A Review of Migration Processes to Open Source Software

Razieh Sobhani, Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran
Habib Seifzadeh, Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran & Big Data Research Center, Najafabad Branch, Islamic Azad University, Najafabad, Iran
Taghi Javdani Gandomani, Department of Computer Engineering, Boroujen Branch, Islamic Azad University, Boroujen, Iran

ABSTRACT

With regard to the increasing spread of information technology in the world, access to this technology is of great importance. Due to their competitive advantages, open source software are more popular than the close ones and they are more widely used, as well. Having an appropriate process for migrating, identifying and ranking activities to prepare and perform the migration design in the organization, prevents the failure of the performance in organizations. Today, there are lots of migration processes in review of literature and this will confuse the managers to choose the most appropriate process for their organization. This article studies the presented processes of migration to open source software in the world and by presenting novel factors. It also attempts to introduce an appropriate framework to select the most efficient process in migration to open source software. So that organizations do not have a concern to choose the best migration process and IT managers are able to select the appropriate process for their organizations quickly and with no confusion.

KEYWORDS

Migration Model, Migration Process, Migration to Open Source Software, Open Source Software

1. INTRODUCTION

Today, unpredicted and hidden expenses of software are known as “software crisis”. Software maintenance cost is considered as the main component of software engineering costs and according to the studies, between 50 to 90 percent of software projects expenses are spent on this cost (de Vasconcelos, Kimble, Carreteiro, & Rocha, 2017; Seifzadeh, Abolhassani, & Moshkenani, 2013). Therefore, software maintenance and software systems promotion issues are of great importance. As a solution, numerous studies have been conducted on the subject of software migration from migration to service-oriented architectures (Athanasopoulos, 2017; Fuhr, Horn, Riediger, & Winter, 2013) to migration to cloud (Bisbal et al., 1997; Botto-Tobar, Ramirez-Anormaliza, Cevallos-Torres, & Cevallos-Ayon, 2017; Fowley, Elango, Magar, & Pahl, 2017; Gholami, Daneshgar, Beydoun, & Rabhi, 2017; Gholami, Daneshgar, Low, & Beydoun, 2016; Kesserwan, Dssouli, & Bentahar, 2018; Mohagheghi & Sæther, 2011). One of the types of migration is the migration to open source software.

DOI: 10.4018/IJOSSP.2018010102

Copyright © 2018, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
The monopoly and information dominance of close source software providers have led countries to follow the approach of using open source software only in private sectors (Pouresmaiel, 2014). Before any measure can be taken in order to migrate software systems, that migration process should be justified and adopted by target organization. While there are extensive studies in the field of migration adoption (Chau & Tam, 1997; Hauge, Maddocks, 2005), the main idea behind open source software is to share knowledge, raise knowledge and enable others to improve what has been shared (GITOC, 2003). Choosing an optimal process for migrating to open source software can partly guarantee the success of migration. In the migration process, the steps should be precisely identified and described so that there would be no ambiguity in practical performance of the design. Various designs are presented in this regard. Some of them are accurate but it is not clear how to do and test the tasks which are depicted in each step. The problems which organizations might be encountered with during the migration process of open source software are not highlighted in these processes and no suggestions are offered to prevent them (Mtsweni & Biermann, 2008). Some processes are theoretically useful but cannot be practiced. In some others, the migration is initiated first and then a report of its successful process is presented, such as the Munich Municipality Migration Project, known as LiNUX (Silic & Back, 2017). The planning of the project started from 2001 to 2003 and the migration process has begun since 2005, and finally, around 14000 systems have been migrated to LiNUX and open source until 2014 (Scholl, Jurisch, Krcmar, & Scholl, 2015). Having studied more than 30 articles in the field of migration to open source software, we aim to compare the existing methods and consequently, take some steps to simplify the selection of the appropriate process for migrating to open source software.

The structure of the article is as follows: the second section is the review of literature which contains 6 models of migration process. The third section is the methodology which concerns successful migration factors and presents a framework to choose the best process. Section 4 discusses the selection of an appropriate migration process with regard to the type of organization. Finally, and in the last section, the article is concluded.

2. REVIEW OF LITERATURE

In this paper, we review and compare the proposed models in the field of migration to open source. Among all the research in this area, we have succeeded in finding 6 process models that consider the migration as a process and categorize the migration operations into phases and activities, which we will discuss later.

Lachniet (Lachniet, 2004) has presented a framework for migrating to open source software. This framework divides the tasks of pre-migration to 3 parts: executive tasks, tasks of software development and IT tasks. Executive tasks primarily focus on providing senior managers’ support of migration to open source. Developing a high-level policy in support of senior management, making
Related Content

Curious Exceptions?: Open Source Software and “Open” Technology
[www.igi-global.com/article/curious-exceptions/101217?camid=4v1a](www.igi-global.com/article/curious-exceptions/101217?camid=4v1a)

Heuristic Guided Selective Path Exploration for Loop Structure in Coverage Testing
[www.igi-global.com/article/heuristic-guided-selective-path-exploration-for-loop-structure-in-coverage-testing/196568?camid=4v1a](www.igi-global.com/article/heuristic-guided-selective-path-exploration-for-loop-structure-in-coverage-testing/196568?camid=4v1a)
Understanding the Development of Free E-Commerce/E-Business Software: A Resource-Based View
www.igi-global.com/chapter/understanding-development-free-commerce-business/10087?camid=4v1a

Are Developers Fixing Their Own Bugs?: Tracing Bug-Fixing and Bug-Seeding Committers
www.igi-global.com/article/developers-fixing-their-own-bugs/62098?camid=4v1a