes to network occur due to the fact that 1) some locations are more valuable due to the network value attached to them and project managers prefer their projects to be in such locations thus causing the network configuration to change and 2) projects that achieve better outcomes tend to attract ties from other projects thus altering the network configuration. We have so far presented an explanation as to what drives the changes in open source project networks and the projected related outcomes.

In our illustrative example, changes occurred to only a few possible connections among projects. The model we have developed so far addressed the timing of the changes and what drives the changes. Now, we provide the explanation as to which of the possible changes will occur based on a probability model. Referring to our example, this will explain as to why the ties associated with Spamkill changed in the time interval *(t1-t2)*, and not the other ties. A new network configuration with respect to project *i*'s location can be obtained through a change in tie with any of the *n* projects that constitute the open source community.

The value of the new network configuration has a random component u_j associated with it. The tie change that results in a network configuration with the *highest* value for a project undergoing the change will be preferred over the set of all possible changes and therefore a Type I extreme value distribution is an appropriate choice for the random component u_j. Therefore, let the probability that the tie between projects *i* and *j* will change in a small interval of time Δt, given the starting network configuration $x(t)$ and outcome variable $y_i(t)$, be expressed by equation 3.6,

$$\Pr\left(x_{ij}(t+\Delta t) \mid x(t), y_i(t)\right) = \frac{\exp\left(f_i\left(x_{ij}(t+\Delta t), y_i(t)\right)\right)}{\sum_{h=1, h\neq i}^{G} \exp\left(f_i\left(x_{ih}(t+\Delta t), y_i(t)\right)\right)} \qquad 3.6$$

Equation 3.6 lays out the probability with which the link between projects *i* and *j* will change, given that project *i* was located in the network configuration *x(t)* and had experienced an outcome $y_i(t)$. In our illustrative example, this probability was realized through the changes that we actually observed to the ties associated with Spamkill in the time interval *(t1-t2)*.

Subsequently, the probability that an open source project will experience a change δ in its outcome variable in a small interval of time Δt starting with $y_i(t)$ at time *t* can be expressed by equation 3.7,

$$\Pr\left(y_i(t+\Delta t)[\delta] \mid x(t), y_i(t)\right) = \frac{\exp\left(g_i\left(x(t), y_i(t+\Delta t)[\delta]\right)\right)}{\sum_{\tau\in\{-1,0,1\}} \exp\left(g_i\left(x(t), y_i(t+\Delta t)[\tau]\right)\right)} \qquad 3.7$$

We have so far explained how the ties between open source projects change and how projects experience changes to performance outcomes. The complex changes that cumulate between two longitudinal observations can be built up using Δ steps in small intervals of time that follow a random process. The stochastic process that we have described can be fully specified by an initial state and the transition probability matrix, the elements of which are given by equation 3.8,

$$q(z(t);z(t+\Delta t))=\begin{cases}\rho_i * \Pr(x_{ij}(t+\Delta t)\,|\,x(t),y_i(t)) & if\ \ x(t)\,changes \quad (a)\\ \omega_i * \Pr(y_i(t+\Delta t)[\delta]\,|\,x(t),y_i(t)) & if\ \ y_i(t)\,changes \quad (b)\\ -\sum_i\left\{\sum_{j\neq i}q(z;(x_{ij}(t+\Delta t),y_i(t)))+\sum_{\delta\in\{-1,1\}}q(z;(x(t),y_i(\delta)))\right\} & if\ \ no\,change \quad (c)\end{cases} \tag{3.8}$$

where $z(t)$ is the initial state and $z(t+\Delta t)$is the state achieved after a delta change has occurred either to the network structure or to the outcome variable. If $z(t+\Delta t)$is reached through a change in the network configuration alone, the transition probability is given by (a) in equation 3.8. If $z(t+\Delta t)$is reached through a change in the project outcome variable alone, the transition probability is given by (b) in equation 3.8. If $z(t+\Delta t)$is identical to its previous state $z(t)$, the transition probability is given by (c) in equation 3.8. Thus, the network and the outcome variable change processes co-influence each other through the transition probability matrix.

The stochastic state, jointly characterized by the network configuration and project outcomes, can experience a transition after every infinitesimal time interval. The possible state transition graph is presented in figure 3.3. Figure 3.3 can be understood by relating it to equation 3.8. A transition occurs from state *A* to the state *B,* with the probability indicated by line 1 in equation 3.8, when a change occurs to the network. A transition occurs from state *A* to state *C*, with the probability indicated by line 2 in equation 3.8, when a change occurs to the project outcome. Finally, no state change occurs, with the probability indicated by line 3 in equation 3.8, when no change occurs to the network or to the project outcomes.

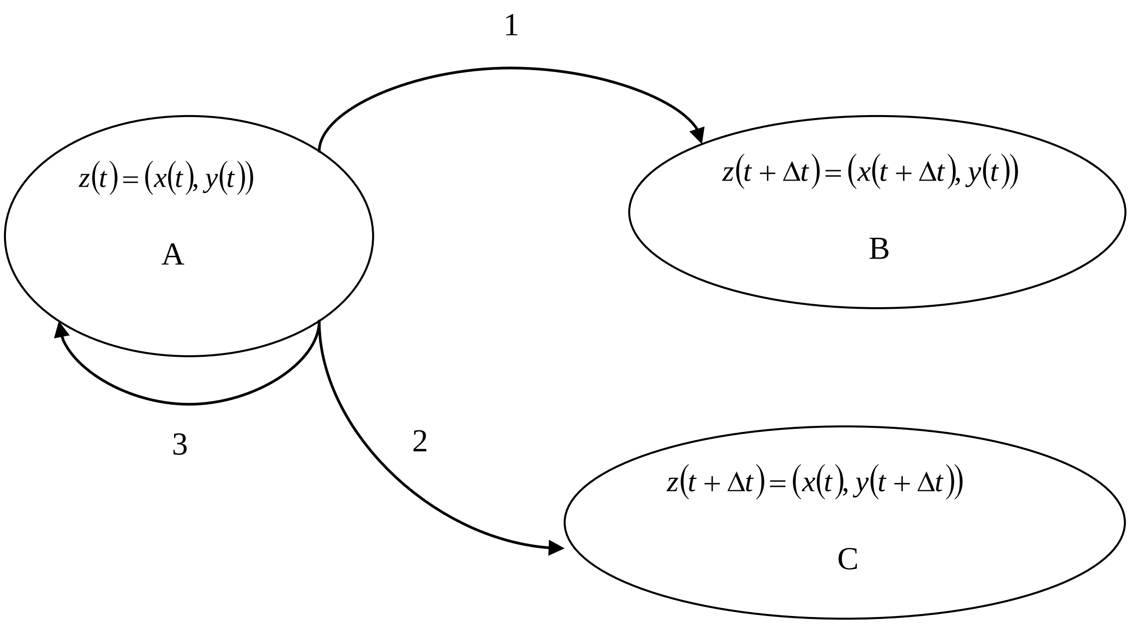

Figure 3.3: Possible state transitions for the stochastic process *z(t)* during Δt

Note:

1) The transition from state A to state B occurs if a change occurs to any tie of any project.
2) The transition from state A to state C occurs if a change occurs to the project outcome of any project.
3) If no changes occur to the ties or to the project outcomes, no state change occurs.

3.2.4 Hypotheses

Our hypotheses primarily concern the effects of social capital on the value associated with network structure. This implies that we specifically propose the expected sign of network statistic effects in equation 3.4 of our model.

Among various organizational issues studied in network theory, the two issues of *coordinating action* and *initiating ideas* are crucial to understand the network structural effects on organizational outcomes (e.g., Obstfeld 2005). Networks in which the nodes are densely connected to each other are efficient in solving the *action* problem as the high degree of interconnectivity allows a smooth dissemination of information and flow of resources, thereby facilitating coordination (Granovetter 1985). Networks that are sparsely connected have been found to be effective in solving the *idea* problem as they present a conducive opportunity for generation of new ideas through the isolation and presence of unique sets of resources in the network (e.g., Obstfeld 2005). The concept of embeddedness effectively captures the extent to which a node is entrenched or connected in the network (Kilduff, Tsai, and Hanke 2006), while the notion of structural holes captures the degree to which sparseness is present around any given node (e.g., Burt 1997; Obstfeld 2005). Theoretical research in social networks has identified many important characteristics of network driven social capital (Wasserman and Faust 1999) chief among them being the notions of embeddedness and structural holes (e.g., Burt 1992; Granovetter 1985; Uzzi 1997). Embeddedness impacts the coordinative processes in networks, and structural holes impacts the entrepreneurial opportunities that are available to actors in the network (Burt 1997; Obstfeld 2005).

3.2.4.1 Effect of embeddedness

Embeddedness has been found to have an impact on coordination processes that facilitate information flow in the network (e.g., Kilduff et al. 2006; Uzzi 1999). Embeddedness of an open source project enables greater access and reach to resources of value in the network, thus providing the open source projects a rich environment to nurture and grow. The impact of embeddedness has been found to be quite ubiquitous on network structure and formation in various organizational contexts (e.g., Kilduff et al. 2006). Projects endowed with a greater level of embeddedness by virtue of their location in the local network neighborhood would be in a unique position to be the conduits for information flows and collaborative resources on the network. A higher degree of connectivity eases the flow of information and resources over the network, facilitates transfer of resources and capital, and thus, solves the *action problem* (e.g., Obstfeld 2005). Embeddedness has been found to have a beneficial effect on organizational outcomes associated with bio-technology start-ups and ability to raise financial capital, thereby indicating that the degree of connectivity in the network is a suitable proxy to social capital (e.g., Powell, Koput, and Smith-Doerr 1996; Uzzi 1996). Therefore, when a new link is being formed or an existing link is being removed with respect to a project, the change will affect the project's location in the network structure. Therefore, we propose that

H1: The network value that an open source project will derive from a change in one of its links will be greater when the change results in an increase in embeddedness for the project.

3.2.4.2 Effect of structural holes

Structural holes are those locations in the network that facilitate the formation of network tie among other actors (e.g., Burt 1992; Burt 1997). Projects that occupy these locations are better located to be able to exploit this opportunity by brokering the tie among other projects in the neighborhood. In an open source project network, projects that occupy locations that have a higher measure of structural holes are less constrained by their existing network tie and therefore have the ability to bridge more gaps. These projects can act as facilitators in the formation of indirect links between other projects in the network. Burt (1992) proposed the concept of structural holes in networks and that their presence in a network indicates the availability of entrepreneurial opportunities in the network. The presence of structural holes also facilitates gaps in the network and thereby solves the *idea problem* (Obstfeld 2005). The presence of structural holes in the network neighborhood of a node indicates the sparseness of tie, thereby providing an opportunity for an entity to extract rent by stepping in to fill gaps in the network. The missing tie allows new ideas to originate and blossom in the network without the interference from denser neighborhoods in the network.

Projects can extract value by brokering tie among other projects and therefore such locations can be conceptualized as being ideal to facilitate the exploitation of entrepreneurial opportunities in the network (e.g., Burt 1992; Gould and Fernandez 1989). Projects can only extract appropriate value from their unique location by actually brokering relationships between other projects, and by doing so, they form new links. Formation of new links imposes a greater constraint on a project's location as its ability to broker new relationships is reduced as a result of the burden imposed by new ties. However, note that a structural hole's potential can only be exploited by actually leveraging that location and brokering an indirect tie between other projects. Therefore, a decrease in degree of opportunity will result in the projects gaining higher benefit from the network structure. Following this line of reasoning, we propose that

H2: The network value that an open source project will derive from a change in one of its links will be greater when the change results in a decrease in structural holes for the project.

