

Applying Open Source Development Practices Inside a Company

Juho Lindman¹, Matti Rossi¹, and Pentti Marttiin²

¹ Helsinki School of Economics, Information Systems Science, PO Box 1210, 00101 Helsinki, Finland. {}@hse.fi, <http://www.hse.fi>

² Nokia Siemens Networks, PO Box 31, 02022 Nokia Siemens Networks, Finland. {}@nsn.com, <http://www.nsn.com>

Abstract. Open Source Software development is seen as a panacea by many companies. The promise of community-style development, innovation and cost savings drive the wider adoption of OSS in companies. However, it is still difficult to institutionalize the open and agile culture of sharing innovation especially into larger departmentalized organizations. The aim of this research paper is to investigate the characteristics of one successful OSS development implementation approach limited inside a company (Inner source). Based on our data, we argue that there are possibilities for employing OSS as a new kind of development process within a company and leveraging thus the innovation potential inside the company.

1 Introduction

It is currently virtually impossible to conduct any business without encountering software created using OSS development methods. Companies are becoming more interested in understanding how to use OSS development inside their organizations or in inter-company development activities [2] [5] [6]. Research indicates that it is possible to build solid business cases around OSS [4].

However, researchers and practitioners are still struggling with the challenge of deploying OSS development practises inside companies. This study aims to provide some initial answers by describing the adoption of inner source portal in a case company. Our viewpoint is that of a support service organization launching new tools for business unit use. The specific case is Nokia, a leading telecommunications company using adopted OSS development process called iSource. We aim to provide an account of the intra-company portal usage and what kind of projects seem to benefit. Findings indicate that 1) there is a growing interest towards iSource, 2) projects that use the portal are heterogeneous, and 3) projects seem to benefit from openness enabled by OSS tooling.

Please use the following format when citing this chapter:

Lindman, J., Rossi, M. and Marttiin, P., 2008, in IFIP International Federation for Information Processing, Volume 275; *Open Source Development, Communities and Quality*; Barbara Russo, Ernesto Damiani, Scott Hissam, Björn Lundell, Giancarlo Succi; (Boston: Springer), pp. 381–387.

2 Research methodology

Our goal is to provide an account of a successful implementation of OSS practices inside a company. This is necessary because it is widely believed that OSS development will gain more footing in different kinds of organizations [5]. Taking account our goal, we select to conduct an interpretative case study on Nokia iSource service organization [16]. Interpretative approach helped us to take into account the interplay of the service organization and its environment [14]. We also used other methods and sources of evidence to develop our understanding of the events and to ensure the validity of our findings [3].

We started out with managerial interviews and direct key user contacts. Our informants included people from the service organization and business units to build solid case description. The respondents were selected to represent the different levels of inner customers and service organizations. We conducted four interviews and reviewed usage data provided by the technology platform to corroborate the interview findings. Usage data consists of user information, number of projects created and their activity. One of the researchers was working for the company and thus has a very good view of the situation and context of the case from the inside. This might cause bias, but that is offset by the depth gained by being an insider.

As the results from the different data sources seem to be aligned, we assume that the findings support each other. We aimed to make analytical, not statistical generalizations. Statistics offered by the platform did not yet enable developing understanding of models and causal relationships between different data. Thus these two were used in conjunction with interviews and participant observation to validate the case findings.

3 Review of literature

Theories describing OSS phenomenon sometimes suffer from a promoted assumption that OSS communities and processes would be a priori similar [10] [12] [13]. In contrast, it has been shown that OSS communities, software companies and the relations between them are not homogeneous [1]. Thus it seems plausible to presume that also inner source communities are heterogeneous.

The companies that wish to launch communities are in different position than those that wish to benefit from existing OSS software [4]. Also the challenges related to the processes are different depending on whether company has outbound or inbound OSS approach [4]. Commercial context enables different roles for companies using OSS [7] and companies have a variety of choice regarding integrating their operations with the selected business model [9] [11].

Hierarchical organizations benefit from open development communities that promote flow of information and ideas between professionals [15]. In traditional proprietary software development software is developed and maintained centrally in a selected business unit or program. Development community is a way to respond to

the misalignment of experience, skills and interests [8]. This is especially beneficial to large organizations prone to re-organization. Company's priorities might change due to market conditions, often leading to the reallocation of developers into different projects.

Inner Source aims to leverage the power of open source practices, methods and tools in company context [7] [5]. This means in practice: 1) software produced is open to everyone working for a company to use and develop. More accurately, all activities relating to the software development are open and visible to everyone, and 2) interest to develop software initiates a community to take care of the development.

