

An Overview of the Technical Challenges of Migration to Open Source Software

Mohamed Sarrab¹, Mahmoud Elbasir² Laila Elgamel³

Abstract— Open Source Software communities have successfully developed a lot of software pieces and suggested Open Source Software (OSS) as a solution to a lot of computing problems, although most of the software users only use proprietary applications. Open Source Software is a matter of software users' freedom to run or execute, copy or distribute, change or modify, study or improve. More precisely, it refers to the freedom to run or execute the program, for any purpose. The freedom to study or analysis how the program works, and adapt it to your needs. The freedom to copy and redistribute. The freedom to improve the program, and provide your released improved program to the public, to provide the benefits to the whole community. Many proprietary software users claim that open source software invites a lot of software problems such as, producing complex software and generating software code that could be prone to hackers and errors. The paper presents an overview of the real challenges to migrate to OSS. The technical challenges include performance, technical infrastructure, usability, integrity, support availability, security, information flow control, data migration, flexibility and ease of use, and management and maintenance of OSS.

Index Terms— Open Source Software, OSS, Performance, Usability, Integrity, Availability, Security, Flexibility and Ease of Use.

I. INTRODUCTION

The development in information and communication technology supports the existence of Open Source Software (OSS). The term OSS refers to execution and original source code of the software, whereas, users can change or modify the source code to make it work as they required. OSS may also include the right of software redistribution; In some case the OSS might be free, a distributor or developer might charge for services including

special training, installation, programming and technical support needed, etc. In general, the term OSS refers to software that is freely available, widely accessible and reusable [1-2].

The existing and increasing popularity of OSS has changed the software industry landscape in a dramatic way in latest years. The OSS is often seen as a possible solution to some of the challenges presently faced in various software communities, especially among different developing countries. Such challenges include controlling the piracy problem, getting greater level of control over software that is acquired and dealing with broader policy perspectives on the development of a domestic software industry. From the policy perspective of software An open source can be defined as a software licensing model in which the software source code is made available modified, redistributed and added, although often with certain restrictions. Although, updates, training and any other software services are possible to be provided by a range of entities under commercial arrangements.

OSS are usually, although not exclusively, developed through the effort of a group of people collaboration contribute in the components of the final software product. Software companies may also contributing by paid programmer time and programs developed in-house to the open source community. Proprietary or Commercial Software is the model in which the software developed under a commercial rules or policies, in other words the software is licensed for a fee to a customer in Binary, Object or Executable code (either directly or through channels). The commercial entity that provided the software usually provides updates, training, and other needed services by customers to use that software efficiently. The software source code might be made available to certain amount of the software users through special licensing or any other agreements, but is often not available and distributed to the general public, and might not be copied, changed or modified except in a manner provided for in such agreements or licensed stated[1,3].

Each of the above software models (Open source software and Commercial Software) can be translated to a viable business strategy for software companies, as well as supporting and providing customers real advantages. Solutions in the software are continually innovate, with software providers concentrating and improving considerably on emerging issues and problems such as addressing reliability, security and information flow

¹Communication and Information Research center, Sultan Qaboos University, P.O. 17, Al-Khodh, Muscat-123, Sultanate of Oman (phone: 968-24143698; fax: 968-24144336; e-mail: Sarrab@squ.edu.om)

²Center for Computing and Social Responsibility, De Montfort University, LE1 9BH, UK ;(e-mail: p06209705@myemail.dmu.ac.uk)

³Software Technology Research Laboratory, Faculty of Technology, De Montfort University, LE1 9BH, Leicester, UK ;(e-mail: Laila@dmu.ac.uk)

concerns. In fact, OSS has surprised many and different people in the software engineering world by gaining a reputation in the software reliability, efficiency and functionality [1,6]. The paper discusses the key challenges and issues to migrate to Free and Open Source Software. The paper reviewed the technical and non-technical issues and challenges of the migration process to the free and open source software. The paper is based on argumentative and philosophical approach. There is no formal research methods used in this type of research; however, personal knowledge and literature review is presented. The technical challenges includes [7-8].

