CISIS 2011

The Fifth International Conference on

Complex, Intelligent and Software Intensive Systems

Korean Bible University, Seoul, Korea
30 June - 2 July 2011

Edited by
Leonard Barolli, Fukuoka Institute of Technology, Japan
Fatos Xhafa, Technical University of Catalonia, Spain
Ilsun You, Korean Bible University, Korea
Nik Bessis, University of Derby, UK

Published by
IEEE computer society

Papers by Session

S5: E-Learning and Groupware Systems

2011 International Conference on Complex, Intelligent, and Software Intensive Systems

Analysis of the Core Team Role in Open Source Communities

M. R. Martínez Torres
Business Administration Department
University of Seville
Seville, Spain
rmtorres@us.es

S. L. Toral, M. Perales, F. Barrero
Electronic Engineering Department
University of Seville
Seville, Spain
toral@esi.us.es, perales@gte.esi.us.es,
fbarrero@esi.us.es

***Abstract*—Open source software represents a new paradigm of software development based on a subjacent community. It is widely accepted in the literature the layered structure of open source communities, being the core group the most active contributors usually located at the center of the community. The tasks of this group include not only an intense activity in terms of contributions but also to promote participation among the rest of the community members. In this paper, the general role of this group is analyzed by modeling communities as Social Networks and applying Social Network Analysis techniques. Findings related their brokerage activity with open source software success.**

Keywords-wireless Open source communities; Virtual communities; Social Network Analysis; brokerage role.

I. INTRODUCTION

Open Source Software (OSS) projects constitute a prominent example of software development based on communities [1]. They are supported by a community of developers and users who work in geographically distributed locations, rarely or never meet face to face, and coordinate their activities making an intense use of Internet.

The wide success of several well known Open Source Software (OSS) projects has attracted much attention. Software engineering researchers and commercial companies alike have been trying to learn lessons from the success of OSS and apply some of them to the development of proprietary software [2]. However, there is a cruel reality that the vast majority of OSS projects fail to take off and soon become abandoned [3]. According to the popular open source portal, SourceForge (http://sourceforge.net/), most OSS projects have ended in failure: 58% do not move beyond the alpha developmental stage, 22% remain in the planning phase, 17% remain in the pre-alpha phase, and some become inactive [4]. Previous studies on OSS have been mainly focused on motivation of people participating in subjacent virtual communities. Motivations have been found to range from problem solving [5] [6], to reputation benefits and career opportunities [7] [8] [9]. But a more reduced amount of work has been focused on communities' structure. It has been demonstrated that much of the OSS development is realized by a small percentage of individuals despite the fact that there are tens of thousands of available developers. Such concentration is called "participation inequality" [10], [11], and it can be explained by the different user profiles of open source communities. Consequently, the structure of OSS communities is not completely flat as it was claimed by the bazaar model of full participation [12].

II. OSS COMMUNITIES STRUCTURE

OSS communities are typically initiated by an individual (or group of individuals) who provides systems and development components, or their access, as well as communication infrastructure. Participants are usually volunteers and contributors are not normally motivated by traditional economic incentives, but rather by instrumental factors associated with fulfilling a need, and by intrinsic factors such as enhanced reputation, expertise development (learning), self-fulfillment, as well as basic fun and enjoyment [13], [14]. The individuals that participate in open source software projects are often described as comprising a community. The influences that members have on the system and the community are different, depending on the roles they play. Participation inequality allows the categorization of OSS community members into three basic groups [15], [16]:

- Core members. They are responsible for guiding and coordinating the development of an OSS project. They are usually involved with the project during a long period of time and have made significant contributions to the development and evolution of the system. Moderators and leaders are included in this group.
- Active developers. They regularly make contributions to the project.
- Peripheral developers. They occasionally contribute with new features to the existing system. This contribution is irregular, and the period of involvement is short and sporadic. Free riders (people who just are seeking answers without making any contributions) are also included in this group.