3.3 Method

3.3.1 Data collection

We collected the data required for our study from the SourceForge.net website. SourceForge.net is the Internet's largest website for collaborative software development. The website facilitates the hosting and development of open source projects and currently hosts over 140,000 open source projects and 1.47 million registered users. Therefore, the site offers a rich context for studying product development occurring in the open source community. The data source has been recommended by scholars for conducting empirical research on the open source community (e.g., von Hippel and von Krogh 2003; von Krogh and von Hippel 2006) and has been used in empirical research projects (e.g., Xu et al. 2006).

The participation of IBM in the open source community has endowed the open source movement with much needed legitimacy. IBM currently sponsors a number of open source projects and engages in development and infrastructure support for these projects. These projects are instances of a hybrid model being experimented by IBM, a harbinger of such hybrid collaborative models in this community. An investigation of these projects will provide richer insights on the nature of firm participation in the open source community. We therefore chose a sample of 43 open source projects that are being sponsored by IBM. From a methodological standpoint, projects being championed by a single firm provide a natural control for variance that might arise in coordination structures of open source projects due to variation in motivation, expectation and efficiency of multiple firms on this front. These projects are listed under the open source section of IBM's website. The corporate website provides a brief description about the projects and a link to the project's homepage either on IBM's own servers or on sites such as SourceForge.net. Some project descriptions also contain information about the identity of IBM employees who work on these projects. We tabulated the names of 43 projects from IBM's webpage that were being hosted on SourceForge.net. After obtaining this information, we identified these projects on the SourceForge.net website and collected information required for our data analysis from the SourceForge.net data warehouse (Madey 2005).

After identifying the projects on the SourceForge.net website, our objective was to develop the network of associations between open source projects and developers required for our analysis. For this purpose, we first identified all the developers who were working on these 43 projects as of November 2003, the first month from which the SourceForge.net data is available. We queried the SourceForge.net data warehouse[5] and extracted the

---

[5] Refer Appendix A for a description of the data collection procedure.

identities of all the open source projects that these developers were working on, apart from the first set of 43 projects. After obtaining the final set of 274 open source projects, we constructed social networks using network relationship information at five time points, with a three-month interval between observations beginning November 2004, i.e., we collected information on the network structure for the months of February 2005, May 2005, August 2005 and November 2005.

Thus, with this procedure, our network finally represented 274 open source projects and 1725 developers over the five observation periods. Such a snowballing procedure has been used in social network methodology (e.g., Grewal et al. 2006) and have been reviewed earlier in literature (Laumann, Marsden, and Prensky 1989). Using this relational information between open source projects and developers we constructed social networks representing the pattern of association between open source projects and developers.

For illustrative purposes, we graphed the network of open source projects at the first and last observation moments. Figure 3.4 is the network at the first observation moment and Figure 3.5 at the last. Relating back to the example that we presented earlier, just as the illustrative network in figure 3.1 changed to figure 3.2 in the time interval t1-t2, the project network in our data sample changed from figure 3.4 to figure 3.5 in the period from november 2004 to november 2005. For instance, compared to figure 3.4 there is a new tie between projects numbered 11 and 23, 40 and 24, and 26 and 24, among others. These new links indicate that these projects now share developers. The network grew and became denser as more projects share developers.

The nodes in figures 3.4 and 3.5 represent the open source projects and the lines joining the projects indicate that the projects share at least one common developer. We also provide a brief summary of descriptive statistics of these networks and the three others in our data in Table 3.1.

Apart from collecting data on longitudinal network structure we also collected data on key variables of interest such as success, degree of community interest, and type of open source license.

Table 3.2: Descriptive statistics of the longitudinal network observations

Statistic/time	November 2004	February 2005	May 2005	August 2005	November 2005
Network density	.011	.011	.022	.022	.025
Average degree	.465	.465	.93	.93	1.07
Number of ties	10	10	20	20	23
Number of changes from previous observation	NA	2	10	0	3

Note:

1) The network changed considerably between February 2005 and May 2005 – the number of ties doubled.
2) The last row of Table 3.2 indicates the differences in overall network structure as time progresses.
3) The number 10 in the last row for May 2005 observation indicates that ten ties changed in the network during the period February 2005 and May 2005.

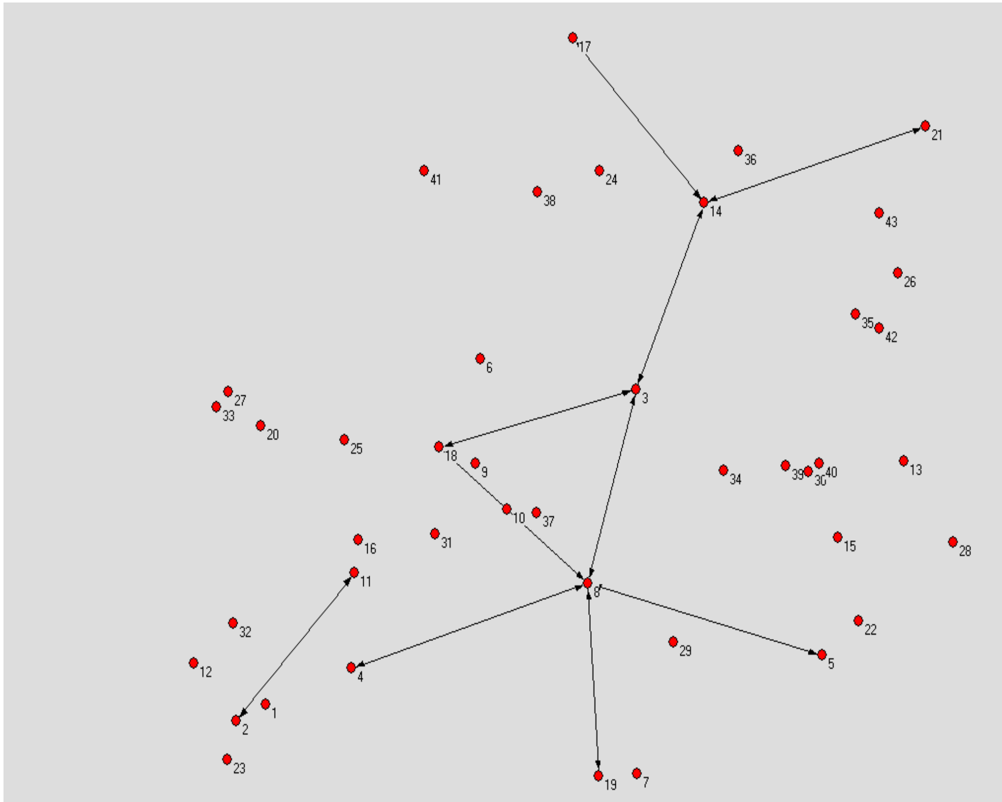

Figure 3.4: Network of open source projects in November 2004 – the first observation in our data sample.

Note:

1) The connecting lines between projects indicate that they share at least one common developer.
2) Referring to our illustrative example this figure is similar to figure 3.1 when the network is observed for the first time.

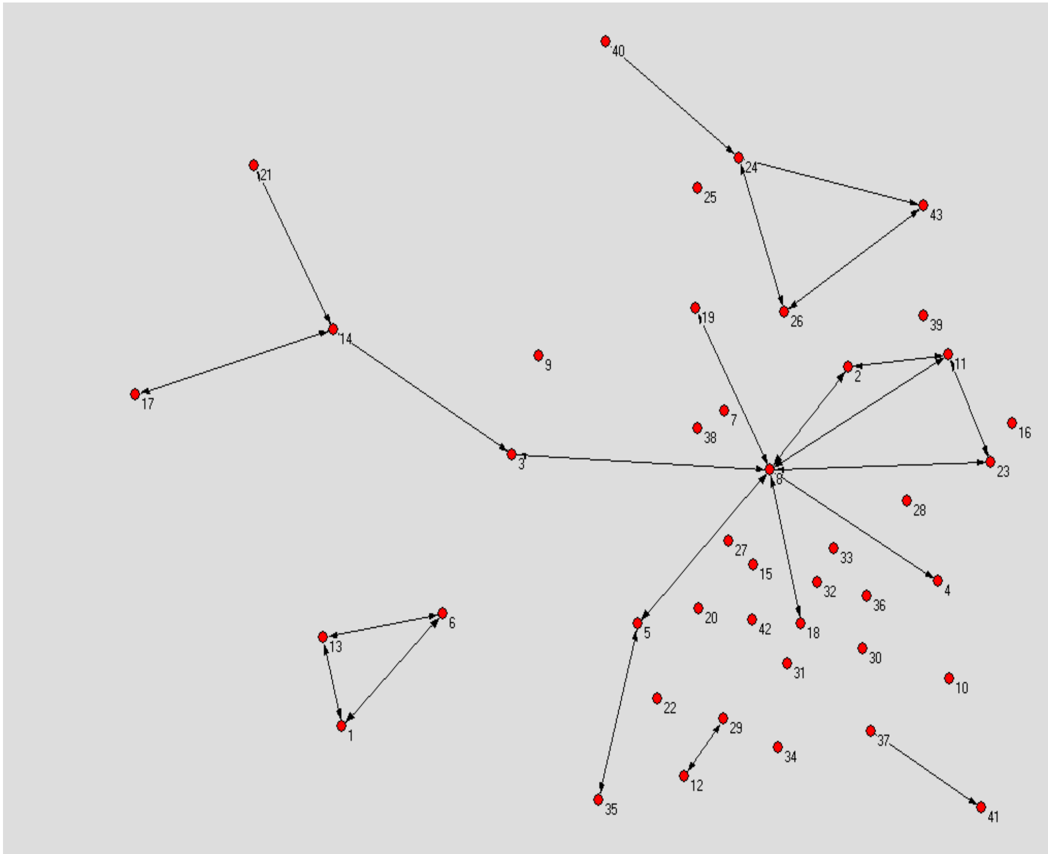

Figure 3.5: Network of open source projects in November 2005 – the final observation in our data sample.

Note:

1. The connecting lines between projects indicate that they share at least one common developer.
2. Compared to Figure 3.4 this figure has twice the number of connections and therefore twice the average density.
3. The increase in number of ties or connections indicates that more projects share developers at this point in time than in the past.
4. Referring to our illustrative example, this figure is similar to figure 3.2 in which the network is observed after changes occur to connections among projects.

3.3.2 Measures

3.3.2.1 Network embeddedness

Consistent with extant research on social capital measures, we operationalize network embeddedness using two measures of centrality: degree and betweenness centrality (e.g., Faust 1997).

Degree Centrality: We use degree centrality as one measure of network embeddedness, following similar conceptualizations in earlier research (e.g., Gulati and Gargiulo 1999). Conceptually, *degree centrality* captures the extant to which the project is entrenched in the local network neighborhood, and is an indication of degree of connectivity in the first-order network of the project. Essentially it measures the number of other projects that the particular project is associated with in the network. We calculate degree centrality using equation 3.9,

$$C_D(X_i^A) = X_{ii}^A \qquad 3.8$$

where X^A is the affiliation matrix of open source projects and X_{ii}^A is the *i*th diagonal element of the affiliation matrix.

Betweenness centrality: We use betweenness centrality (e.g., Faust 1997; Freeman 1977) as the second measure of embeddedness. Conceptually, *betweenness centrality* measures the extant to which the project founder acts as the connecting link between other projects on the network. It captures the degree of connectivity in the extended network (indirect tie) of the founder. It can be calculated by counting the number of times the project founder lies on the shortest path between any two projects on the network. We calculate it using equation 3.10,

$$C_B\left(X_i^A\right) = \sum_{j<k} \frac{g_{jk}(i)}{g_{jk}} \qquad 3.9$$

where $g_{jk}(i)$ is the number of geodesic paths (shortest paths) between any two projects j and k that pass through the project *i*, and g_{jk} is the number of all possible geodesic paths between projects *j* and *k*.