4 Nokia iSource

We selected for analysis a successful OSS inspired implementation, Nokia iSource. Nokia is the world leader in mobile communications. It is a publicly held company with listings in five major exchanges. Today the iSource portal has two instances one for Nokia and another for Nokia Siemens Networks (NSN). NSN is a company created in a merger between Nokia Networks and a part of Siemens in 2007. Our data focuses mainly on events that happened before the merger.

iSource is a company wide portal created in 2001 for managing and working with software assets. Its introduction was originally encouraged by HP's good experiences [2]. Technically, at the time of the study, iSource was a fork of OSS version of Sourceforge portal dated back to 2001 and maintained separately from the mainline. The idea behind iSource was to provide a portal enabling visibility of software and the source code. The goals were to increase individual engineers' awareness of software developed inside the company, and to boost innovation by avoiding the problem of re-implementing the wheel. Currently, projects are counted in hundreds and users in thousands, but activity status varies depending on the project life-cycle and funding. iSource has a support service, that promote iSource service and champions its adoption among business.

5 Observations from cases

5.1 Increase in adoption

According to the user and project statistics iSource has enjoyed stable increase in use. This trend can be observed from Figure 1 below, which shows the number of new projects created per year (the exact numbers were omitted to protect company confidential information, but the trend is visible).

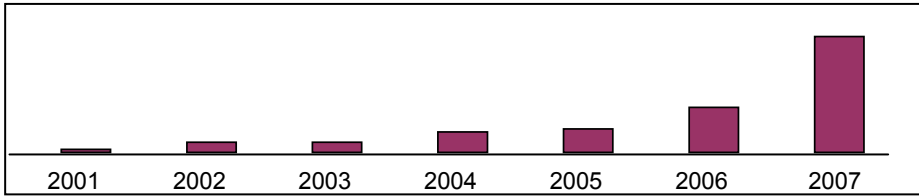


Fig. 1. New iSource projects with their creation year

5.2 Project heterogeneity and openness

iSource has been used for different kinds of projects. This variety can be observed for example when comparing the number of developers and the administrators or the ratio of the two. Figure 2 shows the numbers of developer per project and Figure 3 the number of administrators.

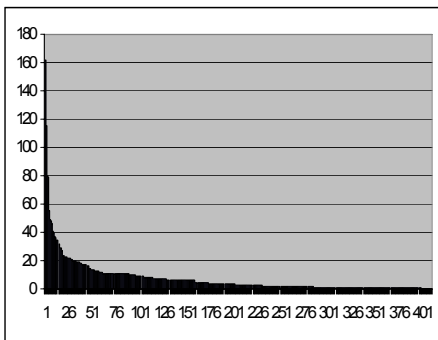


Fig. 2. The number of developers/project

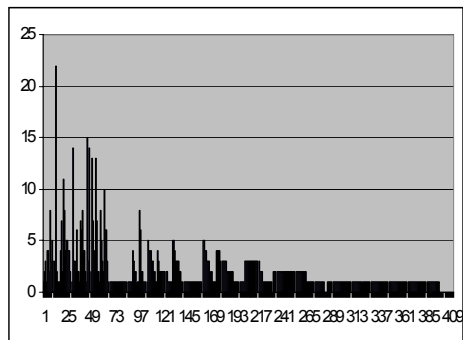


Fig. 3. The number of admins/project number

Most projects using iSource are having one or more of the following characteristics:

1. Projects have selected Subversion or CVS (Concurrent Version System) as their SCM tool (iSource service is the official one for hosting these SCM repositories).
2. Projects are distributed to multiple sites and continents (open source tools require less bandwidth)
3. Projects are using agile development methods (OSS tooling attracts agile projects) or take advance of continuous integration tools (e.g. CruiseControl or Bamboo)
4. Software is developed in inter-company collaboration (iSource is opened to collaborators)

5. Projects are advancing Inner Source (openness when sharing e.g. scripts and libraries; OSS development model in platform development)

The following table shows three top-10 lists of iSource projects. It was what kinds of projects have most developers, are benefitting of file downloading, and are using CVS. The project identification (or categorization) and is done by selecting a key word from the project's public summary.

Table 1. Breakdown of top-10 lists: # of developers, file downloads and CVS usage (commits).

Categorization (Most developers)	#	Categorization (File Downloads)	#	Categorization (CVS commits)	#
Platform	6	Framework	3	Platform	4
Simulator	2	Scripts and libraries	2	Framework	2
Reference Implementation	1	SDK	1	Emulator	1
Testing environment	1	Software suite	1	Simulator	1
		Platform	1	Language	1
		Emulator	1	Protocol architecture	1
		Language	1		

First, 10 projects with the biggest developer number were overviewed. It seemed that among the biggest projects there were several platform projects as indicated in Table 1. Not all the projects are ranked in downloads list. This probably means that no files are stored. The projects on this list are different from those that have the most developers. Smaller projects are able to generate interest and share for example emulators, scripts and libraries. Downloads also correlate with project age. Projects may use the portal with or without SCM. Based on our statistics less than 50% of projects have used or at least tried CVS (Subversion was just introduced by the service). Some projects in our table 1 have also been transferred to Sourceforge at some point of their life-cycle. Two open sourced projects (Sofia SIP <http://sourceforge.net/projects/sofia-sip/> and Python for S60 <http://sourceforge.net/projects/pys60/>) are the top of the iSource lists.

