On the Transition to an Open Source Solution for Desktop Office Automation

Bruno Rossi, Barbara Russo, Paolo Zuliani, Giancarlo Succi

Center for Applied Software Engineering, Free University of Bolzano-Bozen
Piazza Domenicani 3, 39100 Bolzano-Bozen, Italy
{bruno.rossi, barbara.russo, paolo.zuliani, giancarlo.succi}@unibz.it

Abstract. There are many claims about the benefits of Open Source Software (OSS). However, these claims are seldom supported by empirical evidence, while on the other hand there are several impediment factors which might overcome the advantages deriving from the use of OSS in a corporate environment: cost of transition, personnel training and, interoperability and integration with existing technologies. These factors are often used by OSS opponents. In this paper we first report of a small-scale deployment of OSS for office automation in Public Administration bodies. We describe the environment, the process and the problems encountered. Next, we describe a proposal for a future experiment for empirically assessing OSS impediment factors, focusing in particular on personnel productivity. We again propose the deployment of OSS office automation tools in Public Administration bodies and we describe a system for monitoring the effect of the use of OSS on personnel productivity.

1 Introduction

Open Source Software (OSS) has grown a lot in popularity. Linux and the Apache web server are found in respectively 30% and 66% of the Internet's public servers, according to Netcraft's survey [1]. We thus have some empirical evidence that OSS can work well, at least for the server side of a client-server architecture.
By assuming the fact above, supporters claim that OSS leads to a reduction in IT expenditure because:

- OSS is free, one does not have to pay any license;
- Source code is available, so it is possible to tune the software for specific needs by removing unnecessary, resource-consuming features. This translates in the possibility of using less powerful, thus less expensive, hardware.

The first reason is indeed true, while for the second reason we do not have any empirical study comparing OSS and proprietary solutions: proponents usually report common knowledge experience. Furthermore, such knowledge and experience are confined to very specific applications such as server architectures or software development. For example, in 2001 Amazon.com adopted Linux for most of its servers and reduced by 24% ($17 million) the IT expenditure, as reported by the IDG Group ([2], [3]). In August 2002, Verizon Communications, one the biggest telecommunication operator in the USA, replaced the Unix and Windows workstations of its internal developers with systems based on Linux and OpenOffice. The average desktop cost dropped from $20,000 to $3,000 per developer and in the end the company saved $6 million 4.
It seems to be that the common feature of these success cases is that OSS has been able to penetrate the market only for applications which require more reliability and efficiency than user-friendliness and usability.
Therefore, it might be that OSS is not well suited for desktop and client applications, for which we know that Microsoft Office is the de-facto standard. If we think of a hypothetical deployment of OSS for desktop applications in a corporate environment, such as Public Administrations (PA), there are factors which might overcome the claimed advantages of OSS:

- Cost of transition from previous solutions;
- Interoperability and integration with existing solutions;
- Cost of training personnel for the new tools and hostility to change;
- Reduced productivity of the personnel.

The recent FLOSS project [5] funded by the European Union aimed at collecting data about the usage and development of OSS in Europe. Surveys were conducted between February and May 2002 on about 1,500 companies and public institutions, asking whether they were employing, or willing to employ, Open Source software. Four hundred of these were indeed using, or planning to do so in the near future, some kind of OSS.
There are two points of the FLOSS study [5] which are of interest for us:

- OSS for desktop applications (*e.g.* client operating systems, office automation, *etc*.) was employed only by the 20% of those four hundred establishments using OSS. If we further restrict to the use of OpenOffice that percentage drops to 10%. This confirms common wisdom that OSS is better suited for server and IT infrastructure tasks;
- It turned out that companies and public institutions were generally unable to quantify the benefits deriving from the use of OSS. They were also not even able to quantify benefits like license fees savings and hardware cost savings.

It is therefore important to empirically analyze and assess the benefits and the problems deriving from the use of OSS, in order to provide companies and public institutions with more significant data for their strategic decisions.
In particular, we focus on the OpenOffice suite: a set of key desktop applications which includes a word processor, a spreadsheet, a presentation manager, a drawing program, and an equation editor [6].
In this paper we first report of a small-scale deployment of the OpenOffice suite in several PA bodies. We describe the environment, the process and the problems encountered during the transition.

Next, building on that experience, we propose a future experiment for empirically evaluating the benefits and problems caused by the introduction of OpenOffice. The project aims at showing that OpenOffice allows personnel to produce as efficiently as Microsoft Office.

2 Small-scale deployments

The Consortium of the Townships of the Province of Bolzano-Bozen (Italy), in collaboration with the Centre for Applied Software Engineering of the Free University of Bolzano-Bozen, has performed a trial installation of OpenOffice in ten associate townships.

