Successful or Unsuccessful Open Source Software Projects: What is the key?

Xiaohong Chen*, David Probert, Yuan Zhou, Jun Su
Xiaohong Chen*. School of Public Policy and Management,Tsinghua University. Beijing, China
David Probert. Institute for Manufacturing, University of Cambridge. Cambridge, UK
Yuan Zhou. School of Public Policy and Management,Tsinghua University. Beijing, China
Jun Su. School of Public Policy and Management,Tsinghua University. Beijing, China
foreverheart9@gmail.com, drp1001@cam.ac.uk, zhou_yuan@mail.tsinghua.edu.cn, sujun@mail.tsinghua.edu.cn

***Abstract*—This paper aims to analyse the key factors influencing knowledge sharing in open source software projects. Four deep cases are analyzed to develop a conceptual framework based on within-case and cross-case analysis. In terms of data collection, online (including skype meeting, email, email-list, IRC, forum, group meeting, etc) and offline (mainly with face-to-face discussion) activities are two major platforms. The research framework is that distributed innovation (independent variable) will influence shared knowledge (medium variable) and continue to affect the performance of OSS projects (dependent variable). During distributed innovation, developers located on the supply side will affect the shared knowledge from the aspects of Participative Motivation, Social Network and Organizational Culture. Meanwhile, users situated on the demand side will function from the view of user innovation.**

Keywords—knowledge Sharing; Open Source Software Projects; User Innovation; Influencing Factors

I. INTRODUCTION

Open source refers to a computer program in which the source code is available to the general public for use and/or modification from its original design [1]. Open source Software is a specific case under the phenomenon of "Open Innovation" [2].

However, not all open source software projects are successful. There are opposite performances of open source software projects, namely some are quite successful while some are not. In this paper, the author would like to analyse the key factors influencing the quality of open source software projects. In another words, what are the key factors under the success of OSS projects?

II. LITERATURE REVIEW

A. Open Source Software

Open Source Software (OSS) is computer software with its source code made available with a license in which the copyright holder provides the rights to study, change and distribute the software to anyone and for any purpose. The Linux operating system, Mozilla and Chrome are all outstanding representatives of open source software, which is widely used all over the world. The reason why it attracts much attention by academia, industrialists and governments is its special characteristics, including free code [3], qualified production [4], short technology cycle [5] and its information security. Different from proprietary software with "the cathedral model ", OSS possesses " the bazaar model " in which everyone can get involved or leave at any time (Table 1). Any one has equal rights to contribute [6].

TABLE I. DIFFERENCE BETWEEN THE CATHEDRAL MODEL AND BAZAAR MODEL

Factor	**Cathedral Model (Traditional collaboration)**	**Bazaar Model (Distributed Innovation)**
Structure	Hierarchical	Flat network
Participants	Task oriented	Interest oriented
Production	Management by Bureaucracy	Management by Objective
Division of work	Distributed by leaders	Self-control
Knowledge flow	Top-down, bottom-up	Distributed
Release	Release after final revision	Continuous revision and release
Decision model	Concentrated	Decentralized

Adapted from: Jitesh H.Panchal, Meryvn Fathianathan. Product realization in the age of mass collaboration". ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. 2008,1(34): 219-229

TABLE II. RESEARCH DESIGN OF CASE ANALYSIS

No.	Item	Content	
(A)	Research Questions	What are core factors affecting knowledge sharing in OSS projects?	
(B)	Theory Proposition	Distributed innovation (independent variable) will influence shared knowledge (medium variable) and continue to affect the performance of OSS projects (dependent variable).	
(C)	Research Unit	OSS project	
(D)	Logical Relationship	Pattern matching between research questions and case questions	
(E)	Data Assessment	Construct Validity	Various evidence to collect data
		Internal Validity	Causal relationship between variables
		External Validity	Multi-case analysis with copy principle
		Reliability	Data collecting and deep analysis to each case

B. Open Source and knowledge sharing

Knowledge is the hinge of competitive advantages [7]. The value of knowledge is not realized until it is shared. Knowledge sharing is a collaborative process where ideas and knowledge are transferred so as to solve problems [8]. During Open source development, project code, documentation and suggestions are all part of necessary knowledge. Consequently, suitable mechanisms of knowledge sharing in OSS projects will be helpful in creating innovative software. It is universally recognized the key to OSS projects lies in knowledge sharing. Consequently, this paper aims to explore the factors influencing performance of knowledge sharing in OSS projects.

