

Case Study

The EOBI Comes of Age with FOSS

By Ayaz Uqaili, Project Director/IT Head, EOBI

The Employees Old Age Benefits Institution or EOBI's objective is to provide social benefits to old age employees of the private sector, autonomous and semi autonomous bodies in the form of pensions and grants. For this, it collects contribution from employers. The main data intensive tasks that are performed at EOBI are:

- a) Registration of employers,
- b) Registration of insured employees who become future pensioners,

if scenarios were considered. It should be pointed out that the proprietary OS implementation are seldom free of some 'what if' nightmares becoming true in the post acquisition phase. Once the approval was in place, the project kicked off with purchase of about 200 workstations installed with Ubuntu 8.10 across more than 60 office sites for field operations officers. The demographic of the officers was an average age of 35, having with very little or no knowledge of computers earlier. This perhaps was important since there was

If FOSS is free, why is here a cost associated to it?

True, FOSS is free but there is a cost associated with its customization, configuration, integration and technical support. These support services are paid to professionals or consultancy companies, but there is no cost associated with software licensing, which is a major component of any non-FOSS solution. Also, non-FOSS solutions have these costs in addition to licensing.

- c) Registration of pensioners and
- d) Reconciliation of contribution collection with the authorized banks and disbursement details. The span of activity covers 31 regional offices and 32 field offices across Pakistan.

All of these activities had been computerized in various shapes and forms over the past many years. The last instance deployed in 2008 was a decentralized and distributed system over Microsoft the Windows Operating System and an application written over Microsoft Foxpro. This setup had problems of both OS reinstalls, frequent stalls and virus attacks and application issues hindering data entry such as frequent data and file corruption. In the absence of a reliable dataset and timely consolidation, the management decided to deploy a centralized application across Pakistan. At this point, a decision had to be taken: first time adoption of FOSS versus the prevalent proprietary solutions considered which had been tried and tested. Getting approval for FOSS wasn't a walk in the park.

A series of technical committee meetings and evaluations were required where all sorts of fears and 'what

no predisposition towards a proprietary OS which could have only helped in the adoption Linux Ubuntu. Another cost saving attempt was made by opting for refurbished PCs however in government institutions, there is no practice of refurbished hardware acquisition as a matter of practice.

Software development was undertaken in a platform independent Java with the help of Systems Limited, with server side Linux Ubuntu and browser-based web client access. An application server initially used during development was IBM Bea Web Logic but because of high associated costs, the application was switched to the FOSS alternative, TomCat, resulting in substantial saving.

By August 2008, a pilot run had started at 5 locations in Karachi where the fully functional application for regular EOBI operations was put into effect. Meanwhile, a data center was established to support the centralized operations with Wateen's WiMax service. The cut off for the complete transition from the existing setup to the new application was January 22, 2009. We managed to successfully meet the deadline and all

EOBI office sites came on board, using the newly developed application. The results have been impressive.

To date, we have had no hassles with system reinstalls, OS stalls, file corruption due to viruses and this success is reinforced each day with a visible decline in the overall number of support queries.

In parallel, the Karachi-based IT department began experimenting with other FOSS utilities. The manual tracking and movement of files was replaced with jTrack. This reduced the time for locating and finding status of files from the norm of one or two days, down to a few minutes. A PHP-based Intranet portal for the dissemination of information, circulars, helpdesk is also in place allowing for two-way communication between the management and its employees. Enterprise instant messaging has been enabled by OpenFire XMPP.

Other FOSS projects in the pipeline include the conversion of database Postgres or MySQL, conversion of ASP pages to PHP, implementation of a Document Management System, LDAP directory services, mail server PostFix, self-managed call center Asterisk server and enabling VOIP for interoffice communication.

What are the lessons learned by EOBI effort?

Firstly the understanding, appreciation and belief that FOSS is not a fallacy.

It is real and can be used effectively as a viable alternative to proprietary solutions. The major impediment to FOSS acceptability is lack of confidence and awareness about FOSS. Fear of failure and conformist attitude can be removed with projection of FOSS success stories and strong academic link and public sector initiatives to develop the required talent.

One major argument cited against the widespread adoption of FOSS when it comes to large-scale projects is the emphasis on FOSS itself as opposed to the system itself. The success or failure of projects has much to do with the quality of the system being developed, regardless of its underlying operating system or database. Enterprises in Pakistan are riddled with substandard desktop and web applications but that doesn't mean that these results are due to the choice of the proprietary operating system. As the world moves towards hosted applications, the impact of FOSS


and non-FOSS is getting increasingly lesser as for clients (mostly end-users), the results would be transparent. Also, when it comes to public sector decision making, the management must keep in mind an OS that looks and behaves over a personal laptop must not automatically become the only choice for departmental projects too.

What is software freedom?

Free software is a matter of liberty, not price. To understand the concept, you should think of free as in free speech.

Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software. More precisely, it refers to four kinds of freedom for the software users:

- (Freedom 0): The freedom to run the program, for any purpose.
- (Freedom 1): The freedom to study how the program works and adapt it to your needs. Access to the source code is a precondition for this.
- (Freedom 2): The freedom to redistribute copies so you can help your neighbor.
- (Freedom 3): The freedom to improve the program and release your improvements as well as modified versions in general, to the public so that the entire community benefits. Access to the source code is a precondition for this.

<http://www.gnu.org/philosophy/free-sw.html>
Note: Terms 'Free software' and 'Open Source' are different. Free software refers to ideals and values of freedom in use of software. Open Source, on the other hand, is more of a development model that provides source code with licenses that may or may not guarantee the above freedoms. To read about these differences and their philosophies, visit GNU.org and open-source.org. You can contact Ayaz at: ayaz@computer.org for more details on the EOBI project 

About the Author:

Nizar Diamond Ali is a Karachi based freelance writer. He has been writing for a number of national and international publications for the past 8 years. He has a Masters degree in Computer Science and is currently working with an international bank as Manager Service Quality. Nizar can be reached at, nizar.ali@gmail.com