Chapter 5

Investigating the Success of OSS Software Projects

Amir Hossein Ghapanchi
Griffith University, Australia

ABSTRACT
Whereas there are several instances of Open Source Software (OSS) projects that have achieved huge success in the market, a high failure rate has been reported for OSS projects. This study conducts a literature survey to gain insight into existing studies on the success of OSS projects. More specifically, this study seeks to extract the critical success factors for OSS projects. Based on the literature survey in this study, the authors found determinants of success in OSS projects and classified them into three broad categories of project traits, product traits, and network structure. These findings have important implications for both the OSS research community and OSS practitioners.

1. INTRODUCTION
Although there are several successful instances of open source projects that have achieved a huge success in the market, it has been observed that most OSS projects are abandoned after a while and experience failure (Colazo & Fang 2009; Chengalur-Smith & Sidorova 2003). Addressing the current situation of a high failure rate amongst OSS projects, the focus of the present study is on helping OSS projects become aware of the factors that impact their success.

In response to the current situation of a high failure rate amongst OSS projects, our focus herein is on helping OSS projects increase their chance of success by informing them of the factors that may impact their success.

This paper provides an overview of current state-of-the-art research in OSS “success” literature. The aim of this research is to help OSS projects succeed by developing a framework that represents antecedents of “success” in OSS projects according to the literature. Moreover, in order to make more sense of the list of antecedents, this paper aims to bundle the antecedents into meaningful categories. Hence the research question guiding this study is as below:

RQ: What factors impact success in OSS projects, and how can we group them?

DOI: 10.4018/978-1-4666-6485-2.ch005

Copyright © 2015, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
Given the current situation regarding the high failure rate of OSS projects and also the proliferating reliance of firms on OSS applications (Sen 2007), seeking answers for this question is important. The answers to this question can inform many OSS practitioners including OSS project managers, potential OSS adopters, and potential OSS sponsors of the factors that determine the future success of an OSS project.

This paper is organized as follows. Section 2 contains a literature review on OSS success. Section 3 presents the research methodology. Section 4 presents the results of the research, i.e. a framework for antecedents to success in OSS projects. Section 5 presents the concluding remarks.

2. LITERATURE REVIEW

2.1 Open Source Software

Utilising novel intellectual property laws, a new software development and distribution paradigm has emerged over last two decades, which in February 1998 was termed open source software development (Midha 2007). Unlike proprietary software in which the program’s source code is a trade secret and is protected by law, in OSS the source code is publicly available for anyone who would like to see it. OSS products are developed under an open source licence that permits their users to observe, modify and redistribute the program’s source code (Open Source Initiative 2005).

A typical OSS project starts with what (Raymond 1999) calls “scratching a developer’s personal itch”. An OSS project initiator who has a software idea starts writing the code. Since the community is intended to be able to see the software, it is released under a license that allows the community to see the source code and use the software. The developers in the community can contact the project administrator(s) and join the development team. Moreover, the community users can communicate with the project team and ask for help, or request new features or report a bug in the system, or even contribute to the development. As a result of this evolution, OSS is said to meet user needs better than traditional closed-source software (Loshin 2005). Being involved with development process, users may be more satisfied with open source software (Midha 2007).

Unlike closed source software (CSS) in which the program’s source code is a trade secret and is protected by law, in OSS, the source code is publicly available for anyone who would like to see it. However, this cannot fully define OSS because even though access to source code is normally open in OSS applications, there may be some exceptions like software developed under Microsoft’s share source initiative. Normally OSS applications are developed by volunteers rather than paid developers, but there are some contradictory cases like Linux which is developed by volunteers as well as paid developers. Hence, “a software which is developed by volunteer developers” can also not completely define OSS. Open Source Initiative (OSI) defines OSS as software released under a license approved by OSI (Open Source Initiative 2005). Researchers agree on the definition of OSS proposed by OSI. OSS could be free or commercial; but, the focus of this study will be on free OSS.

OSS development has been an important area of research because of the large number of OSS applications that have been highly successful and are being used by millions of users. Apache, Mozilla Firefox, Linux, Unix and Perl are examples of such software. Table 1 shows some application areas of OSS along with examples of popular open source packages in each area.

2.2 OSS History

OSS originated in the early 1960s, when key foundations of Internet were being constructed in academic settings like MIT and Berkeley. That was probably of early attempts to share software source code by developers. OSS was then termed
Related Content

IIR Filters to Sampling Rate Conversion
[www.igi-global.com/chapter/iir-filters-sampling-rate-conversion/27214?camid=4v1a](www.igi-global.com/chapter/iir-filters-sampling-rate-conversion/27214?camid=4v1a)

Architectures for Cognitive and A-Life Agents
[www.igi-global.com/chapter/architectures-cognitive-life-agents/24143?camid=4v1a](www.igi-global.com/chapter/architectures-cognitive-life-agents/24143?camid=4v1a)

A Source Code Plagiarism Detecting Method Using Sequence Alignment with Abstract Syntax Tree Elements

A Case Study: Mobile Service Migration Based Traffic Jam Detection
[www.igi-global.com/article/a-case-study/207349?camid=4v1a](www.igi-global.com/article/a-case-study/207349?camid=4v1a)