1-31-2010

Understanding FLOSS Implementation as a Key for Organization Design

Paolo Spagnoletti
CeRSI-LUISS Guido Carli, pspagnoletti@luiss.it

芽|Sprouts Working Papers on Information Systems ISSN 1535-6078

Understanding FLOSS Implementation as a Key for Organization Design

Paolo Spagnoletti
CeRSI-LUISS Guido Carli, Italy

Abstract
Growing research on Free/Libre Open Source Software (FLOSS) has addressed a variety of questions focusing on aspects ranging from open source development processes and developer motivation to economic and policy implications. Nevertheless, a few authors have examined the use of FLOSS and its implementation in organizations. Implementation studies represent a particularly promising area for information systems researchers who are still in the process of theorizing about the relationship between the distinctive properties of FLOSS and the processes of implementation and use. The goal of this paper is to present a taxonomy of FLOSS initiatives drawn on the outcomes of a previous study and to link the results with the multi-level framework defined by Niederman et al (2006). We also suggest a possible methodology to further investigate the possible combinations of organizational practices in order to achieve the expected benefits from FLOSS implementation.

Keywords: open source, organization design, set-theoretic approach, chemistry of organization

Permanent URL: http://sprouts.aisnet.org/10-11

Reference: Spagnoletti P. (2010). "Understanding FLOSS Implementation as a Key for Organization Design," Proceedings > Proceedings of ALPIS . *Sprouts: Working Papers on Information Systems*, 10(11). http://sprouts.aisnet.org/10-11

UNDERSTANDING FLOSS IMPLEMENTATION AS A KEY FOR ORGANIZATION DESIGN

Paolo Spagnoletti, CeRSI - LUISS Guido Carli University, Roma – Italy

pspagnoletti@luiss.it

Abstract

Growing research on Free/Libre Open Source Software (FLOSS) has addressed a variety of questions focusing on aspects ranging from open source development processes and developer motivation to economic and policy implications. Nevertheless, a few authors have examined the use of FLOSS and its implementation in organizations. Implementation studies represent a particularly promising area for information systems researchers who are still in the process of theorizing about the relationship between the distinctive properties of FLOSS and the processes of implementation and use. The goal of this paper is to present a taxonomy of FLOSS initiatives drawn on the outcomes of a previous study and to link the results with the multi-level framework defined by Niederman et al (2006). We also suggest a possible methodology to further investigate the possible combinations of organizational practices in order to achieve the expected benefits from FLOSS implementation.

Keywords: open source, organization design, set-theoretic approach, chemistry of organization

1. INTRODUCTION

Growing research on Free/Libre Open Source Software (FLOSS) has addressed a variety of questions focusing on aspects ranging from open source development processes (Feller, 2001; Scacchi et al., 2006) and developer motivation (Hann et al., 2004) to economic and policy implications (Méndez-Durón, 2009). Nevertheless, a few authors have examined the use of FLOSS and its implementation in organizations. For example, by querying the Business Source Premier database with the following keywords: 'open source' AND 'implementation', 'open source' AND 'development', 'open source' AND 'business' and 'open source' AND

'motivation', the following results were provided: 81, 598, 363 and 328 respectively, over an amount of 2438 papers on 'open source'. This demonstrates a prevalence of contributions focusing on the level of open source communities and concentrating on their economic, process and behavioural aspects. In their case study on FLOSS implementation in a large Irish hospital, Fitzgerald and Kenny (2003) claim that there is very little research on the deployment of OSS systems within organizations.

Findings from different research streams suggest that FLOSS has several relationships with both the social and the technical variables of an organization. From a socio-technical standpoint, the technical subsystem comprises of devices, tools and techniques needed to transform inputs into outputs in a way which enhances the economic performance of the organization. The social subsystem comprises of the employees (at all levels) and the knowledge, skills, attitudes, values and needs they bring to the work environment as well as the reward system and authority structures that exist in the organization (Lin and Cornford, 2000). With respect to the social variables, FLOSS seems to have an impact, for instance, on the job enrichment of people working in the IT sector, by enabling organizational learning processes for building customized solutions (von Hipple and von Krogh, 2003; Lerner and Tirole, 2002). Job enrichment can also be affected by enhancing the cooperation among the internal teams and the communities external to the organizational context (Jin et al., 2005). Furthermore, the structure of an organization seems to be influenced by FLOSS adoption and its implementation through the development of relationships with IT companies, research institutions and communities of users and developers which are novel with respect to the dichotomy hierarchy and market (Watson et al., 2005). From the point of view of the technical variables, FLOSS intrinsic characteristics (Mockus et al., 2002) can affect the internal processes which are supported by more flexible and customized software solutions. Therefore FLOSS can play different roles in the context of socio-technical systems by

affecting IS performance of an organization. Such roles refer to different levels depending on the typology of products/projects and on the use that organizations assign to these technologies. At an individual level, FLOSS has an impact on the way elementary activities are performed. Moreover, the acceptance of these technologies represents a key issue to be investigated both among IT staff members and other employees. At a group level, FLOSS implementation brings the dynamics of OS communities into the organizations and it contributes to the job enrichment of IT and domain experts.