III. METHODOLOGY

A. Research Design

Case study is one of most popular research methods in the social science field, which plays an important role in theory building and explaining the relationship between various variables [9]. It is usually helpful to realize the significance of facts through deep data collection and analysis [10]. There are five basic factors in a case study, including the research questions, research hypothesis (if included), unit of research, logic between the data and hypotheses, and criteria for explaining the outcomes. Accordingly, the research design for a case analysis is as follows (Table 2)

B. Case Selection

The author obtained a qualified open source software project database with 70 cases. Combined with cases recommended by interviewees, the author selected 10 cases to make a deep analysis from which 4 cases were finally to represent a range of different conditions. In terms of data collection, online (including skype meeting, email, email-list, IRC, forum, group meeting, etc) and offline (mainly with face-to -face discussion) activities are two major platforms by interview, archival review, participation, documentation analysis and so on. Meanwhile, as an ordinary user, the author took part in 3 open source software projects and had a great opportunity to understand the details and process of projects. During the interview, the author asked interview questions according to the research question and theoretical framework that lasted 1.5 to 2 hours for each case interview. After every interview, interview notes were recorded and sent to interviewees to make a confirmation or further alteration. The interviewees are mainly located in China, UK and USA, and data collection lasted about 6 months: From April to July in 2014, the main work was mass interview, theoretical framework and proposition. From August to October in 2014, the author mainly spent time on case data collection, confirmation and analysis.

IV. DATA ANALYSIS

A. Brief Introduction to Cases

1) APDPlat

APDPlat is an abbreviation of Application Product Development Platform started from 2008. After 4 years later, it became open source in the OSS community of Github. The main users of this project are large numbers of JAVA engineers. As to the beginners of JAVA, it is a very good platform to learn software design and coding arts. To the middle learners, it provides an excellent power to help engineers work faster based on information system. In the "Chinese Open Source Software Competition in 2013" sponsored by Chinese Open Source Promotion Association and Chinese greatest IT community, CSDN, the APDPlat project ranked No.3 with 2094 votes. Now during these two years, there are 890 developers who forked this project and continue to communicate with this project, which is a very successful OSS project. It is a very successful OSS project mainly worked on by Chinese.

2) 12306NG

The full name of 12306NG is 12306 Next Generation, which was established on September 27th 2012.

It was inspired by the fact that most developers cannot buy a train ticket to return home online, so the Vice-CTO of Jingdong Company called for this project to develop a new train ticket booking system. The background of 12306NG is that the online booking system always breaks down and has a negative effect on people's lives. The 12306NG project consists of 10 modules, each of which has a specific production and expectation, which has attracted more than 10000 developers to participate from September 30th to November 10th 2002. There were about 80 core members, and it ended in failure, with no important code or outcomes after 6 months. It is an unsuccessful OSS project mainly worked on by Chinese.

3) VNC

VNC (Virtual Network Computing) is a graphical desktop sharing system that uses the Remote Frame Buffer protocol (RFB) to control another computer remotely. It transmits the

keyboard and mouse events from one computer to another, relaying the graphical screen updates back in the other direction, over a network. Now, it is one of the most popular implementations in the world. The project was open source from 1998, developed by the Olivetti & Oracle Research Lab in Cambridge, and then was required into AT&T in 1999. In 2002, this research lab closed down，several core members of the VNC team then formed RealVNC to continue working on open-source and commercial VNC software under that name. There are five core members of VNC and now the company that developed from this project has 41734 developers worldwide. It is widely used in companies, governments and educational institutions all over the world. It is a very successful OSS project mainly worked on by the West.

4) Chandler

The Chandler project was established in 2002 and aimed to create a new personal information manager, including a calendar, email, tasks, notes, etc. The plan was to realize information sharing at any time between a business computer and a home computer without third party intervention. The project attracted great attention from the start due to the social effect of the founder of this project, who even accepted greetings from the Times in America and Vice President Gore. This project lasted 7 years along with 3 core members at the beginning, then developed into 12 members, 27 members and finally developed into a small team with 10 core members. The project finally failed without an essential outcome, even though each member of this project was a computer scientist, wrote tens of thousands of lines of code, fixed 4732 bugs and spent lots of money. It is an unsuccessful OSS project mainly worked on by the West.

