Weighting Structures: Evolutionary Dynamics of Innovation Networks in Virtual Communities

Vitaliano Barberio[1] and Alessandro Lomi[1,2]

[1] Department of Management Sciences, University of Bologna, Department of Management,
via Capo di Lucca 34, 40100 Bologna, Italy
[2] Faculty of Economics, University of Lugano, Switzerland
{vitaliano.barberio}@gmail.com

Abstract. We discuss and illustrate alternative analytical strategies for representing coordination networks in innovative virtual communities bounded by production relationships among participants. We use information on email communication networks reconstructed using data from the Apache Open Source project to give empirical contents to our arguments and to substantiate our claims that: (i) Self-organizing networks provide the basic principles of coordination in such communities; (ii) Once in place, deliberate governance arrangements affect coordination patterns within virtual communities; (iii) Structural properties of communication networks change significantly over time depending on their internal organizational logics, and (iv) Affiliation (a.k.a. two mode) networks provide a useful representation for detecting community structures.

Keywords: distributed work, coordination, communication networks, open source development.

1 Introduction

Recent years have seen the emergence of new conceptual models of innovation which rely on ICT mediated communication to coordinate production and exchange activities. Such models tend to assign a rather limited role to formal governance mechanisms that are viewed as restricted in scope (to regulated tasks) and time (adoption in advanced stages of growth). These two simple assumptions are of great relevance for the study of organizational dynamics of innovation.

When the knowledge needed to generate innovation is both complex and distributed across different organizations or units [1], network partners and institutions affecting patterns of exchange become of central importance for our understanding of innovation processes. For example, [2] argued that successful teams (X-teams) within organizations, today are characterized by porous boundaries and fluid membership allowing organizations to reach the knowledge they need to sustain high innovation rates over the time.

In order to achieve some collective objective organization members are supposed to rely on some form of shared knowledge. When such knowledge is distributed around a community of interacting actors, interdependent tasks typically require some information exchange in order to be coordinated [3]. In the project management literature, for

A.A. Ozok and P. Zaphiris (Eds.): Online Communities, LNCS 5621, pp. 295–304, 2009.

example, information exchange has been argued to map a kind of 'state of the world awareness' to sequences of 'possible actions' [4]. Then, both the amount and structure of known information determine the complexity[1] of decisional processes for project teams in order to perform tasks. Literature on problem solving [5,6] proposed that modular (decomposable in near independent parts) and barely formal organizational architectures should display an evolutionary advantage, when compared with more traditional ones, in complex and fast changing environments.

Despite this recent recognition of the advantages for both distributed and networked models of innovation, comparatively little attention has been paid to communication patterns. We think of communication as an important means of coordination, an enacted structure that links interdependent tasks[2] to be performed [7], [8] in order to 'feed' innovation processes. We also think that Open Source Software (hereafter OSS) projects could be a perfect empirical setting to both develop and test a reflection on distributed organizational dynamics. In this paper we will focus on email communication intended as the main means of coordination for distributed development in a successful OSS project.

We will discuss general issues about the dynamics of organizational structure, and the adequacy of available analytical strategies for detecting it and represent its change over the time. More precisely we want to explore four issues: (i) the evolution of information exchange structures defined in terms of communication networks; (ii) the impact of adopting formal governance arrangements on communication structures; (iii) the usefulness of direct communication networks as a basis for networks decomposition, and (iv) the detection of community structures in communication networks, based on the dual association between programmers and mailing list [9].

To give empirical content to our argument, we collected communication data for our analysis from the mail archive of the Apache Open Source project. Our data cove a ten year time period – from 1995 to 2004. Beginning in 1995, the Apache community created and maintained, over the years, the most widely implemented web server software in the world. The second part of this paper consists of an analytical development of our perspective on the endogenous organizational dynamics of communication and coordination. Each of our four general arguments is introduced by an abridged survey on the state of the art in OSS literature. Then each issue is developed by means of network analytical tools and results are discussed. The paper concludes with research questions that could further extend and strengthen the preliminary results presented.

2 Discovering and Representing Structure

For our analysis we computed simple structure indicators for two kinds of networks linking participants to the Apache community: (i) direct communication networks; and (ii) affiliation networks. The first is a one-mode social network in the sense that it

[1] A complex system of whatever nature (natural, social or symbolic) is intended here as one which is made of a large number of simple but interdependent component parts.