II. PERFORMANCE

One of the main technical challenges of migration to OSS is the software performance. Does the software have a good reputation for performance? In industry, there are several open source software has a very good performance reputation i.e. Linux , Apache web server, GNU Compiler Collection (GCC), Samba, etc... Comparing the performance of Microsoft Windows and GNU/Linux on equivalent hardware has a history of argument claims and different results based on different assumptions. OSS is always trying to be performance competitive with closed software and in some circumstances it beats the competition. However, the performance benchmarks are very sensitive to the environment and assumptions, because in benchmarking, everything depends on the made of the assumptions and configuration. Consequently, high performance is a very big challenge for OSS due to the fact that it is always comparable with the closed software performance. For the migration to OSS, The OSS should always provide higher performance than commercial software [9].

III. TECHNICAL INFRASTRUCTURE.

One of the key barrier of the development of OSS is poor infrastructure. The development of OSS always requires a good Internet infrastructure such as reliable broadband access is very important to develop, disseminate and download OSS documentations and applications. It is also very important in creating OSS communities. Whereas OSS communities always serve as a support centre or help desk. Other factor of the ttechnical infrastructure is the lack of application or software manuals and local languages up to date documentations. Learning and supporting OSS may require a greater training input than equivalent proprietary software [10].

IV. USABILITY.

The usability of OSS is often considered as one reason for this limited distribution. Usability is typically described in terms of five characteristics: learnability, efficiency of use, memorability, error frequency and severity, and subjective satisfaction [11]. To understand the usability of current OSS the current software development process is needed to be examined. Usability is separate from the software utility and characteristics such as cost and reliability. In fact, many development tools and software applications such as editors and compilers, do not appear to represent a major OSS usability problem.

The usability of OSS needs more encouraging for greater involvement of end users and usability experts whereas they can give more user experience reports about the usability issues. Whereas, engaging normal users into the OSS projects development process projects supports the creation of a networked development community which may help in doing for usability what it has already done for functionality [12].

V. INTEGRITY.

The information security is more than just confidentiality. The key important is information security are integrity and availability. The integrity is more than just that information held in databases is correct. e.g. how can be known that the open source software not only does what it is supposed to do but also doesn't do what it is not supposed to do?

An integrity of open source software can be defined as the ability of a software system to withstand attacks to its security. The open source software integrity is an element of software assurance. Whereas, software assurance means that the software functions as intended. The software assurance is regularly discussed in the context of guaranteeing and ensuring that the open source code itself is more secure via the repeatable application of secure development practices of open source software. However, although there has been appropriate focus and increased interesting on reducing open source software vulnerabilities via secure development practices, which represents only one aspect of open source software assurance.

A process security is another key of consideration for software suppliers and customers. The process security is reference to the used processes to handle open source software components during their source obtaining, development and delivery, meanwhile a variety of possible attack vectors exist during the open source software lifecycle. A key problem in open source software systems is that although a software application at the beginning of execution may be verified or validate as authentic, during running, the flow of the execution can be redirected to externally injected malicious code using, i.e. buffer overflow exploit [13].

VI. SUPPORT AVAILABILITY.

Usually, the development of free software starts by small group of people trying to find a solution of specific problems, or needs to realize an idea. This group of people will at least solve some parts of the initial provided problem. Then initial implementation solution will be published on the Net. Others will hear and learn about it, find it interesting, and starting implement things related to it, or things those are needed for their own applications. In such way, the small group of people will be grown which will lead to great speed of development and support free software development in many different aspects such as learning, distribution and funding. Transaction costs are one of the key factors of OSS which are beyond free software developers' budgets. As can be noted that the development environments of commercial software are expensive and frequently beyond people's budgets.

Another factor is generating a reservoir of sufficiently and properly educated persons. People should be able to learn about IT and programming concepts. Specifically, people should be provided with the seed of knowledge and abilities required to extend and continue studies on their own. In fact, people are learning software development via pirated software. These software development environments demonstrate the effectiveness of every-day practical use by the free software community. The education system should include the teaching of the use of these software development environments instead of requiring or using commercial ones. The OSS development support can be provided by: Open source community without legal personality or formal organization, Organized community e.g. developers with legal personality and Commercial company which provides software support [14].

VII. SECURITY.

The advent of the low cost communication technologies and the high speed Internet provides an incredible new opportunity of creating a true information society. However, this advent raises several doubts about the information systems security. Developers of OSS projects are always concerned about security as the developers of proprietary software projects. However, the development processes might be very different and cannot adjudge which is inherently more secure [15].