B. Cases Analysis

1) Relationship between distributed innovation and knowledge sharing

a) Participative Motivation and Shared Knowledge

Open source is not only an economic phenomenon but also a social phenomenon. It is a very interesting phenomenon in that many developers have spent considerable time and energy contributing to open source voluntarily while the code is free [11]. Self-Determination Theory holds the opinion that motivation affects behavior [12].

In open source software projects, intrinsic and extrinsic motivation will have a significant effect on knowledge sharing [13]. From the aspect of intrinsic motivation, there are three main factors, including enjoyment and fun, technical learning and altruism. In terms of extrinsic motivation, it mainly includes the economic incentive, career and reputation.

In APDPlat, the market user is the JAVA engineer and the project aims to provide new simplified technology, and the return has covered the cost. As to 12306NG, the strong motivation to contribute to society and help colleagues to buy train tickets, given that the existing website always crashed, encouraged the founder to launch this project. In terms of VNC, making money and supporting employees was one of the great motivations to develop this project since RealVNC Company was created based on this project, so they have to continually develop and introduce innovations for this project. The creator of Chandler is famous for its social influence in the field of computer science because of Project Lotus 1-2-3. Taking into consideration the existing shortcomings of the software offered by Microsoft Exchange and Outlook, he wanted to launch a new project to replace them and make a contribution to society with strong social responsibility. Accordingly, Participative Motivation has a positive effect on shared knowledge.

b) Social Networks and Shared Knowledge

Social networks emphasize the network relationship among the individuals, teams, organizations and communities that possess social resources, which can be divided into cognitive, relational and structural social networks [14].

Specifically, cognitive social networks emphasize shared language and culture [15]. Relational social networks focus on the link among the participants, such as trust, rules, and reciprocity. Structural social networks focus on the interaction among participants, including the frequency and weakness of social ties. Cognitive social networks influence shared knowledge from three aspects: (a) shared goals (b) shared language and (c) shared culture. Relational social networks will affect shared knowledge in three aspects, including trust, rules and reciprocity. Structural social networks emphasize two aspects, (a) the frequency of interaction and (b) the weakness of ties. There are two important potential problems preventing knowledge sharing, i.e free-riding and information stickiness, which can be effectively weakened or eliminated by social networks.

In the APDPlat project, as creator of this project holds a firm opinion that it is essential to distinguish real open source and fake open source as open source itself has a brand effect. During the process, there will be other participants forking this project if it meets their own aim. The communication is mainly divided into online and offline. As to the 12306NG project, the participants had the public platform to communicate and they organized Wechat group meetings every Sunday evening at 8pm to discuss issues. Even though they had the ticket design plan, there was not having an obvious outcome about the code, which is the basic and necessary part of OSS projects. What is worse, it was difficult to form a recognized decision with so many people involved in the project. In the VNC project, the core members felt enjoyment and that it was easier to discuss issues, as every core member in the VNC project belonged to the same research lab. In the Chandler project, everyone respected and trusted each other since almost everyone involved in this project graduated from top universities worldwide and are experts in computer science. Due to a shared office, they can discuss conveniently and take part in group meetings every week while, because the target of this project is too grand and it is difficult to reach an agreement, it is tough to implement a specific plan for the project. Therefore, Social Networks between participants have a positive effect on shared knowledge.

c) Organizational Culture and Shared Knowledge

Organizational culture has potential but profound effects on shared knowledge [16] through invisible value [17], including openness, collaborative sharing and innovative spirit. In terms of openness, this mainly emphasizes fluency and the speed of

knowledge flow. Generally, when an organization encourages free communication, the speed of knowledge will be dramatically increased. In terms of collaborative sharing, this will encourage the better application of knowledge management tools in organizations [18]. Regarding the spirit of innovation, there is another interesting name, a "geek", for developers who are interested in developing, devoting time and changing people's lives through technology innovation. From this perspective, the history of open source is another history of hacking [19].

In the APDPlat project, the license he selects is the GPLV3 as it is a good way to show the spirit of OSS and the key proposals made by Richard Stallman. There are some IP problems even though the collaboration among participants is fine. As in the 12306NG project, the information within the project is open to everyone and higher expectations were set in this project, hoping to design a new ticket system a year later. There are 2-3 core members and 4-5 main developers in each module; therefore, roughly 80 core members participated in this project but, because most people spent a lot of time designing the framework, there are few essential outcomes. Regarding the VNC project, developers are free to communicate and since the core members are located near to each other, the sharing collaboration is fluent and pleasant. The geek spirit is strong and continuous technology innovations are created. In the Chandler project, the communication among developers is free and the process is open to everyone at any time. The participants had high expectations of this project while there were many problems among them. On the one hand, the decision-making is slow due to collective decisions. On the other hand, the distribution among developers has some problems. The project is open source, while it managed a project through a combination of proprietary software management and open source software management. Many of the core members chose to quit this project due to their extended participation without any valuable outcome or important progress. Hence, Organizational culture in the developers' minds has a positive effect on shared knowledge.