[2] The paper by Cataldo et al. (2006) is a notable exception, linking technical interdependencies to coordination requirement to actual coordination by means of email communication, over the time. This paper is developed in a commercial software development empirical setting.

records activities of relational exchange among individuals. In the second network individuals are connected through their dual association with mailing lists to which they contribute.

Direct communication networks are intended here as networks whose nodes are community members and whose links exist between two nodes when an agent (developer) sent a message in-reply-to another message by another agent (developer). Links were weighted using the number of exchanged messages among dyads of agents. Affiliation networks are built with two kind of nodes – programmers and mailing lists – and nodes of one type only connect with nodes of the other type.

For analysis in both points 2.1 and 2.3 we generated random graphs (normal distribution of nodal degrees) in order to compare them with actually observed networks. For analysis in paragraph 2.3 we just operated a network reduction cutting lines and nodes under a given threshold of connectivity. For analysis in paragraph 2.4 we fist, folded two-mode networks by multiplying original matrices (algebraic representation of networks with nodes of type one on rows and nodes of type two on columns) for transposed ones (which have the same kind of nodes on rows and columns). In this way we obtained one mode networks, weighted for the number of shared nodes (of the other type). On these networks we used Newman clustering algorithm for finding community structures [10]. This algorithm hierarchically decomposes networks in sub component progressively removing nodes with highest betweenness centrality [11].

2.1 Evolutionary Dynamics of Communication: Scale-Free Networks and Self-organization

The diagrams reported in Figure 1 show the evolutionary trajectory of such 'direct communication' networks for the Apache project over ten years time period 1995-2004. Here below we offer some simple metrics on networks over the time (chart 1). The number of nodes increased from 28, in 1995 to 6353 in 2004. The number of edges (network ties) increased from 38 to 16100 during the same time period. The components count was 1 in 1995 and grew to 113 in 2004. In Figure 2 we can see how both network degree centralization (average centrality for the overall network) and network density decreased by one and two orders of magnitude respectively over the observation period.

Weiss and colleagues [12] studied the degree (number of lines incident to a vertex) distribution for the Apache email archive finding that only few developers held a high value (degree distribution follows a power law). They also controlled for the existence of the so called preferential a attachment phenomenon according to which over the time more connected nodes are more likely to become even more connected than others (rich gets richer).

Our analysis confirms this results and we also controlled for both the clustering coefficient values and average distance values over time [13]. Both average distance and clustering coefficient values, were higher then the correspondent values in random networks with the same density and number of nodes (see figure 3). This result could be interpreted as the overall network holding a scale-free topology [14]. It has already been shown [14] how scale-free networks could be generated from an initial network according to a variety of self-organization mechanisms. A well studied mechanism is preferential attachment. If over the time new nodes attach themselves to others