2.1 Environment

The trial installation of OpenOffice involved ten townships of the Alto-Adige region in Italy. Townships ranged from very small (five employees) to small-medium size (twenty employees). The activities performed are the usual office tasks: word processing, spreadsheet, *etc*. Microsoft Office was the only office automation tool used.
In the end, OpenOffice was installed on about one hundred desktop computers. The operating system was Microsoft Windows in all the cases.
We select a set of 16 PC computers uniformly distributed in the Townships. The end-users volunteered for the experiment belong to four different departments. In eight PC computers we have installed OpenOffice.

2.2 Process transition

Transitions lasted from two to four working days and employed two instructors each. Personnel training was performed on-site and one-to-one.
Instructors first went to the site for “exploring” the environment and for collecting the most used documents by offices’ personnel. The instructors then returned the day after with all the documents converted to OpenOffice’s format. They then installed OpenOffice and train the personnel by working on the very same documents they were usually working on.

The conversion of more than two hundred documents from Microsoft Word to OpenOffice was performed without any particular problem and with great efficiency: the size of an OpenOffice document was generally one third of the equivalent Word document.

2.3 Problems

Personnel do not generally look positively at the introduction of new or different technologies and at the abandon of those which is used to: a phenomenon called "hostility to change". The most reported reason is the refuse to use tools different from those of colleagues or from those used at home. However, during the transition to OpenOffice we found only a few employees showing hostility to change.
We instead have found an inefficient use of resources: the personnel routinely used only the very basic features of Office, and did not consider little more complicate features which would have lead to better use of resources.
Users with good knowledge of Office have not had any problem in switching to OpenOffice. Most of the problems have been caused by personnel with little Office knowledge.
Personnel training has been usually performed on-site and one-to-one, but it has turned out that instructors have had to frequently interrupt training because of incoming phone calls, urgent documents delivering, *etc*.

3 The experiment

The aim of the experiment is to studying, analyzing and evaluating the introduction of OpenOffice for all office automation tasks in the PA, while preserving existing proprietary solutions for desktop operating systems (*e.g.* Microsoft Windows). In particular, we investigate whether the use of OpenOffice does not significantly affect personnel productivity.
The experiment also aims at becoming a success case for the introduction of OpenOffice, and of OS desktop software in general, in companies and Public Administrations.
The experiment has been jointly conducted by Consortium of the Townships of the Province of Bolzano-Bozen, the Centre for Applied Software Engineering of the Free University of Bolzano-Bozen and a few local IT firms.

3.1 Design

The sample consists of 16 end-users. Eight of them volunteered to use OpenOffice. The rest continues to use Microsoft Office.

First we have drawn a picture of all the applications calling and called by the office automation tool used, and the macros used by each department.
Then we have monitored the use of the office automation tools for a period of seven weeks before the transition.
Soon after the transition we have monitored the use of both the old and new solution for a period of fourteen weeks.
Then we have configured the access to the documents to automatically opening them in OpenOffice. Although opening the documents with Microsoft Office is still possible, it requires a bit more complex procedure. Again we have monitored the use of Microsoft and OpenOffice for a period of three weeks.

3.2 Collected data

We have selected data from two weeks before the transition, two weeks soon after the transition, and two weeks during the new configuration access to documents.
The data have been collected automatically in background with the PROM tool.

PROM is a software engineering tool originally developed for collecting process and product metrics in software development ([8], [7]). For example, it is possible to record the time lapsed in working on a Java source file, or the number of modifications applied.
By the use of appropriate plug-in's we interfaced the PROM tool with Microsoft Office and OpenOffice, so that it is possible to collect process metrics for any kind of OpenOffice or Office document.

For this preliminary study we have collected data on interaction of applications, on time of use of applications, and on number of documents daily used.
We have not taken into account daily average time shorter than 5 minutes.

4 Data Analysis

In first seven weeks we have analyzed the calls between Word/Excel and the other Microsoft Office applications. We selected the following applications

Table 1. Description of the applications considered

Called by	Description
CPCQM.EXE	Printer driver Canon
MSOHELP.EXE	MS Help Menu
EXPLORER.EXE	Folder viewer
DW.EXE	MS Error Reporting tool
EXCEL.EXE	MS Excel
IEXPLORER.EXE	MS web browser
MSACCESS.EXE	MS Access
MSTORE.EXE	Microsoft Clip Organizer
IFRUN60.EXE	Oracle Forms (Runforms)
OUTLOOK.EXE	MS Mail client
WINHELP32.EXE	MS Help guide