Such a complex and intertwined network of relationships makes difficult to combine organizational practices in a way that determines the success of FLOSS initiatives according with the expected benefits. Indeed, organizations face a number of issues entering into initial open source usage, integrating open source into their portfolio, deciding levels of community participation and assessing the economic, organizational, and technical impacts of open source on operations, tactical and strategic business practices (Niederman et al. 2006).

According with Grandori and Furnari (2008), the organization design problem has shifted from the comparative assessment of organizational configurations to the definition of design rules and processes capable of generating solutions, able to fit any given particular situation. In this view, both the fundamental organizational elements to be combined, and the combinatory rules for generating specified outcomes are the fundamental elements of a "chemistry of organization" which underpins organization design.

Referring to FLOSS implementation, the fundamental organizational elements to be combined depend on the type of FLOSS project as well as the outcomes (expected FLOSS benefits) and the combinatory rules for generating them. Therefore, the first step in the direction of applying the principles of this new approach to organization design is the identification of categories to group FLOSS initiatives. Then, the objectives within each category can be defined. Finally, starting

from a set of given organizational practices it will be possible to apply set-theoretic methods to assess and formulize the complex ways in which causes combine to create outcomes (Fiss, 2007). In this working paper we deal with the first issue by presenting a taxonomy of FLOSS initiatives drawn on the outcomes of a multiple case study conducted on 16 Italian public administrations about FLOSS implementation initiatives (Spagnoletti and Federici, 2010).

2. TYPE OF ARTIFACTS

In order to make order in the above mentioned complex network of intertwined variables, a multi-level framework has been developed by Niederman et al. (2006) as a lens through which MIS scholars can examine the open source phenomenon. This framework, drawn on a literature review, is based on five levels of analysis: the artifact, the individual, the group/project/community, the organization and the broader societal perspective. For each of the five levels, authors have defined variables claiming that most interesting research questions will involve relationships that cross such levels.

According with authors of this framework, artifact level attributes can play both the role of dependent and independent variables in their relationship with variables from other levels (Niederman et al., 2006). In the following table we provide some examples of measures associated to these variables in order to give an idea of the extensive scope of this domain of research.

Table 1: Open Source variables by Level of Analysis (adapted from Niederman et al (2006)

MIS levels of analysis	Variables	Sample Measures
Artifact (A)	- Artifact type	- Infrastructure, package, application, cross functional

Table 1: Open Source variables by Level of Analysis (adapted from Niederman et al (2006)		
	- Licence type - Quality of product	application - GNU GPL, BSD, etc. - Security, reliability
Individual (B)	- Developer - User	- Salaried/unsalaried employees, volunteers - Non developer users/ non using developers - Motivation, job satisfaction, productivity, etc.
Group, project, community (C)	- Organization governance - Mechanics for artifact creation and exploitation	- Participation, satisfaction, conflits - Control/trust - Development methods and tools, communication style, exploitation methods
Organization (D)	- Developer - Distributor - Users	- Economic benefit/loss - Selection methods - Dependence on outside vendor or supplier - Training and staff skills - Open source/ market and hierarchy
Society (E)	- Influence on society	- Cross-national outsourcing - Privacy and security - Local economic development

The first result of a multiple case study conducted on 16 Italian public administrations that recently carried out innovative projects based on FLOSS, has been the definition of a taxonomy which allows us to group cases along five categories by specifying the project type. This taxonomy allows to operationalize the "type" variable the "artifact" level of analysis (Niederman et al. 2006) by defining five set of projects.