Typically, new members are attracted to an OSS community because the system can solve one of their own problems. But the community offers the possibility of migrating from being a passive user to an active user by a process called Legitimate Peripheral Participation [17]. This is the process by which a newcomer is integrated into the community. New members learn how to function as a community member through participation, and acquire the language, values, and norms of the community. Learning is gradually achieved as an individual moves from being a

978-0-7695-4373-4/11 $26.00 © 2011 IEEE
DOI 10.1109/CISIS.2011.25

novice, gaining access to community practices to complete socialization and therefore becoming an insider or full member of the community. The more contributions they make, the higher recognition they earn, and finally, they can enter the highly selected inner circle of Core Members. Not all the members want to become a core member. The majority of them remain as passive users or somewhere in the middle. But the community offers the possibility of increasing the level of expertise through continuous contributions and interaction with other community members. On the other hand, OSS communities must be able to regenerate themselves to survive. New active and core members must emerge from the mass of passive users. Otherwise, the development of projects will stop when current active contributors leave. Because all OSS developers are volunteers who are not bound by any kind of formal contracts, they may leave at any moment for various reasons [2]. In this sense, core developers are called to play an important role, not only on the development of the system itself, but on the creation and maintenance of a dynamic and self-reproducing OSS community. Some previous works have analyzed the evolution of the core team [18], their influence on the success of the underlying project [19], [20] or their role as brokers of knowledge [21], [22]. In this paper, the activity of the core team is analyzed taking into account not only their degree of participation but also their role as brokers respect to active and passive users. For this purpose, mailing lists have been considered as they allow the collective reflection and community discussions, and activities are not just confined to software development or coding alone [21].

III. METHODOLOGY AND DATA

The activity of the core group of developers has been analyzed using social network analysis techniques. OSS community is represented as a social network: a graph G = (V,E) where V denotes a finite set of vertices and E denotes a finite set of edges such that E V × V. In the context of mailing lists, they are usually structured following threads of discussion. Threads are groups of messages sharing the same subject. A thread is initiated by someone who posts a message asking for help, suggesting some improvements, or just considering some new idea. Then people start answering this initial message, posting possible solutions, sources of information or just extending posted considerations. Some members of the community become engaged in a process of conceptualization, leading to some collective innovation and new knowledge. The result is a list of related messages where the sequence of reflections is detailed, so newcomers can follow expert reasoning step by step. When modeling OSS communities as social networks, V is given by all the authors posting messages and E is given by the successive answers among authors inside a thread, which is the basic unit considered [23]. In particular, an author posting to a thread is considered to be tied to all the authors who have previously posted to the same thread when constructing the social network. The use of discussion threads as the basic unit of analysis is highly valid, considering that the epistemic interactions in support of OSS development often take place in discussion threads where individual postings provide the context to encourage participation [10]. In contrast to a reply to a single message, it is more cognitively complex to reply to a threaded discussion, because the ebb and flow of earlier postings must be taken into account to develop a coherent answer [20], [24]. Figure 1 illustrates the representation of the Debian-amd64 community during 2006.

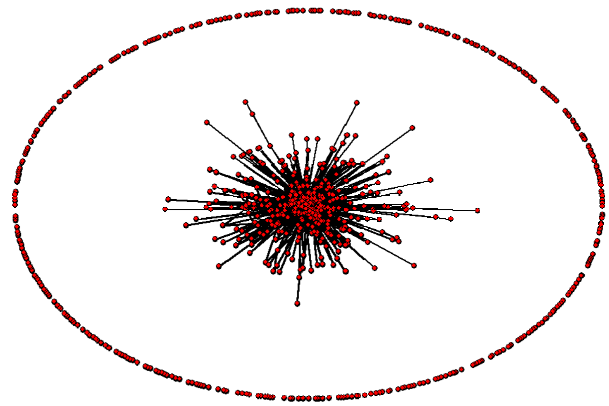

Figure 1. Debian-amd64 community members in 2006.