3.3.2.2 Structural holes

The presence or absence of a structural hole is a theoretical notion and can only be inferred based on the pattern of relationships associated with any node in the network (Burt 1992). Following earlier work in social networks (e.g., Burt 1997; Obstfeld 2005), we adopted the measure of constraint for structural holes as proposed by Burt (1992). Burt's measure of constraint captures the degree to which a project is dependent on the projects it is connected to in the network and thereby the access to unique resources. The measure represents the aggregate constraint that is imposed on the project due to the pattern of ties that are present in its neighborhood. Higher the measure of constraint for a project, fewer is the degree of structural holes present in its neighborhood. Burt's measure of constraint can be calculated from network structure data that we obtained from the SourceForge data warehouse using equation 3.11 (Burt 1992, p. 66),

$$C_S\left(X_i^A\right) = \sum_j (p_{ij} + \sum_k q_{ik} r_{kj})^2 \qquad 3.10$$

where p_{ij} represents the proportional investment of project *i* in project *j*, *k* is a project that is linked to both the projects *i* and *j*, q_{ik} represents the proportional investment of project *i* in node *k*, and r_{kj} represents the proportional investment of project *k* in project *j*. We used the network analysis software package UCINET (Borgatti et al. 2002) for calculating these measures.

3.3.2.3 Non-network variables

Success: Keeping in tradition with earlier research that used number of downloads as a measure of market success for software products (e.g., Chandrashekaran, Mehta, Chandrashekaran, and Grewal 1999) we use the number of times an open source project has been downloaded as a measure of success. This variable is the project outcome variable that we use in our model development section.

Community Interest: We use the number of page views for an open source project as a measure of community interest in the open source project. The measure indicates the visibility garnered by the open source project in the community.

Type of License: Open source projects are released under various software licensing schema. The most popular and widely accepted open source license is the General Public License (GPL). However, there are other popular licensing schemes which are less restrictive than the GPL, such as LGPL (Lesser General Public License), IBMCPL, and BSD, among others. We used a binary coding scheme to indicate whether a project was released under the GPL or an alternate license.

The two project level covariates, i.e., type of open source license and degree of community interest are likely to affect the value associated with a particular network configuration and the project outcomes. We incorporate them as control variables in our model. Moreover, these covariates are also likely to affect ρ_i, the rate of network structure change and ω_i, the rate of project outcome (downloads) change. The rate of change in a project's network connections (i.e., ρ_i) is an indicator of the level of action in the reorganization of ties of that project. Some projects experience a continuous churn in their team size that often leads to depletion of resources available to them, thus resulting in their abandonment (e.g., Shah 2006).

Open source projects are released under various licensing schemes, the most influential and widely regarded among these being the General Public License (GPL). There are other licensing schemes such as the Lesser GPL (LGPL) and IBMCPL (IBM Common Public License issued by IBM), among others. The GPL offers the least amount of flexibility in allowing the open source products releases under its scheme to interface with other proprietary products. Therefore, users of open source products that are released under the GPL banner are often constrained in the application of the product. However, for open source projects to be downloaded and used more frequently they need to be able to provide greater flexibility to programmers in terms of usage rights and interoperability with other software products, including proprietary products. Therefore, the projects that have adopted a GPL license should experience less frequent changes to their network structure and a lower rate of success due to their lower applicability.

The open source development process has been likened to a "bazaar", or an unorganized market where it is said that "given enough eyeballs, all bugs are shallow" (e.g., Raymond 1999). Projects that tend to attract greater interest from the open source community experience a higher number of project page views, that reflects higher level of interest in the community (e.g., Grewal et al. 2006). Thus, the rate of change in network structure and the rate of success will be dependent on the degree of community interest in the open source project such that projects with a higher degree of community interest will experience a faster rate of change in network structure and number of downloads.

3.3.3 Model estimation

We use a method of moments (MoM) approach to estimate our model parameters (McFadden 1989; Snijders, Steglich, and Schweinberger 2005a). In general, variants of the method of moments estimation have been used quite extensively in empirical studies (e.g., Chintagunta 1992; Dutta, Narasimhan, and Rajiv 1999). The estimates are often globally unique and under regularity conditions have been shown to be efficient (Lee 1992; McFadden 1989). Our data context consists of network observations and outcome variable observations collected progressively at five points in time. These observations are outcomes of the random process, $Z = (X, \mathrm{N})$, the probability distribution of which, conditional on the initial observations, is determined by the model parameters. The MoM uses suitable network statistics that are functions of the observations, and are informative about model parameters. The parameters are determined so that the expected values of the statistics, given the parameter values, are equal to the observed values. For a general model with data Z and a parameter set θ, the method of moments estimator for a statistic s(Z) is defined as that parameter value $\hat{\theta}$ for which the expected value and the observed value of s(Z) are the same. The moment equation in our context is solved using a stochastic iteration algorithm that is a variation of the Robbins and Monro (1951) algorithm. The algorithm consists of three stages; in the first stage the sensitivity of the parameters to the initial values is developed. The second stage iterations are used to obtain parameter estimates and finally in the third stage iterations the parameter estimates are held constant to develop estimates for the standard error that can be used for a convergence check and significance testing. The estimation was carried out using the network analysis package SIENA, a part of the StOCNET network analysis software (Huisman and van Duijn 2003). For a more detailed description of the estimation methodology refer to Appendix C.

3.4 Results

Convergence diagnostics for the parameter estimates are presented in Table 3.3. These estimates are based on iterations in the third phase of the estimation procedure in which the parameter values are kept constant and compared against those obtained from simulated configurations of the network. The convergence check essentially compares the simulated values of the network statistics against the observed values. The t-statistic is calculated as the *average/standard deviation* of the deviations between the simulated and the observed values. The t-statistics can be thought of as a Type I error, as one would like the deviations to be ideally close to zero as possible. An absolute t-statistic value less that 0.1 indicates good convergence of the parameter estimate. Our results indicate that convergence is present across the parameter estimates and all the t-statistics have an absolute value less than 0.05. Good convergence of parameter estimates across the board indicates that our model is not misspecified, thus lending confidence to our results.

The final estimates of the model are in Table 3.4. In *H1* we had proposed that the probability of network change would be higher for open source projects when such a change would lead to higher embeddedness. From Table 3.4, we find partial support for this hypothesis. We find that one for component of embeddedness, i.e., betweenness centrality, the effect on the network value is positive and statistically significant ($b = 12.386$, $p < .05$). This finding provides partial support to our hypothesis *H1* that proposed that the value of a new network configuration would be higher when the new configuration results in higher betweenness centrality. The finding implies that a project is more likely to form new relationships with other projects when the changed location in the network increases their ability to be the conduits for the flow of information in the network. For the second component of embeddedness, i.e., degree centrality we find that the effect is negative and statistically significant ($b = -14.026$, $p < .05$), implying that network value is higher when a project has fewer tie after the network change. This finding did not support our hypothesis.

We also find the effect of Burt's constraint (the measure of structural holes) on the objective function to be positive and statistically significant ($b = 3.512$, $p < .05$). This finding lends support to *H2* that suggested that the probability of change in network structure would be higher for open source projects when such a change would lead to a lower degree of opportunity in the surrounding network. This finding implies that open source projects prefer those locations in the network neighborhood that are not structural holes. The implication is that open source projects prefer to be in locations with a higher structural holes measure in the initial stages of their growth and transition to positions with a lower measure of structural holes in the process gaining value by brokering relationships in the network (e.g., Gould and Fernandez 1989).

Apart from the hypothesized theoretical effects, we find that only the number of direct ties of an open source project affects whether an open source project will experience a download or not. This finding tells us that the value that projects gain by achieving a certain number of downloads is only from being connected to a number of other projects, and not downloads per se.

With respect to the control variables, the results indicate that open source projects released through the General Public License (GPL) have a lower rate of change in success than projects which are released under alternative licenses ($b = -1.077$, $p < .05$). This finding is intuitive as the GPL although more widely used is a restrictive license and therefore success which is an indicator of adoption is lower for those projects which are released under the GPL. We also find that the degree of community interest, an indicator of visibility of the open source project has a positive effect on the rate of network change. This result implies that open source projects that attract a higher level of interest among the community experience a faster pace in network changes than those projects that lack visibility in the community.

In Table 3.4, parameters numbered 1- 4 represent the network rate parameters for the four time periods in our observation window. They represent the pace at which changes occurred to network structure during that time period. Apart from the third time period, in which there were no changes in the network structure (the parameter was fixed to reflect this), we observe that the rate of network change is significant in the first (b = -

.387, $p < .05$), second ($b = -.97$, $p < .05$) and, fourth periods ($b = -1.037$, $p < .05$). The network rate decreases gradually as time progresses (except for the third period where the network does not change from the second period), indicating that the reorganization of network structure slows down after a while suggesting evolution towards an equilibrium configuration. This finding is consistent with earlier observations in the open source world that suggest the evolution of a stable network structure over time, a scenario that is characterized by non-changing relationships among projects and developers.

Table 3.3: Convergence statistics of the parameter estimates

Effect	Parameter Estimate	Standard Deviation	t-statistic*
Network rate (Period 1)	0.016	2.26	0.007
Network rate (Period 2)	0.213	8.925	0.024
Network rate (Period 3)	0.046	3.908	0.012
Network rate (Period 4)	0.057	3.481	0.016
Degree Centrality	0.051	3.469	0.015
Betweenness Centrality	-0.012	3.202	-0.004
Structural Holes	0.009	2.234	0.004
Similarity in Type of License	0.147	5.99	0.024
Similarity in Community Interest	0.727	27.099	0.027
Type of License on Network rate	0.474	23.297	0.02
Community Interest on Network Rate	-0.058	4.35	-0.013
Degree Effect	0.011	0.974	0.011
Intercept	0.226	7.254	0.031
Similarity Effect	-0.143	8.017	-0.018
Type of License on rate of Success	-0.038	8.497	-0.004
Type of License on rate of Success	-0.002	4.172	0
Community Interest on rate of Success	-0.034	2.887	-0.012

* The t-statistic is like a Type I error. A value lower than .10 indicates convergence in the parameter estimate. Note that there is convergence across the board indicating that model misspecification is not a major concern.

Table 3.4: Parameter estimates from the Method of Moments estimation

Parameter Number	Parameter Name/Effect	Parameter Estimate
1	Network rate (Period 1)	- 0.387** (0.260)
2	Network rate (Period 2)	- 0.970** (0.746)
3	Network rate (Period 3)	- 7.665 (fixed)
4	Network rate (Period 4)	- 1.037*** (0.211)
5	Degree Centrality	- 14.026** (7.518)
6	Betweenness Centrality	12.386** (8.556)
7	Structural Holes	3.512** (1.597)
8	Similarity in Type of License	0.296 (1.054)
9	Similarity in Community Interest	2.681 (3.627)
10	Type of License on Network rate	1.882 (2.128)
11	Community Interest on Network rate	3.564*** (1.278)
12	Intercept	0.420 (0.461)
13	Similarity Effect	-2.762 (3.394)
14	Degree Effect	-0.474* (0.369)
15	Effect of Type of License	1.093 (0.958)
16	Effect of Community Interest	-0.383 (0.864)
17	Type of license on rate of success	-1.077** (0.574)
18	Community Interest on rate of success	0.625 (0.774)

* - $p < .10$
** - $p < .05$
*** - $p < .01$

3.5 Model validation

Our objective in this study was to provide substantive insights on how networks among open source projects change and what such change implies to coordination of product management and performance. However, there are certain threats to the validity of our results that arise primarily due to 1) model specification error 2) sampling frame 3) observation window and 4) measurement error. To alleviate the concerns that might arise due to these threats, we estimated a series of alternative models.

In general, the findings from these models suggested that the results obtained from our main model are quite robust and therefore the substantive implications do not change drastically. We provide a brief description of the alternative models that we estimated in table 3.5 and, in table 3.6 we provide a comparison of the direction of important theoretical effects across the various models. We contrast the parameter estimates from our base model with the alternative models by presenting them side-by-side in table 3.7, so as to highlight the fact there is a minimal impact of issues raised above on our substantive implications.

