

Chapter 5.9

Open Source Software: A Developing Country View

Jennifer Papin-Ramcharan

The University of the West Indies – St. Augustine Campus, Trinidad and Tobago

Frank Soodeen

The University of the West Indies – St. Augustine Campus, Trinidad and Tobago

ABSTRACT

This chapter presents issues that relate to developing countries' use of open source software (OSS) and the experience of these countries with OSS. Here the terms open source software (OSS), free/libre and open source software (FLOSS) and free software (FS) are used interchangeably. It describes the benefits of FLOSS including its superior quality and stability. Challenges to FLOSS use particularly for developing countries are described. It indicates that despite the greater benefits to developing countries of technology transfer of software development skills and the fostering of information and communication technology (ICT) innovation, the initial cost of acquiring FLOSS has been the key motivation for many developing countries adopting FLOSS solutions. It illustrates this by looking at the experience of a university in a developing country, The University of the West Indies, St. Augustine Campus in Trinidad and Tobago. Strategies for developing countries to benefit "fully" from FLOSS are presented including the implementation of formal organized programmes to educate and build awareness of

FLOSS. The authors hope that by understanding some of the developing country issues that relate to OSS, solutions can be found. These countries could then fully benefit from OSS use, resulting in an increase in size of the global FLOSS development community that could potentially improve the quality of FLOSS and indeed all software.

INTRODUCTION

Open source software (OSS) is understood by many to mean software or computer programs where the source code is distributed and can be modified without payment of any fee by other programmers. The term OSS first came into use in 1998 and is attributed to Eric Raymond (Feller & Fitzgerald, 2002). The Open Source Initiative (OSI) has been formed to promote the use of OSS in the commercial world (www.opensource.org/).

The terminology related to software that is released with its source code and is modifiable and distributable without payment and then developed by a group of users or community can be confusing. For example the literal meaning of *open*

source implies access to the source code, without necessarily implying permission to modify and distribute. Also, the Free Software Foundation (FSF) (founded in 1985) which predates the OSI refers to these programs not as OSS but as *free software* (www.fsf.org/). The term *free software* was created by Richard Stallman where *free* refers to the freedoms to use, modify, and distribute the programs and does not have anything to do with the cost of acquiring the software. Therefore, *free* software does not necessarily mean *zero cost*, and *open source* does not just mean access to the source code.

The differences between free software and OSS have been well documented (Fuggetta, 2003). Stallman (2002) gives details about these differences at www.gnu.org/philosophy/free-software-for-freedom.html. What really then defines software as OSS? Generally, OSS is a software product distributed by license, which conforms to the Open Source Definition¹. The best known of these licenses are the GNU General Public License (GPL) and the Berkeley Software Distribution (BSD) license. Unlike traditional commercial or proprietary software (e.g., Microsoft Word, Windows XP, or Internet Explorer), these licenses permit OSS to be freely used, modified, and re-distributed. The source code for these programs must also be freely accessible.

The term *free/libre open source software* (FLOSS) is used to refer to both free and open source software and was first coined in 2002 by Rihab Ghosh in a study undertaken for the University of Maastricht (Ghosh, Glott, Kreiger, & Robles, 2002). Libre here is the French word for liberty making clear the *free* as in freedom and not free as in “no cost.” It is also common to use the term FOSS (free/open source software) for such programs.

Many countries do not have the luxury of debating the philosophical differences between the OSS or free software movement and so are content to use the all encompassing term of FLOSS (see e.g., www.floscaribbean.org/). For

the purposes of this discussion the terms Free Software, Open Source Software and FLOSS are used interchangeably².

The development of FLOSS has often been contrasted to that of proprietary software. FLOSS has primarily been developed by individuals who volunteer their time to work on FLOSS projects. Normally, modified versions of the source code are posted on the Internet and are available for free to anyone who wants to use or modify it further. Thus a community of developers is created all working on modifications, bug fixing, and customizations of the initial code. An extensive explanation and analysis of OSS development can be found in Feller and Fitzgerald (2002).

The number of open source software projects can be gleaned by visiting <http://sourceforge.net/index.php> where there are over a hundred thousand such projects registered to date. Thus, FLOSS is not a fad or fringe phenomenon. It is important to note that FLOSS has penetrated major markets in countries worldwide. Indeed, some open source products like Linux and Apache are market leaders globally, and major ICT companies like IBM, Sun, and Oracle have adopted the open source model (Bruggink, 2003). In some countries, governments have even made the decision to support the use of FLOSS (Brod, 2003; Evans & Reddy, 2003).

Because of its free cost and its freedoms, FLOSS should be an obvious choice for widespread use in developing countries. In fact, these countries should be virgin territory for FLOSS deployment. What do the developing countries themselves say and what has been their experience? This chapter presents the point of view and experience of developing countries with FLOSS.

7 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/open-source-software/29487

Related Content

Using Timed Automata for Modeling the Clocks of Distributed Embedded Systems

Guillermo Rodriguez-Navas, Julian Proenza, Hans Hansson and Paul Pettersson (2010). *Behavioral Modeling for Embedded Systems and Technologies: Applications for Design and Implementation* (pp. 172-193).

www.irma-international.org/chapter/using-timed-automata-modeling-clocks/36342

Refactoring Flash Embedding Methods

Ming Ying and James Miller (2012). *International Journal of Systems and Service-Oriented Engineering* (pp. 26-40).

www.irma-international.org/article/refactoring-flash-embedding-methods/79237

An Investigation into the Impact of Ethnicity and Culture on the Motivation for using Facebook for Academics and Socialization in Guam

Sathasivam Mathiyalakan, Kevin K.W. Ho, George E. Heilman and Wai K. Law (2017). *International Journal of Systems and Service-Oriented Engineering* (pp. 1-21).

www.irma-international.org/article/an-investigation-into-the-impact-of-ethnicity-and-culture-on-the-motivation-for-using-facebook-for-academics-and-socialization-in-guam/201205

The Impact of Media Richness on the Usage of Web 2.0 Services for Knowledge Transfer

Albert Gyamfi (2018). *Application Development and Design: Concepts, Methodologies, Tools, and Applications* (pp. 928-946).

www.irma-international.org/chapter/the-impact-of-media-richness-on-the-usage-of-web-20-services-for-knowledge-transfer/188241

Plant Leaf Disease Detection Using CNN Algorithm

Deepalakshmi P., Prudhvi Krishna T., Siri Chandana S., Lavanya K. and Parvathaneni Naga Srinivasu (2021). *International Journal of Information System Modeling and Design* (pp. 1-21).

www.irma-international.org/article/plant-leaf-disease-detection-using-cnn-algorithm/273224