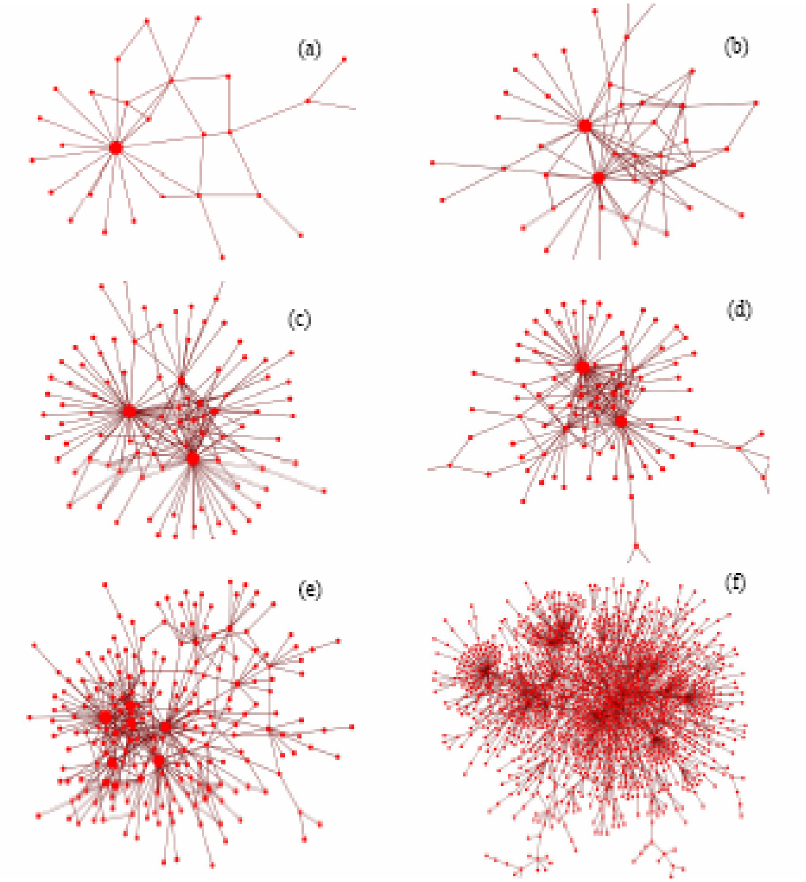

Fig. 1. Six snapshots representing the evolution of direct communication networks over a ten years time period. **(a)** 1995; **(b)** 1996; **(c)** 1997 ; **(d)** 1998; **(e)** 1999; **(f)** 2000.

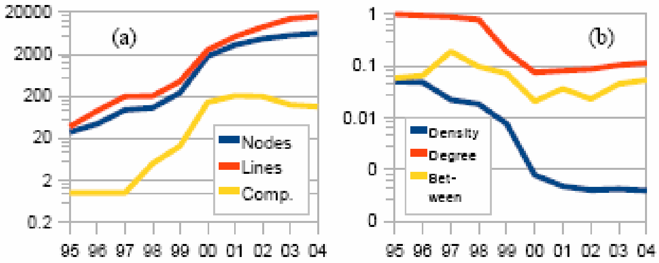

Fig. 2. (a) Node count, edge count and component count (Y axes in logarithmic scale); **(b)** Density, Degree Centralization and Betweenness Centralization (Y axes in logarithmic scale)

according to the simple 'preference' for already highly central others a scale-free network will obtain.

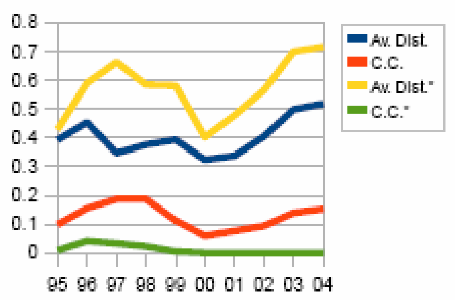

Fig. 3. Original networks. Both the average distance and the clustering coefficient for the real networks are over the values for the random generated networks.

This concept, that literally means the emergence of organizational structures in absence of central planners, seems to be of a certain interest for the study of virtual communities. In economics it has been argued that this behavior could be explained by the signaling incentives for individual programmers on the labor market [15, 16, 12] it.

2.2 Enters Design: The Impact of Formal Institutions on Communication Networks

Self-organization in virtual communities is an obviously important – if emergent - coordination mechanism. However, recent research has shown that in successful fast growing projects self-organization needs to be balanced by formal governance arrangements (or “institutions”) designed to affect the development process in desirable directions [17].

According to the James Thompson’s influential statement, technological interdependencies may assume three basic forms. Arranged in an increasing degree of complexity the forms of interdependence are: (i) pooled; (ii) sequential, and (iii) reciprocal [18]. According to Thompson, organizational structures should be designed in order to cope with the different degrees of complexity coming with task interdependencies to be coordinated.

Basing on our initial assumption that in virtual communities coordination should be mirrored by communication patterns, we expected that those patterns would change after the design and implementation of formal governance arrangements. More precisely, the formalization of organizational structures, which in our case study could be intended as the creation of the Apache Software Foundation (ASF) in 1999, would lead coordination toward simpler forms.

In order to explore this argument, we counted in direct communication networks over time, how many patterns were corresponding to Thompson's interdependencies, as percentage of the total number. As showed in figure 4, our expectations are confirmed because: on the on hand, both 'pooled' and 'sequential' interdependencies tend (on average) to increase before 2000, while they tend to diminish after that time; on the other hand, reciprocal interdependencies – the more complex type – increased before 2000 and diminished after that time.