In the remainder of this section we describe in detail how each of these concerns could possibly affect our conclusions and how we address these concerns.

3.5.1 Effect of model specification error

Model specification error arises when the model is not an accurate representation of the true structure of the underlying physical phenomenon. In our context, we imposed a model structure in which we specified the effects of indirect network effects such as betweenness centrality and structural holes as social network effects. Our formulation suggests that to completely represent the effects of a network structure we need to look beyond direct effects such as degree centrality. The results from our findings would be tenuous if in reality only the direct degree effect was sufficient to describe social network effects. To address this issue, we estimated a model in which we included only the effect of degree centrality and constrained the effects of structural holes and betweenness centrality to a null value. This model tested whether indirect network effects such as betweenness and structural holes that measure closure in the network are required to provide an accurate description of the phenomenon. The superiority of the model with the network effects incorporated would establish that accounting for network effects is required to provide a better description of the phenomenon. The first row of table 3.5 provides a summary of this alternative model and the conclusion we draw from a comparison with our base model.

A generalized Neyman-Rao score test indicated that the model in which all these network effects were included outperformed the model without the effects. The value of the test-statistic approximately follows a chi-squared distribution and indicated that the model in which the effects are constrained to be null is an inferior model compared to the one in which the effects of betweenness centrality and structural holes are incorporated ($\chi_2^2 = 19.58, p = 0.0001$). In addition, we also found across the board convergence of parameters in the hypothesized model. Thus, we were able to alleviate the concerns regarding model specification to a reasonable degree.

3.5.2 Effect of sampling frame

Threats to our findings can also arise due to the nature of our sampling frame. Our data context involves longitudinal network observations of open source projects in the community. These projects differ on a number of characteristics, some of which are accounted for in the hypothesized model (e.g., type of license, and degree of interest). However, age of the projects and their popularity are two characteristics of the project that might influence the nature of network value in such a way that our findings might be affected.

To rule out the threats arising due to this, we estimated two models using a sub-sample of our data and checked the consistency of our findings. First, we estimated a model using the top 33% of the projects in the sample by age to alleviate the concerns arising due to differences in age across projects. We present the results from this analysis in column labeled Model 1 in table 3.7, where we find that the direction and magnitude of the effects of embeddedness and structural holes on network value is consistent with the results from the hypothesized model. A comparison of the parameters numbered 5, 6, and 7 under columns labeled Model A and Model 1 shows the consistency in the theoretical effects. A summary of the comparison of theoretical effects is presented in table 3.6 and a comparison of columns labeled Model A and Model 1 indicates the consistency in findings.

Second, we estimated a model using the top 33% of the projects in the sample by downloads to alleviate concerns arising due to differences in popularity across projects. We present the results from this analysis in column labeled Model 2 in Table 3.7 where we find that the effects of embeddedness and structural holes on network value are consistent with the results from the hypothesized model. A comparison of parameters numbered 5, 6, and 7 under columns labeled Model A and Model 2 in Table 3.7 shows the consistency in the theoretical effects. A summary of the comparison of important theoretical effects is also presented in Table 3.6. A comparison of columns labeled Model A and Model 2 shows that the direction of effects remain consistent across these two models.

3.5.3 Effect of observation window

Our primary interest was in estimating the effects of embeddedness and structural holes on value associated with network configuration. However, as we are dealing with longitudinal observations, our results could be affected by the length of the observation window if the dynamic network process is not yet mature. To validate our results, we estimated a model in which we eliminated the last longitudinal observation in our data. This implies that we estimated the parameters for data beginning in november 2004 and ending in august 2005. The results of this estimation are presented under column labeled Model 3 in Table 3.7. A comparison of parameters numbered 5, 6, and 7 under columns labeled Model A and Model 3 in Table 3.7 shows the consistency in the theoretical effects. These results are summarized in Table 3.6. We find that the results from this estimation are consistent with our findings from the hypothesized model thereby indicating that the dynamic process is quite mature.

3.5.4 Effect of using alternative measures

Error arising due to the measurement of network related constructs could also have an impact on the veracity of our findings. Although there are no alternative measures for many network related constructs, we used measures that were conceptually similar to ours to test for the effects of measurement error. We expected to find conceptually similar results even after we substitute our operationalization with the alternative measures in our analysis. The results for this estimation are presented under column labeled Model 4 in Table 3.5, and although two of the effects were not significant we find that the direction of all the theoretical effects is consistent with our original findings. These results are summarized under column labeled Model 4 in table 3.6. The results from this alternative model estimation strengthen our confidence in the conceptual arguments relating to the effects of embeddedness and structural holes on network value.

In summary, we estimated a series of alternative models to alleviate concerns that might arise due to our choice of data and model specification. The findings of our model validation procedure, presented in tables 3.5, 3.6 and 3.7 and summarized in this section provide a reasonable degree of evidence to believe the substantive implications of the results from our main analysis.

Table 3.5: Description of alternative models

Model Number	Description	Purpose	Conclusion
A	Effects of betweenness centrality and structural holes constrained to null values.	To verify if indirect network effects are present/alternative model specification	Hypothesized model performs better.
1	Top 33% projects by age	To verify if results vary by age	Substantive conclusions do not change.
2	Top 33% projects by success	To verify if results vary by success	Substantive conclusions do not change.
3	First four time periods	Change in effects through time	Substantive conclusions do not change
4	Alternative measure for structural holes	Measure sensitivity	Substantive conclusions do not change

Table 3.6: Comparison of theoretical effects across alternative models

Effect/Model	Model A (hypothesized model)	Model 1 (sub-sample by age)	Model 2 (sub-sample by success)	Model 3 (sub-set of time periods)	Model 4 (alternative measures)
Degree Centrality	-	-*	-*	-*	-*
Betweenness Centrality	+	+*	+*	+*	+
Structural Holes	+	+*	+*	+*	+

Note:
1) The signs indicate direction of the effects.
2) The column labeled Model A (hypothesized model) is the main model.
3) The superscripted stars indicate that we also found these effects significant
4) Model 1 – Model 4 are alternate models estimated as part of model validation

Table 3.7: Model validation results

Number	Parameter Name	Model A (hypothesized)	Model 1	Model 2	Model 3	Model 4
1	Network rate	- 0.387** (0.260)	-2.107 (3.235)	-1.889*** (0.721)	-1.650*** (0.486)	-1.527*** (0.645)
2	Network rate	- 0.970** (0.746)	- 0.666* (0.463)	- 0.607*** (0.3207)	- 3.559* (2.584)	- 3.098*** (1.370)
3	Network rate	- 7.665 (fixed)	- 1.863 (fixed)	- 2.876 (fixed)	- 4.70 (fixed)	- 2.528 (fixed)
4	Network rate	- 1.037*** (0.211)	- 5.538*** (1.476)	- 0.172*** (0.054)	NA	- 2.315*** (0.349)
5	Degree Centrality	- 14.026** (7.518)	- 9.501* (7.269)	- 9.160*** (3.511)	- 10.19*** (4.59)	- 4.34** (2.640)
6	Betweenness Centrality	12.386** (8.556)	8.491** (5.187)	11.081*** (4.577)	7.831*** (4.577)	2.856 (7.013)
7	Structural Holes	3.512** (1.597)	2.874*** (1.458)	2.113*** (0.522)	2.197*** (1.065)	1.104 (1.062)
8	Similarity in Type of License	0.296 (1.054)	1.951 (2.887)	1.933 (4.334)	-0.372 (0.958)	0.382 (0.898)
9	Similarity in Community Interest	2.681 (3.627)	1.917 (4.250)	2.008 (7.118)	2.743 (6.433)	0.044 (4.811)
10	Type of License On Network rate	1.882 (2.128)	1.152 (1.589)	0.227 (2.781)	5.90 (5.029)	3.954 (4.498)
11	Community Interest on Network rate	3.564*** (1.278)	2.252** (1.314)	2.211** (1.360)	3.261*** (1.575)	2.596** (1.715)
12	Intercept	0.420 (0.461)	0.424 (1.104)	-0.309 (0.480)	5.614* (4.314)	0.451 (0.635)
13	Similarity Effect	-2.762 (3.394)	-1.133 (4.104)	-1.362 (3.838)	-10.047 (9.223)	-3.173 (3.956)
14	Degree Effect	-0.474* (0.369)	-1.035*** (0.495)	-2.696*** (1.036)	-1.821** (1.072)	-0.579*** (0.339)
15	Effect of Type of License	1.093 (0.958)	-1.454 (1.945)	-1.476 (2.079)	13.661 (10.383)	0.828 (0.969)
16	Effect of Community Interest	-0.383 (0.864)	-0.182 (1.250)	-0.200 (1.436)	-5.467 (14.780)	-0.462 (0.993)
17	Type of license on rate of success	-1.077** (0.574)	-0.202 (1.098)	-1.274 (1.160)	-1.4260 (0.426)	-1.135* (0.872)
18	Community Interest on rate of success	0.625 (0.774)	0.432 (1.048)	0.395 (1.175)	0.295 (0.884)	0.532 (0.713)

* $p < .10$
** $p<.05$
*** $p < .01$

Model 1 – Model 4 are alternative models estimated as part of model validation

3.6 Discussion

The community based model of collaborative software product development has gained considerable momentum in the technology marketplace. Prominent firms such as IBM, Sun Microsystems, and HP are actively engaged in collaboration with the open source community and have thus provided legitimacy to the open source community effort. An overlap of resources between the community and the firms, achieved through a network of relationships among open source projects and developers, has facilitated an environment of innovation and growth. From the perspective of these firms, it is critical to manage the collaborative network process comprehensively so as to achieve the desired objectives. In this research we build on the emerging stream of research on the open source community (e.g., von Hippel and von Krogh 2003) along with research in sociology on social capital (e.g., Gulati and Gargiulo 1999; Lin 2001; Uzzi 1997) and methodological research in network dynamics (e.g., Snijders 1996) to develop rich insights on the dynamics of open source project networks.

Due to incessant change in the memberships of open source projects, the project networks experience dynamism that affects the extent and quality of network resources that are available to the projects. Using data from 43 open source projects that are being sponsored by a large firm and results from a stochastic network evolution model, we find that 1) rate of change in network structure decreases over time and 2) open source projects seek to occupy locations that increase their embeddedness in the network. To understand the evolutionary dynamics driving open source project network change, we mapped the transition of open source projects that are being sponsored by IBM. We sought to understand the effects of social capital on the probability of change in network structure over time. We relied on social capital theory to suggest that projects would prefer those locations that would increase their structural and betweenness centrality and decrease their measure of structural holes over time. Through our study, we were able to answer three important questions relating to open source project networks. We modeled the dual change processes of the network and project outcomes as Poisson processes and addressed the issue of "when" the network and the projects change. Second, we proposed that 1) locations that have greater value in the network and 2) project outcomes that provide greater value would be preferred and thus addressed as to what drives the changes occurring to the network and project outcomes. Finally, we specified "how" such changes occur by developing a probability model that determines which ties will change as time progresses.

In general the results of this study lend support to our hypotheses that social capital, characterized through the dual notions of embeddedness and structural holes, plays an important role in the evolution of network structure of open source projects. Specifically, we found that locations that allow projects to play the role of conduits (or connecting nodes) for connections among more projects (higher betweenness centrality) would be preferred over time. We also found that locations that allow projects to be more constrained (fewer structural holes) and does not allow them to form more relationships with new projects would be preferred over time. With respect to the effects of project characteristics on rate of change of network and project outcomes, we found that projects experience a faster rate of reorganization in the network when they are released under liberal licensing schemes. We also found that, projects experience a faster success rate when they garner higher interest in the community.