2) Relationship between user innovation and knowledge sharing

User innovation means some individuals or companies creating productions or providing services innovatively [20]. Whatever the industrialized product [21], consumer goods [22] or new business model [23], users are an important source of innovation. Open source is a representative case to explain user innovation [24].OSS projects provide a collaborative platform for users through the OSS community or an innovative OSS tool [25]. As mentioned above, it is clear that user innovation matters in terms of both individual and company users. In this paper, we divide user innovation into two aspects: innovation intention and innovation capability. The former focuses on the ideology level, which explores the extent of user satisfaction with existing products and expectations regarding future products. The latter is analyzed from the perspective of user behavior [26], which emphasizes the extent to which innovative ideas are realized in practice [27].

In the project of APDPlat, users are active in giving feedback to this project and now there are in the roughly 1000 people consulting with creator of this project. The codes and documentations are getting better with the help of users. In the 12306NG project, there are many users; however, the capabilities are different and there is little contribution from them. In the VNC project, users are from all over the world, which is helpful and appreciated. On the one hand, they are active in recommending VNC to others; on the other hand, they will give valuable and technical feedback and suggestions about the project, which promotes VNC innovation greatly. In the Chandler project, because it attracted much attention among the public, many users will log on to the official website to raise questions, submit code and fix bugs, while few contribute as most of them are just interested in this project. Therefore, User innovation has a positive effect on shared knowledge.

3) Relationship between knowledge sharing and OSS project performance

An effective mechanism for knowledge sharing will help individuals to absorb data and share knowledge, while there still great potential for research on how explicit and implicit knowledge affect shared knowledge [28]. Explicit knowledge means that knowledge can be coded or explained visibly [29]. Compared with implicit knowledge, it is much easier to transfer explicit knowledge such as learning, documentation, training, interaction and so on, although it still faces some barriers, for example, it cannot be accessed because of the protection of intellectual rights. Implicit knowledge means knowledge which cannot be recorded but needs to be analyzed. The sharing of implicit knowledge is based on the participants' mutual trust which has a positive effect on innovative performance.

In the project of APDPlat, as the project provides an integrated system which simplifies comprehensive technology and platform, there are many users invited by the creator in this project to give technical support or service. The community is prosperous as well and now there are 890 developers forking the project. In the 12306NG project, there are few real contributions especially regarding the code, even though the community is prosperous. However, due to a lack of code outcome, many plans in open source have never been realized or commercialized. In the case of the VNC project, because of its technological advantages and excellent reputation among users, more and more developers are contributing. When the software is commercialized, it is very successful, and there are at least 100 users per hour from all over the world. As for the Chandler project, although the community is active, the development model is confused with proprietary software management and open source software management. A grand plan, complicated framework and slow progress have resulted in no essential outcomes in seven years. When Chandler made a great breakthrough, Gmail created by Google took up the market share. Chandler missed the best time to develop and it ended up in failure finally. Therefore, Shared knowledge has a positive effect on OSS project performance.

V. Conceptual Framework Based on Data Analysis

The performance of OSS projects is an important criterion for measuring the development of OSS projects, which is significant in promoting knowledge sharing among the participants. Usually, the performance of OSS projects can be

divided into three aspects, including market success, technology success and community success. In terms of market success, it mainly emphasizes the users' using situation. In OSS projects, users are important as well as developers [30] that are the driving force of OSS projects. Generally, it is comparatively easier to measure the developers' situation than that of users. In terms of technology success, this mainly emphasizes software quality. To some extent, some technical characteristics, such as software code, and the mechanism of the developers, can reflect the software quality [31]; therefore, module codes, alternatives, precise are all characteristics of qualified OSS projects. In the aspect of community success, this can be divided into two aspects, activeness among the participants and activeness about the project. The former focuses on the intensity of the submitted code. The value of learning will improve when there are more people and more code. The latter mainly emphasizes activeness in the discussions within the community. It will be far easier to find others' answers, or get feedback or help within a short time [32].