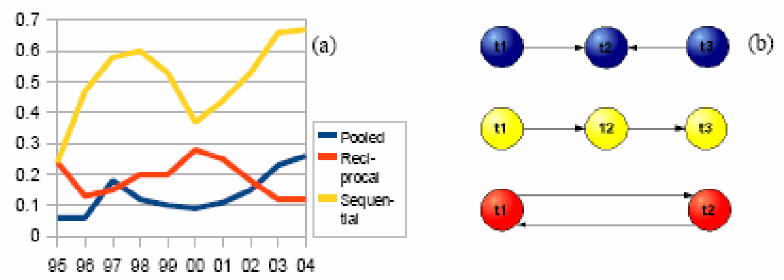

Fig. 4. (a) Thompson's interdependencies count in direct communication networks measured as percentage of the total number of links among nodes; **(b)** Interdependency shapes

2.3 The Adequacy of 'in-Reply-to' Built Networks to Network Decomposition

Another point that should moderate the extent of findings on self-organizing dynamics in OSS projects is about coordination and division of labor in large communities. Early literature on OSS development highlighted how small groups of developers actually accounted for writing the most of software code in Apache and Mozilla [19], and Gnome [20].

This studies also found that the number of contributors who fixed bugs was one order of magnitude higher than the number of those who wrote the code [19]. It seems that, when looking at productivity, large communities display core-periphery structures [21] and nested layers of roles [22]. So we ask here: what happens to communication networks when we just consider the core of interaction processes? In order to explore this issue, we assumed that the more a developer writes code the more hi will use email communication in order to coordinate his actions with other community members. Then, we applied a simple cut (lines and nodes) reduction on our direct-communication networks.

This means that we removed from networks that lines with a value lower than a given threshold (say 3 exchanged emails) and then we removed which those nodes that resulted to have a total degree (in + out) less than 1 (say isolate nodes). The results of this procedure are showed in figure 5. The so reduced network 'captures' on average (over the time) the 21% of nodes, the 17% of lines and the 63% of components.

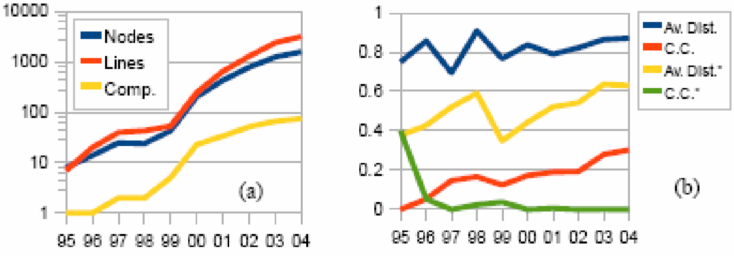

Fig. 5. (a) Nodes, lines and components after net. reduction with cut-threshold = 3; **(b)** Clustering coefficient and average distance, cut-threshold = 3. Values marked with * refers to random generated networks.

It is also to notice that in reduced networks the density is on average the 27% higher than in the original networks. We interpret tis result as a higher connectivity among more active (core) members of the community. It is also to notice that networks, whose links were created using the in-reply-to filed on email headers, are very sensible to cut-like method of reduction.

When we look at values from the reduced networks at least two things are to notice: first, the clustering coefficient is monotonically growing (figure 5.b.) instead of floating (figure 5.a.); second, the values of average distance for real networks is higher than the correspondent values for random generated networks. Combining these two findings we could say that core members tend, over the time, to form clusters which are characterized by high inbound connectivity and low outbound connectivity.

2.4 Modular Architectures, Affiliation Networks and Newman Clustering

The findings in paragraph 2.3 made us thinking about another strand of organizational literature on modular structures[3] on OSS projects whose major claim is that coordination patterns should mirror technical interdependencies [23]. Because software has a more modular architecture than more traditional products have, the organization that produces it should have a modular structure as well.

On the one hand, it is very reasonable to assume, coherently with modularity theory [24], that a programmer working in a peripheral module probably just knows very little about what the development concerns in another 'distant' periphery of the community. On the other hand, community members who frequently work on the same modules should be supposed to reciprocally communicate. Then we think that coordination network structures could be decomposed in modules (mailing lists) according to affiliation patterns of agents (developers).

In order to explore this issue, as mentioned at the beginning of this analytical section, we built a two-mode network where nodes of type one are programmers and nodes of type two are mailing lists (see respectively red nodes and yellow nodes in figure 6. The weight of this affiliation is computed as the number of email that a programmer sent to a mailing list per year.

From the two-mode network we 'derived', a new one-mode network (folded) whose nodes are only mailing lists. The underlying assumption when we build this new network is that the higher the number of programmers who use the same mailing lists the more those mailing list refer to interdependent activities. By construction, two mailing lists were linked when at least a developer wrote an email on both. The weight of these relations have been imposed equal to the sum of developers shared by mailing lists dyads (and adjusted for the weight of affiliation). These new (folded) network loses the property of representing 'exact' communication patterns but it is less sensitive to cut-reduction. This means that we can consider only the developers who, wrote at lest a given number of emails (for example 10) over a year time period without dramatically altering the network structure.