We also found one surprising result in our study. We expected that the locations that endowed open source projects with higher degree centrality, i.e., more number of ties to other projects would be preferred. However, the results indicated that as time progresses the value of such locations decreases. One possible explanation for this finding is that during the early stages of a project's lifecycle it is important to have ties with more projects to be able to have access to a wide range of skills in the network. However, as the project matures the focus shifts to achieving stability in the development process and a fewer number of committed resources than a higher number of shared resources. Therefore, locations that provide fewer shared resources in the network seem to have higher value for the projects.

Considered as a whole, these findings imply that social capital that arises due to network relationships plays an important role in shaping the dynamics of network evolution. The dynamic nature of social networks is not only evident from the change in the structure of relationships of open source projects, but also from the interdependency between such change in structure and the change in characteristics of open source projects. Specifically, we find that the change in structure happens in tandem with the change in project characteristics such as level of community interest and adoption in the community. Therefore, models such as the one implemented in this chapter are useful in developing a deeper understanding of innovation processes in the presence of social networks.

3.6.1 Theoretical implications

Our research contributes to theory on two main fronts. First, it contributes to social network theory by investigating the endogenous effect of social capital on the change occurring to network structure. Clearly, a given configuration of network structure that leads to the accrual of social capital to open source projects also facilitates an expectation of change in social capital that is associated with a network change. By testing a dynamic model that incorporates the effect of social capital on network change coupled with the change in actor characteristics, we enrich the current understanding on the role of social capital.

Second, we contribute towards theory in new product development by investigating the evolution of relationships between product development initiatives in the open source world. Network relationships among open source projects form the basis for self-governance and by investigating the effects of social capital on these governance structures, we lay the foundation for further research in open source product development.

Furthermore, the position taken by this chapter that economic outcomes (such as success of open source projects), are embedded in a broader social context (that of the open source collaborative community), is consistent with the theoretical position taken by economic sociology (e.g., Granovetter 1985). As the real world becomes more connected and firms increasingly compete as alliances or "networks" in the organizational landscape (e.g., Achrol 1991), it is important to alter our understanding of economic action and develop a more comprehensive understanding of this change.

3.6.2 Managerial implications

Our findings provide valuable insights concerning the formation of network relationships among open source projects through the developers who are shared across the projects. Our results indicate that the open source project networks tend towards equilibrium configurations over time and therefore, firms need to focus on managing the structure if they need to achieve specific objectives. By investing in multiple product options in the open source community, firms can pick and choose those that evolve and mature over time and discard those that fail to garner the interest of the community. As changes to network structure affect the value and quality of available resources to open source projects, firms need to continuously monitor these changes and convert the reorganization of network structure to their advantage. By targeting product characteristics such as the type of open source license and community interest in the project, firms can either speed up or slow down the rate of network change and thereby alter the growth trajectories of projects.

Through such Social Darwinism firms can alter the growth of open source projects and thus achieve their strategic objectives. Clearly, the open source ecosystem provides an experimental background for firms to try out new product technologies in a cost effective manner. Firms can also utilize this model to subsidize their investments in high-risk high-return initiatives and develop a more diverse portfolio of future products. Firms such as IBM are beginning to rely on the open source world for some of the products that are absent from their portfolio. Such a strategy allows big vendors such as IBM to work with open source software vendors such as Compiere and eCRM to develop solutions that might be missing from IBM’s product portfolio. Such a nexus with the open source community subsidizes IBM's product development costs and shortens development time (Fink 2003), thus allowing it to offer standards-complying cost effective solutions for its customers. The strategy also provides top-line growth by increasing the scope of product markets and bottom-line growth by lowering product development costs. Moreover, it also allows a large firm such as IBM to develop a portfolio of strategic options that helps diversify the risk associated with focusing on any one strategy (e.g., Amram and Kulatilaka 1999; Leslie and Michaels 1997; McGrath 1997; Taudes, Feurstein, and Mild 2000).

3.6.3 Future research

Open source products are characterized by the "release early, release often" dictum. Research can study the product-market scenarios wherein such "collective wisdom" might not be valid. A non-network approach to open source product development is also required to understand the intricacies in the development and adoption of complex technology products. As various versions of the same open source project often co-exist in the community, research on the evolution of technology trajectories of open source products can provide insights on managing the product lifecycle of such products. Open source products vary on the modularity of their design (e.g., MacCormack, Rusnak and Baldwin 2006) and this modularity changes over time. Research can study the processes that alter the product design characteristics in such dynamic contexts. Game-theoretic models that incorporate strategic tradeoffs between proprietary versus open source technology products can provide guidelines for managerial decision making in IT product adoption scenarios.

We use a homogeneous Poisson process to represent the changes occurring to the network and project outcomes within a time period that results in an exponential distribution of the time interval between events. An alternative non-homogeneous Poisson framework would allow the usage of a more general Gamma representation of the time intervals. However, this would require additional data on how the project covariates change within a time period so that the rate of change can be represented as a dynamic function of time and covariates and provides an avenue for further extensions to our model.

To effectively represent the interdependency between project network and outcome change, we have a made a choice of not to incorporate the strength of relationships between two projects in the network. This implies that we do not use the information as to how many developers are shared between projects and how this strength changes over time. Building on our current model specification, we hope to incorporate this information as a dyadic covariate and estimate the corresponding effects on network change in future research.

The extension, of the “openness” notion and the involvement of connected users in the innovation process, to product domains other than software is unexplored territory and therefore offers exciting opportunities for future research. Due to the rising popularity of social networking sites, targeted marketing to these communities is on the rise as well. Insights learnt from the open source community networks can be applied to better understand how these social communities evolve and mature over time.

3.6.4 Conclusion

Open source projects belong to a self-governing ecosystem that is dynamic. Developers join and leave projects at will and thereby create a market for the most attractive open source projects. Large firms have begun to take interest in hybrid models that involve joint collaboration between the community members and in-house employees. This interest has lead to the evolution of new models of innovation governance in which the structure of network relationships among open source projects acts as the governance mechanism through which the benefits of social capital accrue to participants in the network. Understanding the evolution of network structure would provide critical insights on how to structure best practices in resource allocation and risk diversification in the open source collaborative model. As the prominence of community-driven models of innovation increases, the very nature of economic organization in the business world is bound to change. Our effort to study the open source context is a significant step towards developing a deeper understanding of the implications of such change.

Chapter 4

CONCLUSION

My dissertation presents research on the impact of social networks on the open source community model of software product development through two connected essays. Essay 1 is a cross-sectional empirical study and investigates the additive and multiplicative effects of two important social capital concepts, embeddedness and structural holes on the success measures of open source projects. Essay 2 is a longitudinal empirical study that investigates the affects of social capital on the dynamics of open source project networks.

Essay 1 investigates the effects of embeddedness and structural holes on two very critical aspects of open source product development, whether development activity begins and how long it takes to get to a working release of the product. Innovation in this community model of software development has been producing robust and market leading products. However, apart from the very successful open source products, there are a number of products that do not transition to the maturity stage and end up either abandoned or inactive. (Indeed, most of the NPD literature is focused on the (biased) sample of successful products simply due to lack of data about failures.). Essay 1 conceptualizes the open source product development effort as consisting of two stages, initiation of development effort and the subsequent release of an official working release. Projects are born or are initiated when they get listed on the SourceForge.net site. However, not all projects that are initiated go through development and eventually release working versions of the product. Projects, for which the development efforts do begin, vary on the time they take to get to the version for working release. Essay 1 relies on literature in organizational founding, which suggests that the environment and features developed at the time of birth of organizations have lingering effects on the organizations evolution and survival; and on the literature on social networks, which emphasis the importance of network embeddedness even in the open source environments. The hypotheses are that the network structure into which a project is born will play a dominant role in determining whether development efforts will begin or not; and if they do begin, how long is the time-to-market for the development effort. The persistent effect of network structure at the time of inception is tested using data from SourceForge.net, the largest collection of open source projects on the Internet.

Essay 1 uses a Type II Tobit model that discriminates between projects in which the development effort takes off and projects in which it does not, to estimate a survival model for the time to market. A modeling framework that incorporates selectivity into non-linear models is used and the model is estimated using a simulated maximum likelihood approach. The results suggest that network structure does not have a strong effect on determining which projects will be abandoned. However, network structure plays an important role in determining time-to-market for new products. The first essay makes important contributions to theory on social capital by testing the contingent effects of embeddedness and structural holes on two important aspects of open source product development. The chapter also implements a new modeling approach for incorporating selectivity in non-linear models.

Essay 2 is an empirical study of open source projects that are being sponsored by IBM. The coordination of development activities on open source projects occurs through the social relationships among projects and developers in the open source community. Projects that are being sponsored by traditional firms are of great importance as they represent instances of new models of innovation governance being experimented in the open source context. The structure of open source project networks changes over time as 1) network locations vary on their resource value, and ties are formed or broken by projects to be situated in preferred locations and 2) projects achieve a certain level of success, and ties are formed or broken by projects to be connected to other successful projects. These two change processes (network and outcomes associated with projects), feed into each other thereby altering the dynamics of open source project networks. Essay 2 relies on social capital theory to hypothesize the effects of social capital on the dynamics of open source project networks.

Specifically, we propose that open source projects seek to belong to such a network structure that enhances their embeddedness and reduces their structural holes measure. We build a model in which we model these two change processes jointly. For this, we use a stochastic network model in which we model the two change processes as Poisson processes that are embedded in a continuous time Markov process. A state in this stochastic process is defined as being located in a particular network configuration with a certain level of outcomes for each of the projects. The changes, occurring to the network and outcomes associated with projects, occur depending on whether a change in state increases the net value attached with a particular network configuration and outcome. We estimate the model using a method of moments approach. We find partial support to our embeddedness hypothesis and full support to the structural holes hypothesis. We also find support to our hypothesis involving substantive variables, namely, type of license and degree of community interest. The second essay makes important theoretical contributions to understanding the growth of open source project networks. As the value and quality of resources available to projects changes with the changes occurring to network structure, it is important to understand why such changes occur and how such change would affect the projects. The essay also makes an important methodological contribution by implementing an emerging modeling technique to a very important substantive domain in community-driven new product development.

In general, we find that the two network notions of embeddedness and structural holes play a very important role in shaping the structure of coordination in the open source community model. The open source community seethes with an inherent anarchy and creativity that need to be understood more thoroughly and two notions of embeddedness and structural holes provide an avenue to this purpose. Embeddedness deals with "coordinating action" in this seemingly anarchic world and my thesis provides insights on the effects of embeddedness on important product development outcomes. In the second chapter, we find that the effect of embeddedness plays out through degree centrality and that the effect of the second component of embeddedness – betweenness centrality is not important. This tells us that a founder's direct links to earlier open source projects is of greater importance to new projects. In the third chapter, we find that betweenness centrality plays a more critical positive role in shaping network dynamics and that the effect of degree centrality is opposite to what we proposed. This tells us that, projects benefit a lot more by being the connecting nodes across different projects than being just connected a large number of other projects. Substantively these findings tell us that while coordination is extremely critical in the early stages of an open source project's life, and as governance structures become more stable, projects prefer to be in "hotspots" or pathways rather than just have more number of direct ties to other projects. In Chapter 2, the measure of structural holes influences the probability of a newly founded open source project becoming active positively and reduces its time-to-market. In Chapter 3, those changes to the network are more likely which allow a project to exploit or leverage its ability to broker relationships, although that reduces its measure of structural holes.

Relying on theory in social networks, the two essays of my thesis address related aspects of product development in the open source model of collaborative innovation. They highlight the importance of network relationships and how they play a role in shaping the innovation processes in community innovation. As interest on the open source movement grows among firms, public policy institutions, governments, universities and individuals, it becomes more critical to develop richer theories that elucidate various aspects of innovation governance in this domain. I believe that my dissertation is a significant step towards developing a richer description of the innovation processes in this interesting and evolving context.