Based on cases analysis, Participative Motivation, Social Network, Organizational Culture from developers' side are important factors influencing knowledge sharing, meanwhile User Innovation from users' side is an unavoidable factor impacting it and then play an significant role in the performance of OSS projects. In consequence, the conceptual framework is as follows (Fig 1).

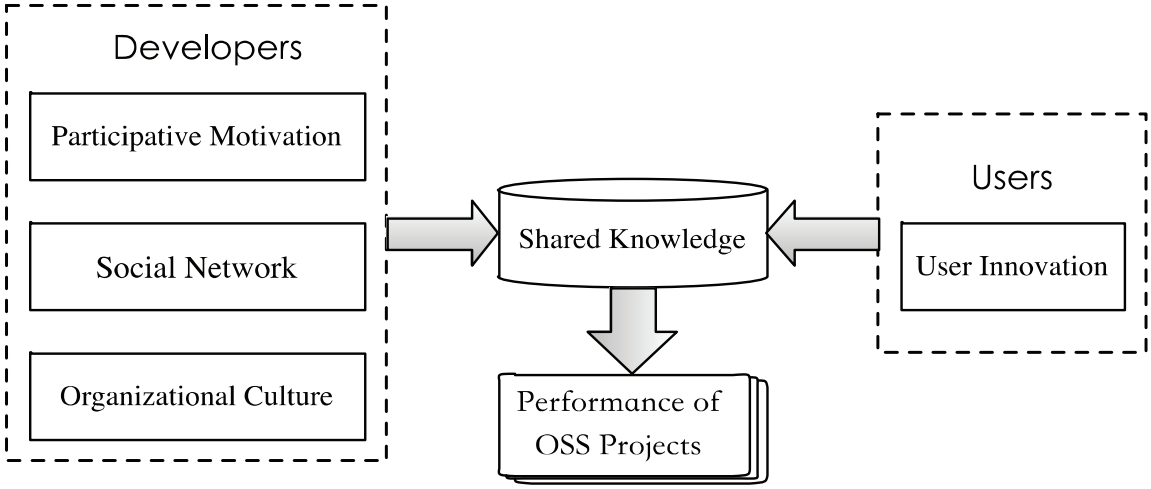

Fig. 1. Conceptual Framework of Mechanism of Knowledge Sharing in OSS Projects

VI. Conclusions and Implications

In OSS projects, distributed innovation affects the shared knowledge among the participants and then influences the performance of OSS projects (Fig 1). In the aspect of distributed innovation, developers and users are the two main bodies. Developers have an effect on shared knowledge (medium variable) through participative motivation, social networks and organizational culture (independent variables), while users will affect the shared knowledge through user innovation (external variable), and shared knowledge will influence the performance of OSS projects (dependent variable).

For the specific conceptual framework, on the developers' side, participative motivation includes intrinsic motivation and extrinsic motivation. Social networks include the cognitive dimension, relational dimension and structural dimension. The organizational culture includes openness, collaborative sharing and the geek spirit. On the users' side, innovative intention and innovation capability are important indexes. Four important variables affect shared knowledge, which includes explicit knowledge and implicit knowledge, influencing the performance of OSS projects reflecting the market success, technology success and community success.

In the development of OSS, the West pays more attention to industrial orientation. That is to say, big companies, foundations and civil senior open source people contribute a lot, but the government's role is extremely slight. The whole OSS ecological environment is mature. The Chinese government clearly pointed out the need 'to support the development and application of OSS, and quicken to form an industrial ecological system based upon the open source mode' in *The 12th Five-Year Plan of Software and Information Technology Service Industry*. However, at present, there are no obvious effects; for example, pirated software runs wild, and intellectual property is ineffectively protected. Therefore, in the field of OSS ecological environment construction, the government should pay attention to formulating fair and reasonable rules, and promoting the completeness of the relevant laws and regulations in China. Specifically, government buying, such as operational systems, basic software and other OSS, is an important policy tool of governments. In the legal system, the protection of intellectual property should be increased including the punishment of illegal behavior. University-industry collaboration with regard to OSS should be strengthened.

REFERENCES

[1] B. Perens, *The open source definition. Open sources: Voices from the open source revolution,* O'Reilly Media. 1999.

[2] S.S.Levine, M. J. Prietula. Open collaboration for innovation: principles and performance. Organization Service, 2013:1-20.

[3] R. M. Stallman. *The free software foundation management,* Free Software Foundation. 2001.