In this particular case, the community is integrated by 941 users (vertices in Figure 1), but just 489 are active members (located at the center of Figure 1) while the rest of them corresponds to free riders (located at the periphery of Figure 1)

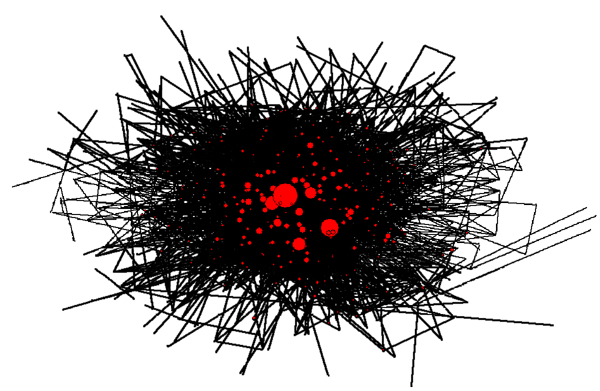

Figure 2. Debian-amd64 community interactions in 2006.

In Figure 2, vertices of the same community are represented proportional to the number of interaction in which they are involved. It can be clearly observed the participation inequality phenomenon: most of contributions are due to a reduced number of members.

Once communities are modeled as social networks, several SNA features can be extracted:

- Size and interactions: the number of vertices represents the number of community members and the arcs represent the interactions among them.
- Density: it is defined as the number of lines in a simple network, expressed as a proportion of the maximum possible number of lines. The main problem of this definition is that it does not take into account valued lines higher than 1 and it depends on the network size. A different measure of density is based on the idea of the degree of a node, which is the number of lines incident with it [19]. A higher degree of nodes yields a denser network, because nodes entertain more ties, and the average degree is a non-size dependent measure of density. Finally, density can be measured alternatively using an egocentric point of view; the egocentric density of a node is the density of ties among its neighbors [25].
- Brokerage roles: A broker is a middle node in a directed triad (a set of three vertices and the lines among them). Different types of brokerage roles can be distinguished considering mediation between members of the same or different groups. In this context of OSS communities, these groups are given by active member and free riders. Therefore, two possibilities of mediation can be considered as shown in Figure 3.

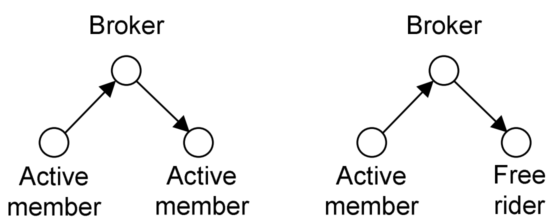

Figure 3. Brokerage roles of a vertex.

- Distance: it is defined as the number of steps in the shortest path that connect two vertices. In the case of OSS communities, the most relevant vertices are those corresponding to the core team of developers. Consequently, it makes sense to measure the distance of to these vertices.
- Closeness centralization: it is an index of centrality based on the concept of distance. The closeness centrality of a vertex is calculated considering the total distance between one vertex and all other vertices, where larger distances yield lower closeness centrality scores. The closeness centralization is an index defined for the whole network, and it is calculated as the variation in the closeness centrality of vertices divided by the maximum variation in closeness centrality scores possible in a network of the same size [19].
- Betweenness: it is a measure of centrality that rests on the idea that a person is more central if he or she is more important as an intermediary in the communication network [25]. The centrality of a vertex depends on the extent to which this node is needed as a link to facilitate the connection of nodes within the network. Then, they are said to develop a brokerage role. If a geodesic is defined as the shortest path between two nodes, the betweenness centrality of a vertex is the proportion of all geodesics between pairs of other vertices that include this vertex, and betweenness centralization of the network is the variation in the betweenness centrality of vertices divided by the maximum variation in betweenness centrality scores possible in a network of the same size. This measure allows detecting gateways connecting separate sub networks [26].

IV. RESULTS

Several Debian Linux ports to embedded processors and environments have been considered to be analyzed using SNA techniques. The Debian Project is an association of individuals who have made common cause to create a free operating system called Debian GNU/Linux, or simply Debian for short [27], [28]. Table 1 details the list of considered communities.

Table 1. OSS projects considered in the case study.