The scope for research on the open source phenomenon is immense and opportunities for access to data are rising. There are other areas of research interest, including product design, team dynamics, competition, motivation, interpersonal communication, and new product diffusion, that are yet to be tackled in this domain. As interest on open source rises in the business world, it becomes highly critical to develop a comprehensive agenda for research on open source product development.

As an ardent admirer of open source software and a grudging user of propriety software, I clearly understand the limitations of the open source model. While there are quite a few usage contexts in which open source software is the dominant alternative, there are many more where a lot of work is required. Open source software has made huge leaps on the server side and has become a most preferred choice for enterprise customers. However, more work needs to be done to get a good end-user friendly desktop on the market. New initiatives by Dell, IBM and RedHat indicate that the dream might not be too far away in the future. A recent announcement from Dell that it would ship desktops and laptops loaded with Ubuntu, a popular desktop Linux distribution, is an important step in this direction. This announcement is also bound to have an impact on other computer vendors. Initiatives by firms, developing countries, non-profit foundations, and individuals are required to realize the true potential of the open source context. The world would definitely be a better place with more choice and the open source model is clearly the paradigm that can create this choice in the information technology world.

Open Source is ushering in a big change in the software world and only those firms that are open to adaptation might survive. I end with a conversation between Alice and the Red Queen, two characters from Lewis Carroll's story, "Through the Looking Glass" that highlights the truism that nothing is permanent except change,

> "Well, in our country," said Alice, still panting a little, "you'd generally get to somewhere else — if you run very fast for a long time, as we've been doing."
>
> "A slow sort of country!" said the Queen. "Now, here, you see, it takes all the running you can do, to keep in the same place. If you want to get somewhere else, you must run at least twice as fast as that!"

Bibliography

Achrol, Ravi S. (1991), "Evolution of the Marketing Organization: New Forms for Turbulent Environments," *Journal of Marketing*, 55 (October), 77-93.

---- (1997), "Changes in the Theory of Interorganizational Relations in Marketing: Toward a Network Paradigm," *Journal of the Academy of Marketing Science*, 25 (1), 56-71.

Agarwal, Rajshree and Barry L. Bayus (2002), "The Market Evolution and Sales Takeoff of Product Innovations," *Management Science*, 48 (8), 1024-41.

Ahuja, Gautam (2000a), "Collaboration Networks, Structural holes, and Innovation: A Longitudinal Study," *Administrative Science Quarterly*, 45 (3), 425-55.

---- (2000b), "The Duality of collaboration: Inducements and Opportunities in the Formation of Interfirm Linkages," *Strategic Management Journal*, 21 (3), 317-43.

MacCormack, Alan, John Rusnak, and Carliss Y. Baldwin (2006), "Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code," *Management Science*, 52 (7), 1015-30.

Amemiya, Takeshi (1984), "Tobit Models: A Survey," *Journal of Econometrics*, 24, 3-61.

---- (1985), *Advanced Econometrics*. Cambridge, MA: Harvard University Press.

Amram, M. and N. Kulatilaka (1999), *Real Options: Managing Strategic Investment in an Uncertain World*: Harvard Business School Press.

Bagozzi, Richard P. and Utpal M. Dholakia (2006), "Open Source Software User Communities: A Study of Participation in Linux User Groups," *Management Science*, 52 (7), 1099-115.

Barker, Colin (2006), "For Dell, Industry Standard now Includes Linux," *ZDNet.co.uk*, Accessed October 15, 2006, http://news.zdnet.co.uk/0,39020330,39273865,00.htm.

Batagelj, Vladimir and Andrej Mrvar (1996), "Pajek: Program for Large Network Analysis." 0.99 ed.

Baum, Joel A. C. and Christine Oliver (1991), "Institutional Linkages and Organizational Mortality," *Administrative Science Quarterly*, 36 (June), 187-218.

Bayus, Barry L. (1985), "Word of Mouth: The Indirect Effects of Marketing Efforts," *Journal of Advertising Research*, 25 (3), 31-39.

---- (1998), "An Analysis of Product Lifetimes in a Technologically Dynamic Industry," *Management Science*, 44 (6), 763-75.

Bayus, Barry L., Sanjay Jain, and Ambar G. Rao (1997), "Too Little, Too Early: Introduction Timing and New Product Performance in the Personal Digital Assistant Industry," *Journal of Marketing Research*, 34 (1), 50-63.

---- (2001), "Truth or Consequences: An Analysis of Vaporware and New Product Announcements," *Journal of Marketing Research*, 38 (1), 3-13.

Bayus, Barry L. and William P. Putsis, Jr. (1999), "Product Proliferation: An Empirical Analysis of Product Line Determinants and Market Outcomes," *Marketing Science*, 18 (2), 137-53.

Bollen, Nicolas P. B. (1999), "Real Options and Product Life Cycles," *Management Science*, 45 (5), 670-84.

Borgatti, Stephen P., Martin G. Everett, and Linton C. Freeman (2002), "Ucinet 6 for Windows: Software for Social Network Analysis." Harvard, MA: Analytic Technologies.

Bosio, Lori (2004), "IBM Contributes Database Software to the Apache Software Foundation," *IBM Press Release*, Accessed 24 October, 2006, http://www-1.ibm.com/press/PressServletForm.wss?MenuChoice=pressreleases&TemplateName=ShowPressReleaseTemplate&SelectString=t1.docunid=7254&TableName=DataheadApplicationClass&SESSIONKEY=any&WindowTitle=Press+Release&STATUS=publish.

---- (2005), "IBM Pledges 500 U.S. Patents To Open Source In Support Of Innovation And Open Standards," *IBM Press Release*, Accessed 25 October, 2006, http://www-1.ibm.com/press/PressServletForm.wss?TemplateName=ShowPressReleaseTemplate&SelectString=t1.docunid=7473.

Brown, Jacqueline Johnson and Peter H Reingen (1987), "Social Ties and Word-of-Mouth Referral Behavior," *Journal of Consumer Research*, 14 (3), 350-162.

Burt, Ronald S. (1992), *Structural Holes: The Social Structure of Competition*. Cambridge, MA: Harvard University Press.

---- (1997), "The Contingent Value of Social Capital," *Administrative Science Quarterly*, 42, 339-65.

Carroll, Glenn R. and Jacques Delacroix (1982), "Organizational Mortality in the Newspaper Industries of Argentina and Ireland: An Ecological Approach " *Administrative Science Quarterly*, 27 (2), 169-98.

Casadesus-Masanell, Ramon and Pankaj Ghemawat (2006), "Dynamic Mixed Duopoly: A Model Motivated by Linux vs. Windows," *Management Science*, 52 (7), 1072-84.

Chandrashekaran, Murali, Raj Mehta, Rajesh Chandrashekaran, and Rajdeep Grewal (1999), "Market Motives, Distinctive Capabilities, and Domestic Inertia: A Hybrid Model of Innovation Generation," *Journal of Marketing Research*, 36 (1), 95-112.

Chandy, Rajesh K. and Gerard J. Tellis (1998), "Organizing for Radical Product Innovation: The Overlooked Role of Willingness to Cannibalize," *Journal of Marketing Research*, 35 (4), 474-87.

Chintagunta, P. K. (1992), "Estimating a Multinomial Probit Model of Brand Choice Using the Method of Simulated Moments," *Marketing Science*, 11 (4), 386-407.

Cloer, Jennifer (2005), "OSDL Claims Linux Making Major Gains in $6 Trillion Global Retail Sector," Accessed, http://www.osdl.org/newsroom/press_releases/2005/2005_08_04_beaverton.html.

Cover, Robin (2005), "Open Source Development Labs (OSDL) Announces Patent Commons Project," *Coverpages.org*, Accessed October 15, 2006, http://xml.coverpages.org/ni2005-08-10-a.html.

Cyert, Richard M. and James G. March (1963), *A Behavioral Theory of the Firm*. Englewood Cliffs, NJ: Prentice Hall.

Delacroix, Jacques and Glenn R. Carroll (1983), "Organizational Foundings: An Ecological Study of the Newspaper Industries of Argentina and Ireland " *Administrative Science Quarterly* 28 (2), 274-91.

Dutta, Shantanu, Om Narasimhan, and Surendra K. Rajiv (1999), "Success in High-Technology Markets: Is Marketing Capability Critical?," *Marketing Science*, 18 (4), 547-68.

Economides, Nicholas (1996), "The Economics of Networks," *International Journal of Industrial Organization*, 14 (2), 673-99.

Economides, Nicholas and Evangelos Katsamakas (2006), "Two-Sided Competition of Proprietary vs. Open Source Technology Platforms and the Implications for the Software Industry," *Management Science*, 52 (7), 1057-71.

Eisenhardt, Kathleen M. and Tabrizi N. Behnam (1995), "Accelerating Adaptive Processes: Product Innovation in the Global Computer Industry," *Administrative Science Quarterly*, 40 (1), 84-110.

Farrell, Joseph and Garth Saloner (1985), "Standardization, Compatibility and Innovation," *The Rand Journal of Economics*, 16 (1), 70-83.

Faust, Katherine (1997), "Centrality in Affiliation Networks," *Social Networks*, 19 (2), 157-91.

Faust, Katherine and Romney A. Kimball (1985), "Does Structure find Structure?: A Critique of Burt's use of Distance as a Measure of Structural Equivalence," *Social Networks*, 7 (1), 77-103.

Fink, Martin (2003), *The Business and Economics of Linux and Open Source*. Upper Saddle River, NJ: Prentice Hall.

Fiol, Marlene C. and Marjorie A. Lyles (1985), "Organizational learning," *Academy of Management Review*, 10 (October), 803-13.

Freeman, John, Glenn R. Carroll, and Michael T. Hannan (1983), "The Liability of Newness: Age Dependence in Organizational Death Rates," *American Sociological Review*, 48 (5), 692-710.

Freeman, L.C (1977), "A Set of Measures of Centrality Based on Betweenness," *Sociometry*, 40, 35-41.

Freeman, Linton C. (1979), "Centrality in Social Networks: Conceptual Clarification," *Social Networks*, 1 (3), 215-39.

Frenzen, Jonathan and Kent Nakamoto (1993), "Structure, Cooperation, and the Flow of Market Information," *Journal of Consumer Research*, 20 (3), 360-75.

Galli, Peter (2005a), "IBM's New Linux Strategy Pushes Solution Sets " *www.eweek.com*, Accessed May 18, 2006, http://www.eweek.com/article2/0,1895,1850734,00.asp.

---- (2005b), "IBM gives 500 Patents to Open-Source Developers," *eweek.com*, Accessed October 28, 2006, http://www.eweek.com/article2/0,1759,1750358,00.asp.

Golder, Peter N. and Gerard J. Tellis (1997), "Will It Ever Fly? Modeling the Takeoff of Really New Consumer Durables," *Marketing Science*, 16 (3), 256-70.

Gould, Roger V. and Roberto M. Fernandez (1989), "Structures of Mediation: A Formal Approach to Brokerage in Transaction Networks," *Sociological Methodology*, 19, 89-126.

Granovetter, Mark S. (1973), "The Strength of Weak Ties," *American Journal of Sociology*, 78, 1360-80.

---- (1985), "Economic Action and Social Structure: The Problem of Embeddedness," *The American Journal of Sociology*, 91 (3), 481-510.

Greene, William H. (1995), "Sample Selection in the Poisson Regression Model," *Working Paper*, Working Paper EC-95-6, Department of Economics, Stern School of Business, New York University.

---- (2003), *Econometric Analysis* (5th ed.). Upped Saddle River, NJ: Prentice Hall.

---- (2006), "A General Approach to Incorporating Selectivity in a Model," *Discussion Paper*, EC-06-10, Department of Economics, Stern School of Business, New York University.

Grewal, Rajdeep, Gary L. Lilien, and Girish Mallapragada (2006), "Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems," *Management Science,* 52 (7), 1043-56.