Firstly, there are cases in which open source software has been used at a system level by implementing either a centralized or a distributed server infrastructure (A category). Secondly, client application packages (i.e. office automation tools, specialized tools) have been introduced in public administrations with differences in terms of the number of users involved and the

migration strategy applied (B category). Thirdly, some cases provide insight on the deployment and configuration of customized server side applications (i.e. content management systems) based on open source components (C category). Fourthly, a class of cases shows the possible role of FLOSS when a business process management approach is adopted to increase the efficiency and effectiveness of internal processes through the development of new software modules based on open standards and frameworks (D category). Finally, a set of cases provides examples of cooperation models through which FLOSS can be spread among public administrations (E category). Since some of the cases contain aspects related to more than one category, we decided to cite them more than once by analyzing the relevant aspects each time.

3. CONCLUSIONS

Implementation studies of FLOSS represent a particularly promising area for information systems researchers who are still in the process of explicitly theorising about the relationship between the distinctive properties of FLOSS and the processes of its implementation and use. However, the diversity of the emerging constructs, in terms of level of analysis and variables typology, delineates a complex scenario for testing and developing theories.

In this paper we linked an empirically based taxonomy of FLOSS implementation projects with the related attribute at the artifact level ("artifact type") adopted by Niederman stressing the need for explorative implementation studies. Moreover we suggest a possible methodology based on set-theoretic methods and on the new science of "organizational chemistry" to further investigate how to combine organizational practices in order to achieve the expected benefits from FLOSS implementation.

References

Feller, J. (2001) "Thoughts on Studying Open Source Software Communities," in: Realigning Research and Practice in Information Systems Development: The Social and Organizational Perspective, N.L. Russo, B. Fitzgerald and J.I. DeGross (eds.), Kluwer, 2001, pp. 379–388.

Fiss, P. (2007) A Set-Theoretic Approach to Organizational Configurations. Academy of Management Review (32/4), 1180-1198

Fitzgerald B. (2006) The transformation of open source software. MIS Quarterly Vol. 30 No. 3, pp. 587-598/September 2006.

Fitzgerald, B., and Kenny, T. (2003) "Open source software in the trenches: Lessons from a large-scale OSS implementation", in Proceedings of 24"' International Conference on Information Systems, S. T. March, A. Massey, and J. I. DeGross (eds.), Seattle, December 2003, pp. 316-326.

Grandori A., Furnari S., (2008), A Chemistry of Organization: Combinatory Analysis and Design. Organization Studies 29(03): 459-485

Hann, I.-H., Roberts, J., and Slaughter, S.A. (2004) "Why developers participate in open source software projects: An empirical investigation," in: Twenty-Fifth International Conference on Information Systems, Washington, DC, 2004, pp. 821–830.

Jin, L., Robey, D., and Bourdreau, M.C. (2005) “Beyond Development: A Research Agenda for Investigating Open Source Software User Communities”, Proceeding of the International Resources Management Association Conference, San Diego, May 15-18, pp. 642-645.

Lerner, J. And J. Tirole (2002) “Some simple economics of open source”, *The Journal of Industrial Economics* (50)2, pp. 197.

Lin, A. and T. Cornford (2000) "Sociotechnical Perspectives on Emergence Phenomena" in The New Sociotech: Graffiti on the Long Wall, Cookes, E.D., Willis and R. Lloyd-Jones (eds) Godalming, Springer.

Méndez-Durón, R., and García, C.E. (2009) "Returns from Social Capital in Open Source Software Networks," Journal of Evolutionary Economics (19), pp 277–295.

Mockus, A., R. T. Fielding, et al. (2002) “Two Case Studies of Open Source Software Development: Apache and Mozilla”, *ACM Transaction on Software Engineering and Methodology* (11)3, pp. 309-346.

Niederman, F. , Davis A., Greiner M.E., Wynn D., York P.T., (2006) “A research agenda for studying open source I: a multi level framework, Communication of AIS, Volume 18, Article 7

Scacchi, W., Feller, J., Fitzgerald, B., Hissam, S., and Lakhani, K. "Understanding free/open source software development processes," Software Process— Improvement and Practice (11:2) 2006, pp 95–105.

Spagnoletti, P., Federici T. (2010) “Exploring FLOSS implementation through a multiple case study” (to appear)

von Hippel, E. and G. von Krogh (2003) “Open Source Software and the “Private-Collective” Innovation Model: Issues for Organization Science”, *Organization Science* (14)2, pp.209.

Watson, R. T., et al. (2005). Governance and Global Communities”. Journal of International Management 11(2): pp. 125-142.

芽|Sprouts

Working Papers on Information Systems | ISSN 1535-6078

Office:
Sprouts
University of Amsterdam
Roetersstraat 11, Room E 2.74
1018 WB Amsterdam, Netherlands
Email: admin@sprouts.aisnet.org

芽|Sprouts