Community	Description	Years
Debian port to m68k (D-68k)	Motorola 68k port of Debian GNU/Linux. Debian currently runs on the 68020, 68030, 68040 and 68060 processors	98-09
Debian port to ARM (D-ARM)	ARM port for Debian GNU/Linux. Debian fully supports a port to little-endian ARM	99-09
Debian port to ia64 (D-IA64)	IA64 port of Debian GNU/Linux	01-09
Debian port to Alpha (D-Alpha)	The purpose of this project is to assist developers and others interested with the ongoing project to port the Debian distribution of Linux to the Alpha family of processors.	98-09
Debian port to amd64 (D-AMD64)	Porting Debian to AMD x86-64 architecture	04-09
Debian port to BSD (D-BSD)	This is a port of the Debian operating system, complete with apt, dpkg, and GNU userland, to the NetBSD kernel.	01-04
Debian port to HPPA (D-HPPA)	This is a port to Hewlett-Packard's PA-RISC architecture.	01-09
Debian port to Hurd (D-HURD)	The GNU Hurd is a totally new operating system being put together by the GNU group.	99-09
Debian port to MIPS (D-MIPS)	MIPS port of Debian GNU/Linux, able to run at both endiannesses	99-09

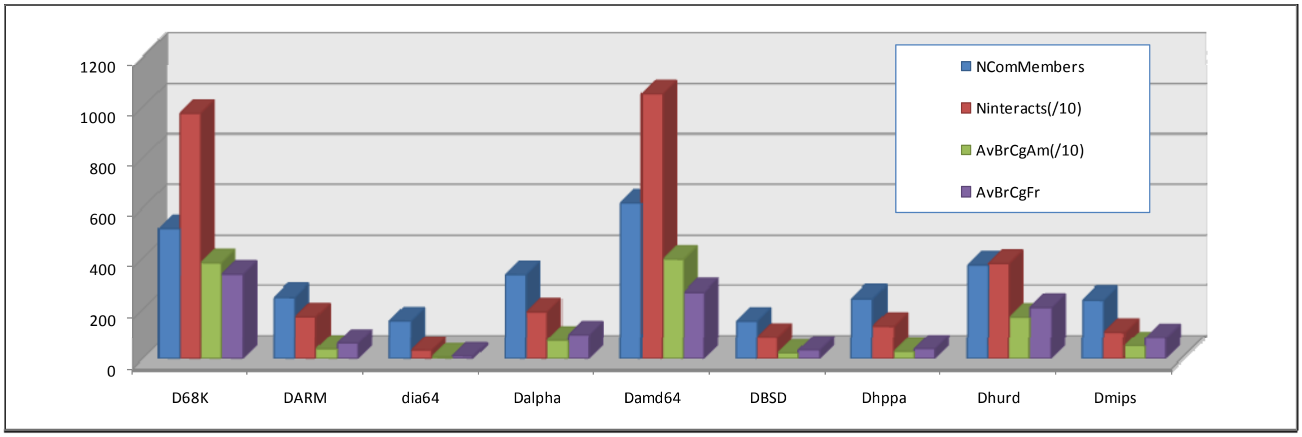

Figure 5. Community success and core member brokerage role with active members and free riders

A social network has been obtained for each one of the years listed in the third column of Table 1. The initial year for each community is the year in which the community exhibited a certain level of activity during the twelve months of the year. As a result, 90 social networks have been analyzed obtaining a set of SNA features.