[3] A modular structure, or architecture, is intended as one in which components (building blocks) are barely interdependent among them. A practical consequence for product-project management is that near independent components can be developed in parallel.

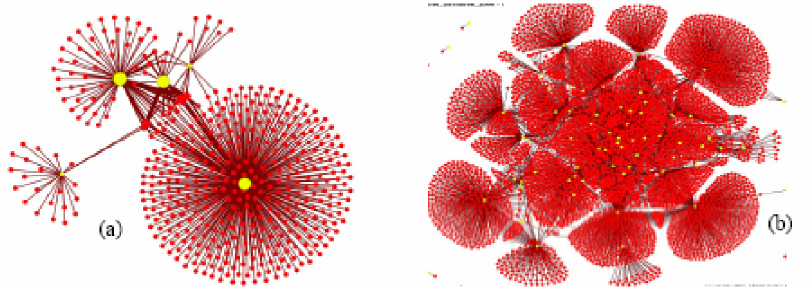

Fig. 6. Affiliation networks with data for 1997 **(a)** and 2000 **(b)**. Yellow nodes are mailing lists (intended as coordination modules), while red nodes are community members (intended as coordination agents). In figure (a) the size of nodes is proportional to its weighted degree.

In order to find sub-communities of coordination modules (represented for example as clusters of mailing lists), we used a folded one mode (mailing list to mailing list) networks (year 2000) reduced applying a cut-with threshold = 10. Then, we used the Newman clustering algorithm for detecting community structures. The modularity level was measured by a clustering coefficient Q ranging from 0 (non modular structure) to 1 (totally modular structure). We found a Q = 0.2013 in the mailinglist-mailinglist network (see figure 3.b where the same color is assigned to nodes that belong to the same cluster).

This research strategy was intended as a test for the 'resistance' of folded networks to cut-like reductions, we tried it for increasing cut-thresholds, ranging from 0 to 10, before running the Newman clustering algorithm and we found that the Q (modularity coefficient) only changed by a 0.1% for that range. The resulting modularity coefficient (Q) values could be interpreted as detecting a low modular organizational structure (of coordination). We also repeated the clustering process using a folded network with only developer-nodes and obtained (Q) values which were very close to 0.8, highlighting a very modular social structure, for both original and reduced (having cut threshold from 0 to 10) networks.

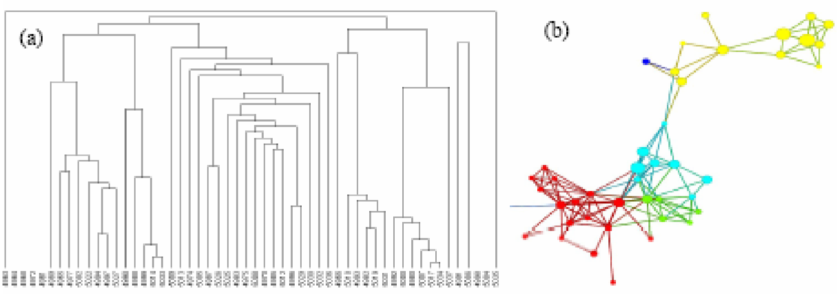

Fig. 7. (a) Dendogram; module-module (lists) network, cut threshold = 10, Q = 0.2013. **(b)** Reduced network. Node colors reflect Newman algorithm clusters.

3 Conclusions and Further Research

The results that we reported in this paper confirm that, when we build direct communication networks using the in-reply-to field of email headers for generating links, the overall network topology tends to develop scale-free qualities. This could be interpreted as the presence of self-organization in virtual communities, that is coordination structures could be thought to emerge in absence of central planners.

Despite this finding, we showed how the same networks could reveal that organizational design, which may be viewed as an almost opposite exogenous organizing principle, could have been affecting coordination-communication patterns over the time. We think that a further exploration of connectivity patterns could advantage the knowledge in the field of emergence of governance in virtual communities. In particular it could be interesting to control for the existence of eventual correlations among developers attributes (productivity, tenure etc.) and Thompson's typical interdependencies.

Driven by contrasting (or balancing) dynamics that we have documented, we further explored the issue of finding core interaction components in the overall networks. At a macro-level we observed a more clustered structure after reduction. However, direct-communication networks resulted very sensible to a low cut-reduction threshold, that is the shape of networks changed a lot when we just assumed that core community members exchanged at least three emails over a one year time period. This means that further micro-level analysis, like the one conducted in paragraph 2.2 could not be significant anymore.