Measuring security of OSS or proprietary software is very difficult. But clearly it can be said that OSS systems are not invulnerable from security bugs or flaws. The issue that appears to be an increasingly settled issue is that whether or not OSS reduces or improves software security. In fact, it has been reported in [16], that the OSS does not provide of guarantee any security benefit or limitation. In security the OSS is often superior to proprietary systems, whereas the security is still cited as one of the most important advantages of OSS. The detecting process of security errors, bugs and risks in OSS is fast and so is their eliminating process, due to the fact that in OSS the source code is made public [13]. Finally, it can be easily seen that OSS has a distinct advantage over proprietary systems, where it supports ease and quick identifying of potential security problems and correct them [17,19]

VIII. DATA MIGRATION.

Data is managed and stored by database applications. Virtually all public administrations have huge databases. Often this data is of critical importance and huge (financial) resources have been and are allocated to collect, organize, and maintain the data. It is important to divide the data into categories namely [13]

1. Data which can be discarded.
2. Data which is useful and in open format such as PDF or Postscript, or can be easily translated into open format. The cost should be considered.
3. Data which must be kept but which is in a legacy closed format which cannot be easily translated into an open

one. This data may need copies of the legacy software [20].

IX. FLEXIBILITY AND EASE OF USE

Flexibility and ease of use are two highly interrelated attributes.

- The Flexibility can be defined as the measures of how well software can be used to handle uncommon circumstances that the software wasn't originally designed for.
- The Ease of use can be subjective in nature depending on factors such as familiarization on the software, user experience, or even its direct impact to solve the problem. This criterion is usually associated with the user interface and how easy it is to use the software with the development environment. In my opinion, If it is possible to easily read and write the software code then it is easy to use software.

OSS products have an important advantage over most proprietary programs in both ease of use and flexibility: The different type of OSS programs can be modified to fit the user needs as much as needed for customer circumstance. But, having this advantage might require either programming skill or paying someone with such skills to do so. Another advantage is that some OSS programs are easier to be extended than others [20, 21].

X. MANAGEMENT AND MAINTENANCE OF OSS.

As OSS is becoming ever more important and widespread, thus its management and maintenance are becoming important issues as well. Generally, the software management and maintenance are resource and time consuming processes. Error and bugs detection and correction are the main duties in the software management and maintenance. The best way is to perform and finish these duties while the software development process and before the software release.

Hence, finding software parts where testing efforts must be concentrated can help project managers and software engineers, in restructuring, inspecting and testing efforts towards these critical software parts [13]. As a result, the software resources can be used more efficiently by the developers to deliver higher quality software products in a timely manner [22], due to the fact that applying equal verification and testing effort to all software system parts has become cost-prohibitive [23].

XI. CONCLUSION.

The philosophy of OSS is to be an alternatives approach for proprietary software and traditional software engineering processes. The research in OSS area is growing and governments over the world are trying to free their software products from the dominance of large software companies and external governments' policies. However, more investigations are needed to study the quality of each

approach end product, and overcome the issues and challenges of OSS. The products of OSS have advantage of being available to testers and users before the first complete release is issued. These testers and users are able to examine the OSS at its early stage and suggested any possible improvements. The paper has presented an overview of the main challenges to migrate to free and open source software (OSS) and suggested some important criteria for assessing migration challenges. The technical challenges to migrate to OSS were performance, technical infrastructure, usability, integrity, support availability, security, information flow control, data migration, flexibility, ease of use, management and maintenance of OSS. Despite these technical challenges to migrate to OSS, the impact of free open source is expected to be quite noticeable in the software technology and community as a whole. OSS supports novel development models, which have already been demonstrated to be especially well suited to efficiently take advantage of the work of developers spread across world. In generally, it has, a very positive impact as an enabler for the creation of new markets and business opportunities.