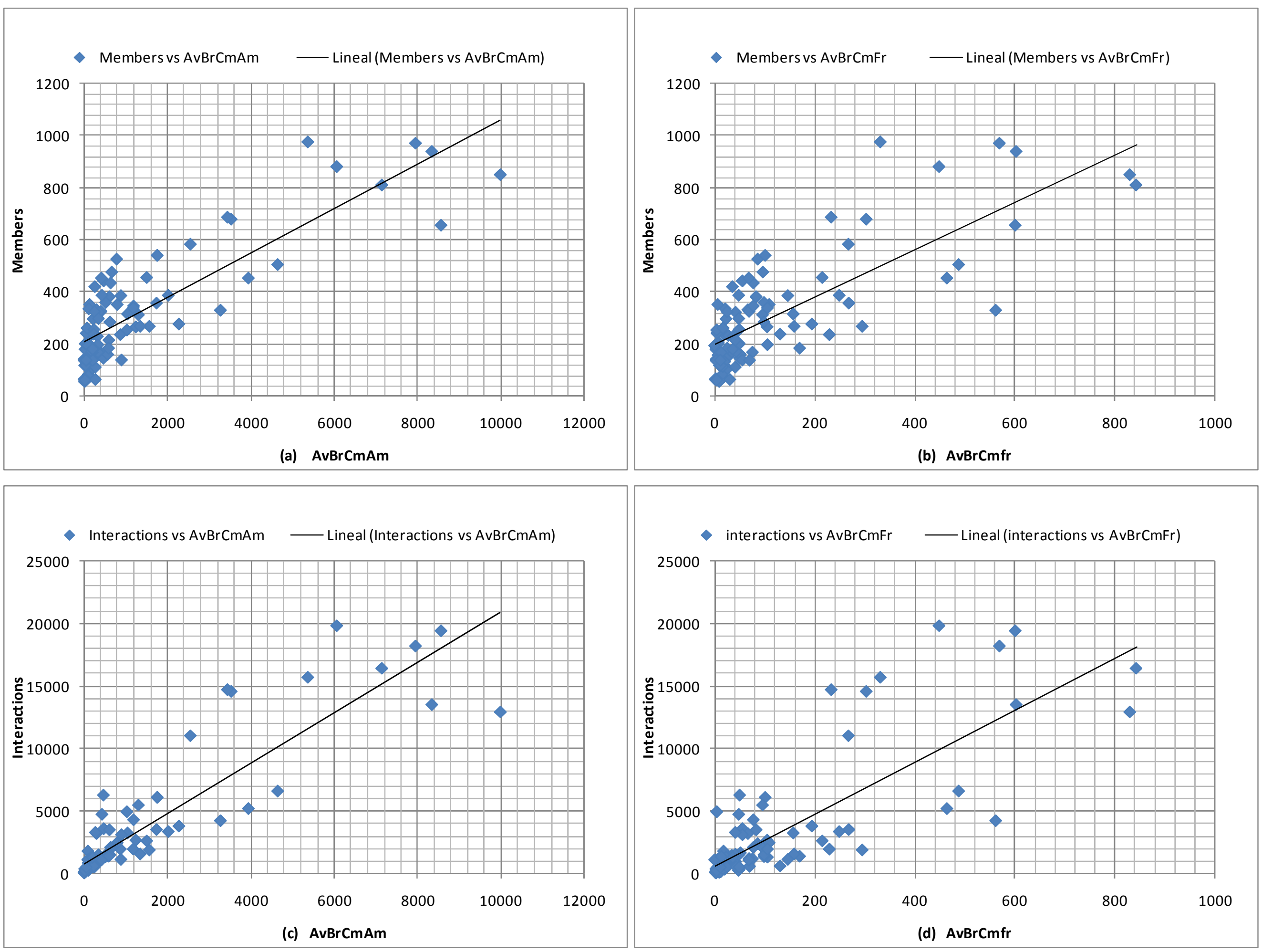

Figure 4.Correlations found: (a) Members vs AvBrCmAm; (b) members vs AvBrCmfr; (c) interactions vs AvBrCmAm; (d) interactions vs AvBrCmfr

Success in a virtual community could be manifested through the level of participation, which can be understood as the number of participants and the number of messages posted in the community [29]. In this case, the average value of community members (*NComMembers*) and number of contributions (*Ninteracts*) are taken as a measure of a community success (see Figure 5). The topology of the network is given by indicators such as centrality and betweeness.

On the other hand, the layered structure of virtual communities has been extracted using the out-degree value as a discriminant criterion [22]. In general, vertices' out-degree follows a power-law distribution so the core group has been selected as members responsible of more than 50% of contributions. However, the mission of the core group is not just posting the majority of messages, but also acting as brokers or mediators among users. As a difference to previous works, it has been distinguished two types of brokerage roles: a brokerage role between active members (*AvBrCgAm*) and a brokerage role between active members and free riders (*AvBrCgFr*). Their average values have been included in Figure 5 as two more indicators. Notice that those communities where the brokerage role of active members is devoted to active members are wealthier: If the brokerage role of core members is distributed more equally between active members and free riders, the contributions are less than ten times the number of members.