Griffin, Abbie (1997), "The Effect of Project and Process Characteristics on Product Development Cycle Time," *Journal of Marketing Research*, 34 (1), 24-35.

Gulati, Ranjay and Martin Gargiulo (1999), "Where do Interorganizational Networks Come From?," *American Journal of Sociology*, 104 (5), 1439-93.

Hauser, John, Gerard J. Tellis, and Abbie Griffin (2006), "Research on Innovation: A Review and Agenda for Marketing Science," *Marketing Science*, 25 (6), 687-720.

Holland, Paul W. and Samuel Leinhardt (1977), "A Dynamic Model for Social Networks," *Journal of Mathematical Sociology*, 5, 5-20.

Huisman, Mark and Marijtje A.J. van Duijn (2003), "StOCNET: Software for the Statistical Analysis of Social Networks," *Connections*, 25(1), 7-26.

Iacobucci, Dawn and Nigel Hopkins (1992), "Modeling Dyadic Interactions and Networks in Marketing," *Journal of Marketing Research*, 29 (1), 5-17.

IDC (2004), "The Linux Marketplace - Moving From Niche to Mainstream," *OSDL*, Accessed October 20, 2006, http://old.linux-foundation.org/docs/linux_market_overview.pdf.

Katz, Daniel and Robert L. Kahn (1966), *The Social Psychology of Organizations*. New York, NY: John Wiley & Sons.

Kiefer, Nicholas M. (1988), "Economic Duration Data and Hazard Functions," *Journal of Economic Literature*, 26 (2), 646-79.

Kilduff, Martin, Wenpin Tsai, and Ralph Hanke (2006), "A Paradigm too far? A Dynamic Stability Reconsideration of the Social Network Research Program," *Academy of Management Review*, 31 (4), 1031-48.

Lacy, Sarah (2005), "Open Source: Now It's an Ecosystem," Accessed 28 October, 2006, http://www.businessweek.com/technology/content/oct2005/tc2005103_0519_tc_218.htm.

Lakhani, Karim R. and Eric von Hippel (2002), "How Open Source Software Works: “Free” user-to-user Assistance," *Research Policy*, 1451, 1-21.

Lakhani, Karim R. and Robert G. Wolf (2003), "Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects," in MIT Sloan Research Paper Series, Working Paper no. 4425-03.

Lambkin, Mary and George S. Day (1989), "Evolutionary Processes in Competitive Markets: Beyond the Product Life Cycle," *Journal of Marketing*, 53 (3), 4-20.

LaMonica, Martin (2005), "Open Source, Open Wallet," *cnet.com*, Accessed September 15, 2006, http://news.com.com/Open+source,+open+wallet/2100-7344_3-5934144.html.

Laumann, Edward O., Peter V. Marsden, and David Prensky (1989), "The Boundary Specification Problem in Network Analysis," in *Research methods in Social Network Analysis*, Linton C. Freeman and Douglas R. White and Romney A. Kimball, Eds. Fairfax, VA: George Mason University Press.

Lee, Gwendolyn K. and Robert E. Cole (2003), "From a Firm-Based to a Community Based Model of Knowledge Creation: The Case of the Linux Kernel Development," *Organization Science*, 14 (6), 633-49.

Lee, Lung-Fei (1992), "On Efficiency of Methods of Simulated Moments and Maximum Simulated Likelihood Estimation of Discrete Response Models," *Econometric Theory*, 8 (4), 518-52.

Lerner, Joshua and Jean Tirole (2002), "Some Simple Economics of Open Source," *Journal of Industrial Economics*, 50 (2), 197-234.

---- (2005), "The Economics of Technology Sharing: Open Source and Beyond," *The Journal of Economic Perspectives*, 19 (2), 99-121.

Leslie, Keith J. and Max P. Michaels (1997), "The Real Power of Real Options," *The McKinsey Quarterly* (3).

Lessig, Lawrence (2001), *The Future of Ideas: The Fate of The Commons in a Connected World*. New York: Random House.

Lilien, Gary L., Pamela D. Morrison, Kathleen Searls, Mary Sonnack, and Eric von Hippel (2002), "Performance Assessment of the Lead User Idea - Generation Process for New Product Development," *Management Science*, 48 (8), 1042-59.

Lin, Nan (2001), "Building a Network Theory of Social Capital," in *Social Capital: Theory and Research*, Nan Lin and Karen Cook and Ronald S. Burt, Eds. Hawthorne, NY: Aldine De Gruyter.

Lowry, Bruce (2006), "Microsoft and Novell Announce Broad Collaboration on Windows and Linux Interoperability and Support," *www.novell.com*, Accessed 9 May, 2007, http://www.novell.com/news/press/microsoft_and_novell_announce_broad_collaboration_on_windows_and_linux_interoperability_and_support.

MacCormack, Alan, John Rusnak, and Carliss Y. Baldwin (2006), "Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code," *Management Science*, 52 (7), 1015-30.

Madey, Greg (2005), "SourceForge.net Research Data Archive," Accessed October 2005, 2005.

Mahajan, Vijay, Eitan Muller, and Roger A. Kerin (1984), "Introduction Strategy for New Products with Positive and Negative Word-of-Mouth," *Management Science*, 30 (12), 1389-405.

Mahajan, Vijay, Eitan Muller, and Yoram Wind Eds. (2000), *New Product Diffusion Models*. Boston, MA: Kluwer Academic Publishers.

Mann, Jennifer (2007), "IBM Unveils New Business Continuity Specialty Accreditation for Select Mid-Market-Focused Business Partners," *IBM press release*, Accessed April 19, 2007, http://www-03.ibm.com/press/us/en/pressrelease/21385.wss.

Mansfield, Edwin and Samuel Wagner (1975), "Organizational and Strategic Factors Associated with Probabilities of Success in Industrial R & D," *The Journal of Business*, 48 (2), 179-98.

Marson, Ingrid (2004), "IBM Releases Source Code of Speech Software," Accessed, http://asia.cnet.com/builder/program/unix/0,39009368,39195316,00.htm.

McFadden, Daniel (1989), "A Method of Simulated Moments for Estimation of Discrete Response Models Without Numerical Integration," *Econometrica*, 57 (5), 995-1026.

McGrath, R. G. (1997), "A Real Options Logic for Initiating Technology Positioning Investments," *The Academy of Management Review*, 22 (4), 974-96.

McMillan, Robert (2004), "IDC: Linux to take 29 percent of 2008 server shipments " *infoworld.com*, Accessed, http://www.infoworld.com/article/04/06/17/HNidclinux_1.html.

Menchaca, Lionel (2007), "Dell to Offer Ubuntu 7.04," *www.direct2dell.com*, Accessed May 9, 2007, http://direct2dell.com/one2one/archive/2007/05/01/13147.aspx.

Mizruchi, Mark S. and Blyden B. Potts (1998), "Centrality and power revisited: actor success in group decision making," *Social Networks*, 20 (4), 353-87.

Obstfeld, David (2005), "Social Networks, the Tertius Iungens Orientation, and Involvement in Innovation," *Administrative Science Quarterly*, 50 (1), 100-30.

Oehlert, Gary W. (1992), "A Note on the Delta Method," *The American Statistician*, 46 (1), 27-29.

Pallatto, John and John G. Spooner (2005), "Dell, IBM Offer Server Products Geared for Linux " *eweek.com*, Accessed October 24, 2006, http://www.eweek.com/article2/0,1895,1845503,00.asp.

Powell, Walter W., Kenneth W. Koput, and Laurel Smith-Doerr (1996), "Interorganizational Collaboration and the Locus of Innovation: Networks of Learning in Biotechnology," *Administrative Science Quarterly*, 41 (March), 116-45.

Raymond, Eric (1999), *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. Cambridge, MA: O'Reilly.

Reingen, Peter H., Brian L. Foster, Jacqueline J. Brown, and Stephen B. Seidman (1984), "Brand Congruence in Interpersonal Relations: A Social Network Analysis," *Journal of Consumer Research* 11 (3), 771-83.

Rindfleisch, Aric and Christine Moorman (2001), "The Acquisition and Utilization of Information in New Product Alliances: A Strength-of-ties Perspective," *Journal of Marketing*, 65 (2), 1-18.

Robbins, H. and S. Monro (1951), "A Stochastic Approximation Method," *Annals of Mathematical Statistics*, 22, 400-07.

Rodan, Simon and Charles Galunic (2004), "More than Network Structure: How Knowledge Heterogeneity Influences Managerial Performance and Innovativeness," *Strategic Management Journal*, 25 (6), 541-62.

Schumpeter, Joseph A. (1975), *Capitalism, Socialism and Democracy*. New York, NY: Harper & Row.

Shah, Sonali K. (2006), "Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development," *Management Science*, 52 (7), 1000-14.

SIIA (2006), "Software Growth Report," *Software and Information Industry Association*, Accessed 20 April, 2007.

Snijders, Tom A. B. (2005), "Models for Longitudinal Network Data," in *Models and Methods in Social Network Analysis*, Peter J. Carrington and John Scott and Stanley Wasserman, Eds. New York: NY: Cambridge University Press.

Snijders, Tom A. B., Christian E.G. Steglich, and Michael Schweinberger (2005a), "Manual for Siena Version 2," *ICS, University of Groningen*, Accessed 24 October, 2006, http://stat.gamma.rug.nl/snijders/siena.html.

---- Eds. (2005b), *Modeling the Co-evolution of Networks and Behavior*: Lawrence Erlbaum.

Snijders, Tom A.B (1996), "Stochastic Actor-oriented Models for Network Change," *Journal of Mathematical Sociology*, 21 (1-2), 149-72.

Swaminathan, Anand (1996), "Environmental Conditions at Founding and Organizational Mortality: A Trial-by-Fire Model " *Academy of Management Journal*, 39 (5), 1350-77.

Taudes, Alfred, Markus Feurstein, and Andreas Mild (2000), "Options Analysis of Software Platform Decisions: A Case Study," *MIS Quarterly*, 24 (2), 227-43.

Tellis, Gerard J. and Merle C. Crawford (1981), "An Evolutionary Approach to Product Growth Theory," *Journal of Marketing*, 45, 125-32.

Terza, Joseph V. (1998), "Estimating Count Data Models with Endogenous Switching: Sample Selection and Endogenous Treatment Effects," *Journal of Econometrics*, 84 (1), 129-54.

Torvalds, Linus and David Diamond (2001), *Just for Fun: The Story of an Accidental Revolutionary*: HarperCollins.

Train, Kenneth E. (2003), *Discrete Choice Models with Simulation*. Cambridge: Cambridge University Press.

Uzzi, Brian (1996), "The Sources and Consequences of Embeddedness for the Economic Performance of Organizations: The Network Effect," *American Sociological Review*, 61 (August), 674-98.

---- (1997), "Social Structure and Competition in Interfirm Networks: The Paradox of Embeddedness," *Administrative Science Quarterly*, 42 (1), 35-67.

---- (1999), "Embeddedness in the Making of Financial Capital: How Social Relations and Networks Benefit from Firms Seeking Financing," *American Sociological Review*, 64 (4), 481-505.

Van den Bulte, Christophe and Rudy K. Moenaert (1998), "The Effects of R&D team Co-location on Communication Patterns among R&D, Marketing, and Manufacturing," *Management Science*, 44 (11), S1-18.

Van den Bulte, Christophe and Gary L. Lilien (2001), "Medical Innovation Revisited: Social Contagion Versus Marketing Effort," *The American Journal of Sociology*, 106 (5), 1409-35.

von Hippel, Eric (1989), "New Product Ideas from 'Lead Users'," *Research Technology Management*, 32 (3), 24-27.

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

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

von Krogh, Georg and Eric von Hippel (2006), "The Promise of Research on Open Source Software," *Management Science*, 52 (7), 975-83.