[4] M. Aberdour. Achieving quality in open source software', *IEEE Software*, 2007, 24(1): 58-64.

[5] J.H. Panchal, M. Fathianathan, 'Product realization in the age of mass collaboration', *ASME 2008 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. 2008,1(34): 219-229.

[6] E. Raymond. *The cathedral & the bazar-Musings on Linux and open source by an accidental revolutionary*, 2001, O'Reilly Media.

[7] N. J. Foss and T. Pedersen, 'Transferring knowledge in MNCs: The role of sources of subsidiary knowledge and organizational context', *Journal of International Management*. 2001,8(1): 49-67.

[8] J. N. Cummings. Work groups, structural diversity and knowledge sharing in a global organization. Management Science, 2004,50(3): 352-364.

[9] K.M. Eisenhardt. Building theories from case study research. Academy of Management Review, 1989b, 14: 532–550.

[10] R. K. Yin. Case study research: Design and methods (2nd ed.). Newbury Park, CA: Sage.1994.

[11] J. Lerner, J. Tirole. Some simple economics of open source. The Journal of Industrial Economics, 2002, 50:197-234.

[12] M. Gagné, E. L. Deci. Self-determination theory and work motivation. Journal of Organizational Behavior, 2005, 26: 331-362.

[13] K. Lakhani, R. Wolf. Why hackers do what they do: Understanding motivation and effort in free/open source software projects. Perspectives in free and open source software, MIT, Cambridge.2005.

[14] L. Hu, A. E. Randel. Knowledge sharing in teams: social capital, extrinsic incentives and team innovation. Group & Organization Management, 2014,39(2):213:243

[15] J. Nahapiet, S. Ghoshal, S.Social capital, intellectual capital and the organizational advantage. Academy of Management Review, 1998, 23:242–266.

[16] M.Friesl, S. A. Sackmann, S. Kremser. Knowledge sharing in new organizational entities: The impact of hierarchy, organizational context, micro-politics and suspicion. Cross Cultural Management: An International Journal, 2011, 18(1), 71–86.

[17] D. DeLong, L. Fahey. Diagnosing cultural barriers to knowledge management. The Academy of Management Executive, 2000, 14(7):3-4.

[18] D. R. Raban, S. Rafaeli. Investigating ownership and the willingness to share information online. Computers in Human Behavior, 2007, 23 (5):2367-2382.

[19] J. Hope. Biobazaar: the open source revolution and biotechnology. Harvard University Press, 2009.

[20] E. von Hippel. The sources of innovation. New York: Oxford University Press.1988.

[21] P. D. Morrison, J. H. Roberts, E. von Hippel. Determinants of user innovation and innovation sharing in a local market. Management Science, 2000, 46(12): 1513–1527.

[22] S. Hyysalo. User innovation and everyday practices: Micro-innovation in sports industry development. R&D Management, 2009, 39(3): 247–258.

[23] I. Alam. Removing the fuzziness from the fuzzy front-end of service innovations through customer interactions. Industrial Marketing Management, 2006, 35(4):468–480

[24] M. Bogers, A. Afuah, B. Bastian. Users as innovators: A review, critique, and future research directions. Journal of Management, 2010, 36(4):857–875.

[25] E. von Hippel, R. Katz. Shifting innovation to users via toolkits. Management Science, 2002, 48(7): 821–834.

[26] G. Bin. A reasoned action perspective of user innovation: Model and empirical test. Industrial Marketing Management, 2013, 42:608-619.

[27] C. Lüthje, C. Herstatt, E. von Hippel. User-innovators and "local" information: The case of mountain biking. Research Policy, 2005, 34:951–965.

[28] J.C. Pai. An empirical study of the relationship between knowledge sharing and IS/IT strategic planning. Management Decision, 2006, 44(1):105-122.

[29] M. Polanyi. Personal Knowledge, Chicago: The University of Chicago Press, 1962.

[30] E. E. Kim. An introduction to open source communities. Blue Oxen Assocites, 2003.

[31] G. F. Lanzara, M. Morner. Artifacts rule! How organzing happens in open source software projects. In: Czarniawska B, Hernes T (eds) Actor-network theory and organing. Liber and Copenhagen Business School Press, 2005.

[32] S. L. Toral, M. R. Martínez-Torres, F. Barrero. Analysis of virtual communities supporting OSS projects using social network analysis. Information and Software Technology, 2010, 52 (3): 296-303.