Open Source Software Evolution: A Systematic Literature Review (Part 2)

Kuljit Kaur Chahal, Department of Computer Science, Guru Nanak Dev University, Punjab, India
Munish Saini, Department of Computer Science, Guru Nanak Dev University, Punjab, India

ABSTRACT

This paper presents the results of a systematic literature review conducted to understand the Open Source Software (OSS) development process on the basis of evidence found in the empirical research studies. The study targets the OSS project evolution research papers to understand the methods and techniques employed for analysing the OSS evolution process. Our results suggest that there is lack of a uniform approach to analyse and interpret the results. The use of prediction techniques that just extrapolate the historic trends into the future should be a conscious task as it is observed that there are no long-term correlations in data of such systems. OSS evolution as a research area is still in nascent stage. Even after a number of empirical studies, the field has failed to establish a theory. There is need to formalize the field as a systematic and formal approach can produce better software.

KEYWORDS
ARIMA Modelling, Automation Support, Co-Evolution, OSS Prediction, Programming Languages, Software Evolution Theory, Software Reuse

INTRODUCTION

Open Source Software (OSS) has emerged as a new discipline of software development as an answer to the traditional approach that produced software products that were not only costly and complex, but the complete process was full of constraints as well. In the traditional proprietary model of software development, the source code was kept as a top secret and never shared with anyone outside the development team. In contrast, the OSS development approach makes the source code freely available to the worldwide community of users and developers. Open source software is an important phenomenon, and needs to be deeply studied and understood.

Understanding software evolution in general and open source software evolution in particular has been of wide interest in the recent past. Understanding the way a software system evolves can help in better software project management, improvement of software development as well as evolution processes, and advancing the software evolution theory. Therefore, stakeholders of OSS projects are interested in understanding the evolutionary process of OSS projects. A wide range of research investigations have analysed open source software (OSS) evolution from different points of view. A wide variety of methods, tools, and techniques have been employed to study the phenomenon.

DOI: 10.4018/IJOSSP.2016010102

Copyright © 2016, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
This research study is an attempt to systematically summarize the empirical evidence obtained on these techniques from the existing literature and studies. It reports a review of the studies between the period of 1997 and 2016. In order to achieve this aim we extensively searched through nine online digital databases. We identified 190 studies to answer the research questions. The primary studies were selected according to the quality assessment of the studies and their relevance. Then the data from these studies was extracted, and synthesized with the research questions.

This paper is in continuation of a preceding one (Chahal and Saini, 2016) in which details of the review methodology for the systematic literature review are discussed. The preceding paper presents the review results related to OSS evolution analysis only (the first research question). In this paper, we continue with the discussion of numerous other methods used by researchers for studying OSS evolution. Our results suggest that there is not enough evidence on the theory of software evolution.

The remainder of the paper is organized as follows: Section 2 briefly describes the steps followed for the systematic review. Section 3 discusses the review findings. A comparison of the current study with other similar works is given in Section 4 followed by conclusions and suggestions on possible future work in Section 5.

THE REVIEW METHODOLOGY

A systematic review provides the means to identify, evaluate, and interpret all available research relevant to a particular research question, and phenomenon of interest. We briefly discuss the review process in this section. For a detailed discussion, the interested readers may refer to the preceding paper of the current one (see footnote at page 1).

The Review Process

1. Setting the Research Questions: This review aims to provide and assess the empirical evidence obtained from the studies for understanding OSS evolution process. We classified research questions broadly into nine categories for the purpose of discussion. The aim is to summarize, analyze and assess the empirical evidence regarding: (1) OSS evolution analysis techniques (2) OSS evolution prediction techniques (3) comparison between the OSS and the CSS (Closed Source Software) evolution (4) evolution of artifacts other than source code (5) usage of programming languages as OSS projects evolve (6) usage of the software reuse paradigm (7) evolution of community contribution (8) automation support available for OSS evolution analysis, and (9) the theory of software evolution;

2. Search strategy and study selection: A search string was formed by incorporating alternative terms and synonyms for the search terms derived from the research questions. These search terms were then combined using Boolean expressions ‘OR’, and ‘AND’. The following general search string was formed for identification of primary studies:

\[[\text{OSS} \text{ | Open Source | Libre/Free | FLOSS}] \text{ AND } [\text{Software | System(s) | Project(s) | Process}] \text{ AND } [\text{Evolution | Maintenance | Change(s) | Commit(s)}] \text{ AND } [\text{Analysis | prediction | Forecasting | Visualization}] \text{ AND/OR } [\text{Developer | Community | Contributor}]\]

Using the derived search string, a set of primary studies were found from the following online digital databases: Scopus, IEEE Xplore, ACM digital library, ISI web of knowledge, SpringerLink, ScienceDirect, Flosshub, Wiley online library, and the Open University. This set was augmented by
19 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the product's webpage:

www.igi-global.com/article/open-source-software-evolution/179924?camid=4v1


Recommend this product to your librarian:

www.igi-global.com/e-resources/library-recommendation/?id=2

Related Content

Free Assistive Technology Software for Persons with Motor Disabilities
www.igi-global.com/chapter/free-assistive-technology-software-for-persons-with-motor-disabilities/120932?camid=4v1a

Hacker Culture and the FLOSS Innovation
www.igi-global.com/article/hacker-culture-and-the-floss-innovation/101204?camid=4v1a
Riki: A System for Knowledge Transfer and Reuse in Software Engineering Projects
Jörg Rech, Eric Ras and Björn Decker (2007). *Open Source for Knowledge and Learning Management: Strategies Beyond Tools* (pp. 52-121).
www.igi-global.com/chapter/riki-system-knowledge-transfer-reuse/27809?camid=4v1a

An Empirical Study of Open Source Software Usability: The Industrial Perspective
www.igi-global.com/article/empirical-study-open-source-software/54243?camid=4v1a