In Table 2, the correlations of some indicators are measured. The first two indicators are the average brokerage roles developed by core members among active members and among active members and free riders, respectively. Table 2 shows that the maximum correlation value is obtained between the average brokerage roles developed by core members among active members (AvCmBrAm) and the number of interactions. This result highlights the importance of the brokerage role played by the core members, and how this role can make the community to grow when this role is focused on active members.

The next three indicators are related to the distance between core members and the rest of community members. Distance-1 refers to the average number of core neighbors, considering neighbors as those nodes directly linked with core nodes (accessible through one step). Distance-all refers to the average number of core neighbors, considering neighbors as those nodes directly or indirectly linked with core nodes (that is, in one or more than one step). Finally, AvSteps is the average number of steps in which community members linked with the core group can be accessed. The distance related indicators measure the extent to which the community is accessible and can be controlled by the core group. The AvSteps indicator measures the dispersion of the community respect the core group of developers. A high value of AvSteps means an indirect connection between core group and the majority of community members. Table 2 results show that community success is promoted as the community is concentrated around the core group of developers. The activity of the core group also promotes a centralized topology of the network, both in terms of distance (centrality) and mediation (betweeness).

Table 2. Correlations found

	Interacts	Members	Centra-lity	Bet-weeness
AvCmBrAm	0,90*	0,84*	0,42*	0,54*
AvCmBrFr	0,79*	0,77*	0,41*	0,57*
Distance-1	0,85*	0,82*	0,46*	0,61*
Distance-all	0,94*	0,93*	0,45*	0,57*
AvSteps	0,32*	0,35*	0,10*	0,22*

* Correlation is significant at the 0,01 level (2-tailed).

In Figure 4, the correlations are plotted, showing the dispersion of dots in parts (a), (b) and (d), and a better linearity correlation in case (c).

V. CONCLUSION

This paper highlights the role of the core group in virtual communities. For this purpose, virtual communities have been modeled as social networks, extracting the core group by means of the out-degree as a discriminant criterion. Several Social Network Analysis indicators have been obtained to describe the behavior of the core group. Findings emphasize the mediation activity that must be developed by the core group, especially in the case of active members. From a topological point of view, the success of the underlying software is also affected by the extent to which community members are linked to the core group.

ACKNOWLEDGMENT

This work has been supported by the Consejería de Innovación, Ciencia y Empresa (Research Project with reference P07-TIC-02621).

REFERENCES

[1] S. K. Shah, Motivation, governance, and the viability of hybrid forms in open source software development, Management Science, Vol. 52, no. 7, pp. 1000-1014, 2006.

[2] Y. Ye, K. Kishida, Toward an Understanding of the Motivation of Open Source Software Developers, Proceedings 25th International Conference on Software Engineering, 2003, pp. 419-429.

[3] K. Ngamkajornwiwat, Z. Dongsong, A. G. Koru, L. Zhou; R. Nolker, An Exploratory Study on the Evolution of OSS Developer Communities, Proceedings of the 41st Annual Hawaii International Conference on System Sciences, pp. 305-315, 2008.

[4] S.-Y. Tom Lee, H.-W. Kim, S. Gupta, Measuring open source software success, Omega, Vol. 37, pp. 426-438, 2009.

[5] J. Feller, B. Fitzgerald, Understanding Open Source Software Development, Addison-Wesley, Boston, 2002.

[6] S. Shah, Motivation, governance and the viability of hybrid forms in open source software, Management Science, Vol. 52, no. 7, pp. 1000–1014, 2006.

[7] J. Lerner, J. Tirole, Some simple economics of open source, Journal of Industrial Economics, Vol. 5, pp. 197–234, 2002.

[8] A. Bonaccorsi, C. Rossi, Why open source software can succeed? Research Policy, Vol. 32, no. 7, pp. 1243-1258, 2003.

[9] J. Roberts, I.-H. Hann, S. Slaughter, Understanding the motivations, participation, and performance of open source software developers: a

longitudinal study of the Apache projects, Management Science, Vol. 52, no. 7, pp. 984–999, 2006.