Wasserman, Stanley and Katherine Faust (1999), *Social Network Analysis: Methods and Applications*. Cambridge, UK: Cambridge University Press.

Weick, Karl E. (1976), "Educational Organizations as Loosely Coupled Systems," *Administrative Science Quarterly*, 21, 1-19.

Weiss, Todd R. (2004), "HP Touts Four-in-One Linux PC," *pcworld.com*, Accessed, http://www.pcworld.com/news/article/0,aid,116899,00.asp.

Whitaker, Teri (2007), "What Customers Are Saying About Oracle® Unbreakable Linux Support " *www.oracle.com*, Accessed May 7, 2007, http://www.oracle.com/corporate/press/2007_mar/unbreakablelinuxcustomers.html.

Xu, Jin, Scott Christley, and Greg Madey (2006), "Application of Social Network Analysis to the Study of Open Source Software," in *The Economics of Open Source Software Development*, Jürgen Bitzer and Philipp J.H. Schröder, Ed.: Elselvier Press.

Zukin, Sharon and Paul J. DiMaggio Eds. (1990), *Structures of Capital: The Social Organization of the Economy*, New York: Cambridge University Press.

Appendix A

Data access from SourceForge.net

The data that I have used in my two essays has been obtained from the SourceForge.net website.

Almost all the data that was used in the first essay was from the SourceForge.net data warehouse hosted at the University of Notre Dame. Data concerning projects can be downloaded from this database by using the query page that has been provided on the data warehouse project page. For convenience sake, I downloaded the required data on project birth, release timing, activity status, project-developer relationships, project characteristics for all the projects in the system by using the data access form and stored it on my storage drives in Microsoft Access format. I then used this data to extract information for the sample used in my first essay by creating my own SQL queries on this database.

In the second essay, I needed data on the longitudinal transition of project-developer relationships and on the number of downloads over time. However, as the data warehouse did not have updated data on the number of downloads, I visited the project home pages of the open source projects in my sample, and manually collected the data from the web pages of the projects.

Appendix B

Overview of the Open Source Movement

B.1 Open Source: Free as in "free speech" not "free beer"

B.1.1 The open source ideology

Software that we use in everyday life is primarily from commercial vendors who charge the users a license fees, implying that the user does not actually "own" the software product but is only licensed to use it within the functional constraints placed by the vendor. Such a licensing system awards the user very little rights over the functionality of the product and often customization cannot be done, and this is achieved by keeping the source code of the product closed. In contrary, the open source movement, which started with Stallman's FSF and resulted in products such as Linux and MySQL, allows users access to the source code thereby creating an open environment. The products are developed by individuals who collaborate voluntarily on peer-reviewed software which are made available for usage through various open source licensing mechanisms[6].

The closed licensing paradigm has been followed in the information product world as it is very easy to create a copy of the product. For instance, it would be almost impossible for a user to recreate a car or a computer from scratch, and sell it for profit, after having purchased it from the original vendor. In contrary, a user can create copies of

---

[6] For a detailed review on open source licenses, please refer Fink 2002.

an information product at very low cost and gain profits from the redistribution of it. Although, copyright laws and their enforcement act as legal deterrents for such behavior, the licensing mechanism places additional constraints on the rights of users. These include constraints on redistribution as well as access to the source code of the software. In reality, the actual value of the software does not come from the source code of the product but from the actual usage of the product itself. By restricting access to the source code, the vendor usually gains long term strategic advantages which I detail below.

By keeping the source code closed: 1) the vendor restricts the user from modifying the product 2) the vendor can determine which other products cane be used complementarily and 3) can create lock-ins for future upgrades and additional functionality. The open source movement takes its roots in this fundamental notion of “freeness” or “openness” of the source code. Popular open-source licenses such as the GPL and LGPL allow users access to the source code of the product thereby allowing them to modify the product to their advantage, use the product complementarily with other products as they deem fit.

B.1.2 A brief history of the open source movement

B.1.2.1 Stallman’s free world

It is very important to understand the historical underpinnings of the Open Source movement to fully appreciate the tenacity with which it is often defended in the marketplace. The notion of collaboration on Internet technologies itself is quite an old one dating back to the efforts of DARPA and the evolution of the TCP/IP protocols in the early seventies. With the advent of ARPANET (precursor to the modern Internet), technology enthusiasts at various US universities where collaborating with each other in developing solutions to everyday software technology problems.

In such an open collaborative environment, Richard Stallman, arguably the founder of the free software movement belonged to a collaborative group at MIT's AI lab. The community's objective was to develop solutions to software related problems through open sharing of the source code, peer reviews and peer development. However, the rise of commercial software firms in the early 80's, a number of the developers at the lab (they were referred to as hackers then) were hired away. Therefore, Richard Stallman launched the Free Software Foundation with a goal to develop a free and open source operating system like UNIX, and gave it a name that is a recursive acronym GNU (GNU's not UNIX). In Stallman's ideology free software did not mean software at zero cost. In his own words it was, "free as in free speech, not as in free beer" (). Stallman succeeded in building most of the components (GNU C Compiler, Emacs editor) that were needed to assemble a fully functional open source operating system with a reliability matching that of commercial offerings.

B.1.2.2 Linus Torvalds and the birth of Linux

However, his efforts on building the kernel, the core of the operating system were delayed. Around this time, Linus Torvalds, a young Finnish student, who required a more advanced operating system than the one he was using (MINIX), developed one on his own using the tools that Stallman developed and called it Linux. He posted news about his effort on newsgroups resulting in scores of talented programmers joining his effort to improve the Linux Operating system. Soon, he designed a penguin to stand as the logo for his OS.

Linux that started in an individual's effort to solve a personal problem resulted in an operating system that would eventually be adopted by millions of individuals. As of today there are over 10,000 developers who actively participate in the Linux kernel project with Linus Torvalds and heading the initiative. The success of Linux gave birth to killer applications such as the Apache web server, MySQL database and the Mozilla web browser, to name a few popular projects. It renewed the interest in open source ideology and created huge online efforts such as SourceForge.net with over a million registered users and a hundred thousand projects. More importantly from a marketing perspective it also renewed the interest of corporations such as IBM, Dell, Sapient and others which see a strategic advantage in participating in the open source revolution early on. Innovation through peer-reviewed, peer developed software solutions has begun to emerge as a viable threat to the firm driven closed development model ().

B.1.2.3 Killer apps and the rise of Linux

The Apache Web Server which is arguably the best known application in the open source world holds an impressive 60% market share in the Internet web-server[7] market. It is not commercially backed by any single firm and yet beats mainstream commercial

---

[7] Web-servers are software programs which host content on the Internet. Web pages written in various languages such as html, java, and perl etc. are “served” on request from client browsers.

offerings such as Microsoft's IIS hands down in technological sophistication and addressing users' needs. There are many standing examples similar to Apache which have shown that customer needs can be served outside the traditional firm-driven paradigm of product innovation.

The primary reason for this is path dependence created by Microsoft's efforts in developing a GUI based OS suited to meet the requirements of desktop users. In contrary, notions of GUI did not make inroads into the Linux world late until 2000, when two software foundations GNOME and KDE began. However, since then the open source world has come up with stable and robust GUI's for the Linux OS thereby allowing it to make inroads into the desktop market. Moreover, the design strength[8] of the Linux OS enables it to be ported onto diverse platforms varying from wrist watches to CRAY super computers.

B.1.2.4 Free to open: Beginnings of a new age

Although the concept of open source itself is over twenty years old (then referred to as the Free Software Movement), the impact of the movement on the business world took off with the start-up of a firm called Cygnus Solutions, which effectively came up with the first viable open-source business model[9]. Cygnus offered support on the GNU C complier and created the first service model revolving around the open source ideology. Involvement of big commercial vendors publicly took off with Netscape Communications

---

[8] It is based on a monolithic kernel design, for a detailed discussion see

[9] Cygnus Solutions was acquired by RedHat in 2000.

deciding to open up the source code of its popular browser Netscape Communicator, following the publication of Eric Raymond's famous article, The Cathedral and the Bazaar. Eric and a few others formed the Open Source Initiative (www.osi.org) in a few days, to create a common voice for the open source world.

The re-branding of the free software movement began with more stress being placed on the openness of it than the freeness of it. Although Stallman differs in ideology with the open source world, the current state of the movement owes a lot to his fanatical belief in the core - that of innovation through peer development and peer review. As of today, the open source revolution has gained commercial acceptance with major hardware vendors such as IBM and Dell announcing the availability of Linux based solutions in their portfolio.

Appendix C

Technical appendix for the estimation procedure used in Chapter 3

For the sake of convenience we replicate the technical details of the estimation procedure that was used in the data analysis for Chapter 3. The model parameters can be estimated by solving the following moment equation,

$$E_{\hat{\theta}}(s(Y)) = s(y) \qquad \text{C.1}$$

where $s(y)$ is the observed value of the network statistic – a function of the network data. The asymptotic covariance matrix of the estimator can be obtained using the delta method as,

$$Cov_{\theta}(\hat{\theta}) \approx D_{\theta}^{-1} \sum\nolimits_{\theta} D_{\theta}^{'-1} \qquad \text{C.2}$$

where D_{θ} is the matrix of partial derivatives obtained as,

$$D_{\theta} = \left(\frac{\partial E_{\theta}(s(Y))}{\partial \theta} \right) \qquad \text{C.3}$$

And $\sum_{\theta}$ is the estimate of the covariance matrix given by,

$$\sum\nolimits_{\theta} = \operatorname{cov}_{\theta}(s(Y)) \qquad \text{C.4}$$

To calculate the conditional expectations of the moment equation c.1, the estimation method uses a stochastic approximation by simulating the stochastic process. The simulations of the stochastic process use the transition probability matrix as defined by equation 3.8 and stated briefly, either the network or the outcome variable is chosen as the conditioning variable, and the condition is that the simulated distance on the variable chosen for conditioning is equal to the observed distance. The observed distance for the variables is the difference between two states and is calculated as the number of unitary steps that are required to transition to the next observation moment from the current one. The first observation moment is taken as given and the simulations are initiated till the second observation moment is reached and then the process is continued till all the observation moments are exhausted.

VITA

Girish Mallapragada

Assistant Professor of Marketing
Kenan-Flagler Business School
The University of North Carolina at Chapel Hill
Campus Box#3490, McColl Building
Chapel Hill, NC 27599

Phone: (919) 962 2149
Email: girishm@unc.edu

EDUCATION

2002 - 2007	PhD in Marketing, Penn State University, University Park (expected May 2008)
March 2001	MBA, Indian Institute of Management, Lucknow, India *Focus: Marketing and Information System*
June 1999	B.E (Mechanical), Osmania University, Hyderabad, India

PUBLICATIONS

Grewal, Rajdeep, Gary L. Lilien, and Girish Mallapragada (2006), "Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems," Management Science, 52, 7, 1043-1056.

- Equal Contribution

HONORS and RESEARCH AWARDS

- ISBM Scholarship, Penn State University, 2005-2007
- Smeal Doctoral Dissertation Award, Penn State University, 2006
- Peter Liberti and Judy D. Olian Scholarship, Penn State University, 2006
- ISBM Business Marketing Doctoral Support Award, 2005
- David T. Wilson Scholarship, Penn State University, 2005
- AMA Doctoral Consortium Fellow, University of Connecticut, 2005
- Paul F. Anderson Scholarship, Penn State University, 2004
- Haring Symposium Fellow, Indiana University, 2004
- Smeal Small Research Grant, Penn State University, 2003
- Best Candidacy Exam performance, Penn State University, 2003
- Smeal Scholarship, Penn State University, 2002

INDUSTRY EXPERIENCE

- Pramati Technologies, Hyderabad, India, 2001-2002, Area Sales Manager
- Hindustan Coca-Cola Beverages Pvt. Ltd., 2000, Graduate Summer Intern.