A possible way to cope with this issue is presented in paragraph 2.4, where we proposed a different way to represent communication networks based on the idea of affiliation of developers to mailing list as 'modules' of the overall coordination structure. We have shown that 'folded' networks, either agent-agent or list-list from affiliation ones (with a cut threshold of ten) are respectively highly modular and low modular ones. A further contribution in this direction could be the construction of networks where developers affiliate to a more micro-level of coordination-communication that is emails threads. This could offer a representation which is closer to direct communication without suffering from obvious problems of sensibility to cut reduction.

References

1. Powell, W.W., Koput, K.W., Smith-Doerr, L.: Inter-organizational collaboration. Admin. Sci. Quart. 41(1), 116–145 (1996)
2. Ancona, D., Bresman, H., Kaeufer, K.: The comparative advantage of X-teams. MIT Sloan Manage. Rev. 43(3), 33–39
3. von Hippel, E.: Task Partitioning. Res. Policy 19, 407–418
4. Pich, M.T., Loch, C.H., de Meyer, A.: On Uncertainty, Ambiguity, and Complexity in Project Management. Manage. Sci. 48(8), 1008–1023 (2002)
5. Nickerson, J.A., Zenger, T.R.: A Knowledge-Based Theory of the Firm: The Problem-Solving Perspective. Organ. Sci. 15(6), 617–632 (2004)
6. Ethiraj, S.K., Levinthal, D.: Bounded Rationality and the Search for Organizational Architecture. Admin. Sci. Quart. 49(3), 404–437 (2004)

7. Kleinbaum, A.M., Stuart, T.E., Tushman, M.: Communication (and Coordination?) in a Modern, Complex Organization. Admin. Sci. Quart. (2008) (under review)
8. Monge, P., Heiss, B.M., Margolin, D.B.: Communication Network Evolution in Organizational Communities. Communication Theory 18(4), 449–477 (2008)
9. Breiger, R.L.: The Duality of Persons and Groups. Social Forces 53(2), 181–190 (1974)
10. Newman, M.E.J., Givran, M.: Finding and Evaluating Community Structure in Networks Phys. Rev. E 69 (2004)
11. Freeman, L.C.: A Set of Measures of Centrality Based on Betweenness. Sociometry 40(1), 35–41 (1977)
12. Weiss, M., Moroiu, G., Zhao, P.: Evolution of Open Source Communities. International Federation for Information Processing 203, 21–32 (2006)
13. Watts, D., Strogatts, S.: Collective Dynamics of ‘small-world’ networks. Nature 393, 440–442 (1998)
14. Barabasi, A.L., Albert, R., Jeong, H.: Scale-free Characteristics of Random Networks: the Topology of the World-Wide Web. Physica A 281(15), 69–77 (2000)
15. Lerner, J., Tirole, J.: Some Simple Economics of Open Source. J. Ind. Econ. 50(2), 197–234 (2002)
16. Dalle, J.M., David, P.A.: SimCode: Agent-based Simulation Modeling of Open-Source Software Development (2006) (working paper)
17. O’Mahony, S., Ferraro, F.: The Emergence of Governance in an Open Source Community. Acad. Manage J. 50(5), 1079–1107 (2007)
18. Thompson, J.D.: Organizations in Action. McGraw-Hill, New York (1967)
19. Mockus, A., Fielding, R., Herbsleb, J.: Two Case Studies of Open Source Software Development. ACM Trans. Softw. Eng. And Methodology 11(3), 309–346
20. Koch, S., Schneider, G.: Effort, Co-operation and Co-ordination in an Open Source Software Project: GNOME. Inform. Syst. J. 12(1), 27–42 (2002)
21. von Hippel, E., von Krogh, G.: Open Source Software and the “Private-Collective” Innovation Model. Organ. Sci. 14(2), 209–223 (2003)
22. Crowston, K., Howison, J.: The Social Structure of Free and Open Source Software Development. First Monday 10(2) (2005)
23. MacCormack, A., Rusnak, J., Baldwin, C.: Exploring the Structure of Complex Software Designs. Manage. Sci. 52(7), 1015–1030 (2006)
24. Simon, H.A.: The architecture of complexity. Proceedings of the American Philosophical Society 106, 467–482 (1962)