[10] G. Kuk, Strategic Interaction and Knowledge Sharing in the KDE Developer Mailing List, Management Science, Vol. 52, no. 7, pp. 1031–1042, 2006.

[11] S. L. Toral, M. R. Martínez-Torres, F. Barrero, F. Cortés, An empirical study of the driving forces behind online communities, Internet Research, Vol. 19, no. 4, pp. 378-392, 2009.

[12] E. Raymond, The Cathedral and the Bazaar, O'Reilly, Sebatopol, CA, 1998.

[13] J. Lerner and J. Tirole, Some simple economics of open source, Journal of Industrial Economics, Vol. 50, no. 2, pp. 197-234, 2002.

[14] E. Von Hippel and G. von Krogh, Open source software and the "private-collective" innovation model: issues for organization science, Organization Science, Vol. 14, no. 2, pp. 209-223, 2003.

[15] A. Mockus, T. Fielding, and D. Herbsleb, Two Case Studies of Open Source Software Development: Apache and Mozilla, ACM Trans. Software Eng. and Methodology, Vol. 11, no. 3, pp. 309-346, 2002.

[16] J. Xu, Y. Gao, S. Christley, and G. Madey, A Topological Analysis of the Open Source Software Development Community, Proceedings of the 38th Annual Hawaii International Conference on System Sciences, HICSS '05, 188-198, 2005.

[17] J. Lave and E. Wenger, Situated learning: Legitimate peripheral participation. Cambridge University Press, 1991.

[18] G. Robles, J. M. Gonzalez-Barahona, I. Herraiz, Evolution of the core team of developers in libre software projects, 6th IEEE International Working Conference on Mining Software Repositories, MSR '09, 2009.

[19] S. L. Toral, M. R. Martínez-Torres, F. Barrero, F. Cortés, An empirical study of the driving forces behind online communities, Internet Research, Vol. 19, no. 4, pp. 378-392, 2009.

[20] S. L. Toral, M. R. Martínez-Torres, F. Barrero, Virtual Communities as a resource for the development of OSS projects: the case of Linux ports to embedded processors, Behavior and Information Technology, Vol. 28, no. 5, pp. 405-419, 2009.

[21] S. Sowe, I. Stamelos, and L. Angelis, Identifying knowledge brokers that yield software engineering knowledge in OSS projects, Information and Software Technology, Vol. 48, no. 11, pp. 1025-1033, 2006.

[22] S. L. Toral, M. R. Martínez Torres, and F. Barrero, Analysis of Virtual Communities supporting OSS Projects using Social Network Analysis, Information and Software Technology, Vol. 52, no. 3, pp. 296-303, 2010.

[23] G. Jones, G. Ravid, and S. Rafaela, Information overload and virtual public discourse boundaries, In Proceedings of Eighth IFIP Conference on Human-Computer Interaction, Tokyo, Japan, 2001.

[24] N. Knock, Compensatory adaptation to a lean medium: An action research investigation of electronic communication in process involvement groups, IEEE Trans. on Professional Communication, Vol. 44, no. 4, pp. 267–285, 2001.

[25] W. Nooy, A. Mrvar, & V. Batagelj, Exploratory Network Analysis with Pajek, New York, Cambridge University Press, 2005.

[26] C. Faba-Pérez, F. Zapico-Alonso, V. P. Guerrero-Bote, and F. Moya-Anegón, Comparative analysis of webometric measurements in thematic environments, Journal of the American Society for Information Science and Technology, Vol. 56, no. 8, pp. 779–785, 2005.

[27] G. Robles, J. M. Gonzalez-Barahona, and M. Michlmayr, Evolution of volunteer participation in libre software projects: Evidence from Debian, Proceedings of the First International Conference on Open Source Systems, Genova, 100-107, 2005.

[28] J. Mateos-Garcia, & W. E. Steinmueller, The institutions of open source software: Examining the Debian community, Information Economics and Policy, Vol. 20, pp. 333–344, 2008.

[29] J. Preece, Sociability and usability in online communities: determining and measuring success. Behaviour & Information Technology, Vol. 20, Iss. 5, pp. 347–356, 2001.