Table 2: Top score applications calling Word and Excel

Calling	Word	Excel
EXPLORER.EXE	80.74%	94.38%
OUTLOOK.EXE	14.61%	3.42%
DW.EXE	0.64%	0.92%
IFRUN60.EXE	0.11%	0.00%
IEXPLORER.EXE	0.19%	0.00%
EXCEL.EXE	0.04%	0.00%
UNKNOWN	3.68%	1.28%

Table 3: Top scores applications called by Word and Excel

Called by	Word/Excel
CPCQM.EXE	71.05%
MSOHELP.EXE	13.16%
EXPLORER.EXE	7.89%
DW.EXE	5.26%
EXCEL.EXE	2.63%
IEXPLORER.EXE	2.63%
MSACCESS.EXE	0.00%
MSTORE.EXE	0.00%
IFRUN60.EXE	0.00%
OUTLOOK.EXE	0.00%
WINHELP32.EXE	0.00%

The patterns expressed in the tables indicate the interoperability of the desktop applications that needs to be taken into account in the transition: customization and adaptation of the office tools impact on effort and costs.
In fact despite the difference in percentage of calls all the applications that are in the table needs to be considered in the transition. For example we need to customize the call to an oracle DB as there has been at lest one call to this DB (IFRUN60.EXE).
For the same reason we analyze the existence of macros: for the accessible excel files it has been reported 43 macros for a total of 21,482 lines of code for 526 files inspected. No macros have been found in place for Word files.
We monitor the number of documents used and the daily time spent on the documents.
We derive a formula on daily productivity for each user

$$P_i = \frac{\#\,\text{Documents}}{\text{time}} \qquad (\mathbf{1})$$

We have different types of productivity. The documents are Office documents (Docs(O)) or OpenOffice documents (Docs (OO)). The time is the time of use of an application. So we have time for using the OpenOffice applications ad time for the use of Office applications – Open Office, time (OO) or Office, time (O).
Considering that OpenOffice files cannot be opened by Office there are three types of productivity.

$$P_{tot} = \frac{\#\ \text{Docs}\quad (O) \;+\; \#\ \text{Docs}\quad (OO)}{\text{time}\quad (O) \;+\; \text{time}(OO)} \qquad (\mathbf{2})$$

$$P_1 = \frac{\#\ \text{Docs}\quad (O)}{\text{time}\quad (O) \;+\; \text{time}(OO)} \qquad (\mathbf{3})$$

$$P_2 = \frac{\#\ \text{Docs}\quad (OO)}{\text{time}(OO)} \qquad (\mathbf{4})$$

Then we average the daily productivity in each period of analysis. The following graphs report the three productivities in the three periods in each of the two groups.

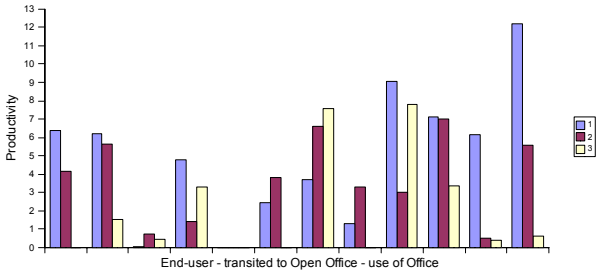

Fig. 1 Partial productivity (eq. 3) in the three periods in the group transited to OpenOffice

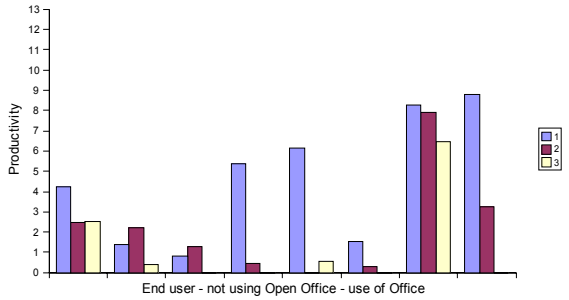

Fig. 2 Partial productivity (eq. 3) in the three periods in the group not using OpenOffice

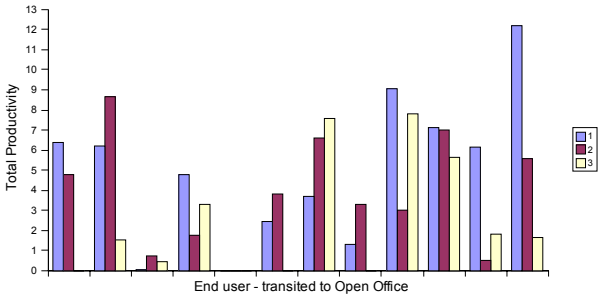

Fig. 3 Total productivity (eq. 2) in the three periods in the group transited to OpenOffice

The first picture represents the daily average productivity when documents are saved/modified as office documents (.doc, .xlm extension) disregarding the application used - within OpenOffice or Office (eq. 3). Therefore the productivity here is an upper bound of the productivity related to the solo use of Office: a user may have modified a file .doc with OpenOffice. Further analysis would consider this refinement (eq. 4).