6 Findings and discussion

The aim of our research was to describe usage and show what kind of projects benefit from intra-company portal. There is a clear growing trend in iSource usage. As projects adopted it voluntarily, there is clearly an expectation of benefit. The portal launch needs to be planned, projects need support and there needs to be a robust technical support and evangelists to keep the platform alive.

Projects inside iSource are heterogeneous: most of the biggest projects are platform projects, but on the other hand frameworks and libraries are more downloaded. So, we cannot claim homogeneity between projects. One way to characterize the different project types would be to divide them into: 1) SCM projects, 2) distributed projects, 3) agile projects, 4) intra-company collaboration projects, and 5) Inner Source projects utilizing the full potential of OSS approach.

Openness seems to support adoption of the project results. Not only have projects been able to gain visibility inside the company, but some iSource projects have been moved into open domain once tested inside the global company.

When describing one case study, there is always a question of how to transfer the findings into other companies. This is especially true when dealing with global company tool adoption. However, from our industrial experience and discussions with several other companies in scope of a European research project, it seems that similar challenges need to be solved when moving towards developer communities. The experience of the companies already implementing inner source are very valuable to other industry actors considering moving towards similar development practices.

Acknowledgements

The authors would like to thank the ITEA-COSI project.

References

1. L. Dahlander, M. Magnusson. Relationships Between Open Source Companies and Communities: Observations from Nordic Firms. *Research Policy*, 34(4): 481-493, 2005.
2. J. Dinkelacker, P. Garg, R. Miller, D. Nelson. Progressive Open Source. *Proceedings of the 24th International Conference on Software Engineering, Buenos Aires, Argentina, May 2002*.
3. K. M. Eisenhardt. Building Theories from Case Study Research. *Academy of Management Review* 14 (4): 532-550, 1989.
4. M. Fink. *The Business and Economics of Linux and Open Source*. Prentice Hall, New Jersey, 2002.
5. B. Fitzgerald. The Transformation of Open Source Software. *MIS Quarterly*, 30(2):587-598, 2006.
6. V. Gurbani, A. Garvert, and J. Herbsleb. A Case Study of Open Source Tools and Practices in a Commercial Setting. In *Open Source Application Spaces: Proceeding of the Fifth Workshop on Open Source Software Engineering, St Louis, USA, 17-21 May 2005*.
7. Ø. Hauge, C. Sørensen, and A. Røsdal. Surveying Industrial Roles in Open Source Software Development. *OSS 2007. June 11-14 2007*, Limerick, Ireland.
8. C. Melian, CB. Ammirati, P. Garg, and G. Sevon. Building Networks of Software Communities in a Large Corporation. Hewlet Packard POS, 2001.

9. A. Osterwalder, Y. Pigneur. Clarifying Business Models: Origins, Present and Future of the Concept. *Communications of the AIS*, v. 15, May 2005.
10. B. Perens. The open source definition. In C. DiBona, S. Ockman, and M. Stone, eds, *Open Sources: Voices from the Open Source Revolution*. O'Reilly & Associates, Sebastopol, CA, 1999.
11. R. Rajala, M. Westerlund, and J. Nissilä, *Software for Free? Revenue Models In The Open Source Software Business*. Proceedings of the 5th Global Conference on Business&Economics, Cambridge University, Cambridge, UK, 2006.
12. E.S. Raymond. *The Cathedral and The Bazaar – Musings on Linux and Open Source by Accidental Revolutionary*, O'Reilly&Associates, Sebastopol, CA., 1999.
13. A.M. Szczepanska, M. Bergquist and , J. Ljunberg. High Noon at OS Corral: Duels and Shoot-Outs in Open Source Discours. In: J. Feller, B. Fitzgerald., S. A. Hissam, and K. R. Lakhani, eds. *Perspectives on Free and Open Source Software*, MIT Press, Cambridge, Mass., 2005.
14. G. Walsham. *Interpreting Information Systems in Organizations*. Cichester, Wiley&Sons. 1993.
15. G. von Krogh, E. von Hippel. The Promise of Research on Open Source Software. *Management Science*, 52(7): 975-983, July 2006.
16. R. K. Yin, *Case Study Research, Design and Methods*, 2nd ed., Sage Publications, Newbury Park, 1994.