REFERENCES

- [1] Hasan. N. *Issues and Challenges in Open Source Software Environment with Special Reference to India*, TECHNOLOGY, POLICY AND INNOVATION, ICAL 2009.
- [2] Zuliani, P. & Succi, G. *Migrating public administrations to open source software*. Proceedings of eChallenges on Software Engineering. Vienna, Austria.2004.
- [3] Seow. GOH, *BSA Business Software Alliance, Open source and commercial software*. An in-depth analysis of the issues. www.bsa.org/usa/policy. September 2005.
- [4] Cassell. M, *5 Factors for Open Source Success*, 2011, Available at <http://www.govtech.com/pcio/5-Factors-for-Open-Source-Success-021111.html>
- [5] Chawner, B. (2004). “*Free/open source software: new opportunities, new challenges*”, paper presented at the 12th VALA Biennial Conference – Breaking boundaries: Integration & Interoperability, 2004, Available at www.vala.org.au/vala2004/2004pdfs/33chawn.pdf
- [6] Thomas. D, *Going Open Source Software in IT Opportunities and Challenges*, Journal Of Object Technology, Vol. 4, No. 2, March-April 2005. Available at <http://www.jot.fm/>
- [7] Open Source Initiative, Licenses by Name, Available at <http://opensource.org/licenses/alphabetical/>
- [8] Huda M. A. ElHag, *Migration to FOSS: Challenges*, 2009, Available at http://www.nctr.sd/images/stories/docs/Migration_to_FOSS.pdf
- [9] Metcalfe. R, *Top tips for selecting open source software, SS watch Open Source Software Advisory Service*,2012, Available at <http://www.oss-watch.ac.uk/resources/tips.xml>
- [10] Boateng. R, *Potential Benefits and Challenges of FOSS In Developing Countries*, PC Tech Technology/Business/Society, 2011.
- [11] Nielsen, J. (1993). *Usability Engineering*. Academic Press, Boston, ISBN 0-12-518405-0 (hardcover), 0-12-518406-9 (softcover). Japanese translation ISBN 4-8101-9009-9.
- [12] Nichols, D.M. & Twidale, M.B. *The usability of open source software*. *First Monday*, 8(1), January 2003. Available from: <http://firstmonday.org/article/view/1018/939>
- [13] Huda M. A. ElHag, Hisham M. Abushama, *Migration to FOSS: Readiness and Challenges*, Free Open Source Software (FOSS) Workshop, June 8th, 2009, Sudan.
- [14] Marit H, Kristian K. and Andreas P, *The Open Source Approach : Opportunities and Limitations with Respect to Security and Privacy*. *Computers & Security Journal*, 21(5), 461:471. February 2002.
- [15] Wheeler D. *Why open source software/free software (OSS/FS, FLOSS, or FOSS)? Look at the numbers!* Available from, 2005: http://www.dwheeler.com/oss_fs_why.html [Accessed: 05/06/2007].
- [16] Government information officers’ council – gitoc. *Using open-source software in the South African government: a proposed strategy compiled by GITOC 2003*. [Online]. Available from: <http://www.oss.gov.za> [Accessed: 16/04/2008].
- [17] Carlo D, Jesús M, Edmund H, Werner K, Bernard L, Ben L, Philippe A, Laurent C and Michel L (Working group on Libre Software). *Free Software / Open Source: Information Society Opportunities for Europe*. Version 1.2, April 2000. [Online]. Available from: eu.conecta.it/paper.pdf.
- [18] Anderson, R. *Security Engineering: A Guide to Building Dependable Distributed Systems*, Wiley Computer Publishing. 2001.
- [19] Sarrab M, *Runtime Verification of Information flow: Policy-Based Runtime Verification of Information Flow Control*, LAP LAMBERT Academic Publishing. 2011.
- [20] The University of Sheffield (USFD). *Computer Technology Institute & Press "Diophantus" (CTI). Open Source software usage by European Public Administrations (OSEPA). Technical efficiency guidelines for selecting between and among FOSS and proprietary SW solutions*. August 2011. [Online]. Available from: http://osepa.eu/pdeliverables/TAL17_3%20%202%20-%20Technical_Efficiency_Guidelines.pdf.
- [21] Wheeler D. *How to Evaluate Open Source Software / Free Software (OSS/FS) Programs*. [Online]. Available from: http://www.dwheeler.com/oss_fs_eval.html [Accessed: 05/08/ 2011].
- [22] Michael E, Chris E, Irene R and Brendan C. *Fault Detection and Prediction in an Open-Source Software Project*, Proceedings of the 5th International Conference on Predictor Models in Software Engineering, Vancouver, British Columbia, Canada. Article No. 17. 2009.
- [23] Fernando B and Rogério C. *Candidate Metrics for Object-Oriented Software within a Taxonomy Framework*, Originally published in Proceedings of AQUIS'93 Conference, Venice, Italy, October 1993. Selected for publication in the Journal of Systems and