The second picture displays the trends in the control group. From the picture we deduce that the first period of monitoring is characterized by a bigger productivity (eq. 3).

The third picture reports the productivity compute as ratio of number of files – indifferently Office or OpenOffice – and time spent with OpenOffice or Office applications (eq. 2).

5 Results

The preliminary analysis based on the groups comparison in the three periods of the experiment (Fig. 2), indicates that there is no lack of productivity in the group transited to OpenOffice.
Even more some of the members of the group transited to OpenOffice present a higher productivity when working both with only Office documents (eq. 3, Fig. 1 and Fig. 3 period n. 1) and with any kind of document (eq. 2, Fig. 3, period n. 3). In the transition, when the choice to use one or the other application is even (Fig. 1 and Fig. 3, period n.2), no increase of productivity has been registered: no documents new or saved as OpenOffice files have been produced.
In the third period the use of OpenOffice has increased as the path to access to Office applications has become more complex.

To facilitate the transition we have performed an analysis on application interoperability and existence of macros. This has helped to customize the new solution in terms of the needs of the end-user.

6 Analysis of the Problems

Again, the fact that Microsoft Office is by far the most used office automation tool raises the problem of training the personnel for OpenOffice. To this end, we have organized part-time courses on OpenOffice. The courses are held off-site, to avoid the disturbing factors experienced in the trial installation. In these courses, offices' personnel have been taught the basic and most used OpenOffice features, with the possibility of suggesting some particular topic of interest.
Another problem which might occur is the hostility to change. In this case, in order to maintain the efficacy of the training action, we might think of motivating the personnel by a series of "bonuses for change". Another solution is to train homogenous groups of people, that is, personnel coming from the same of closely related offices.
The choice of introducing OpenOffice while maintaining the same client operating systems is motivated by the need to minimize the training load for the personnel. That choice allows also a smooth transition, minimum interruption of public services and limits any possible hostility to change.
We will also establish: a hotline, a data base of success cases, a FAQ and a knowledge base. These services are aimed at PA personnel already trained and will offer user and technical support on various OS software of interest for the PA.

7 Limits of the experiment

This is a preliminary experiment conducted on a small sample of public administrations. Results may only fit the context of this case study. Nevertheless the value of the experiment reported relays more on the identification of a suitable experiment design and few valid statistical variables rather than on comprehensive results. In any case although results are not the major issue here, they still indicate no loss of productivity in passing to Open Source solution.
The Open Source applications here are just desktop applications for office automation, further analysis will consider a wider/different set of open tools.

8 Conclusions

In the past years OSS has proved to be a very reliable solution for many server applications. However, the claims about the benefits and advantages deriving from the use of OSS are seldom supported by empirical evidence or studies.
We considered the use of OSS for office automation tasks, for which no significant success case is known. We described a trial installation of the OpenOffice suite, reported the problems encountered and describe a possible experiment design and data analysis.
The data collection has been made with a non-invasive tool working in background.
The good results of the trial installation motivate the instantiation of a more extended experiment aimed at studying, analyzing, and evaluating the introduction of OpenOffice in public institutions.
We suggest an extensive analysis of the application interoperability before a transition process. This would help the customization of the new solution according to the end-user needs.

9 Acknowledgements

This work has been done with the help of the Autonome Provinz Bozen - Südtirol under the extreme valuable supervision of Dr. H. Ladurner. The authors would also like to acknowledge the help and support

provided by Dr. Hugo Leiter, Director of the EDP Unit of the Consortium of the Townships of the Province of Bolzano-Bozen.

References

1. Netcraft Survey, http://www.netcraft.com/survey/
2. P. Hochmuth, "Amazon cuts costs with Linux", *Computerworld,* http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,65674,00.html
3. M. Berger , "LinuxWorld: Amazon.com clicks with Linux",*ComputerWorld,* http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,73617,00.html
4. S. Shankland, "Verizon switches programmers to Linux", CNET News.com, http://news.com.com/2100-1001-949913.html
5. Free/Libre/Open Source Software: Survey and Study. June 2002. http://www.infonomics.nl/FLOSS/
6. The OpenOffice project, http://www.openoffice.org
7. W. Humprey, *Introduction to the Personal Software Process*, Addison-Wesley, 1997.
8. N.E. Fenton and S.H. Pfleeger, *Software Metrics: a Rigorous and Practical Approach,* Thomson Computer Press, 1994.