The Impact of Project License and Operating System on the Effectiveness of the Defect-Fixing Process in Open Source Software Projects

Abstract

Open source software (OSS) products have been widely adopted by commercial as well as government organizations. However, despite their increased adoption, many OSS projects still fail in responding to users’ quality needs such as resolving software defects. Hence, this paper investigates the responsiveness of OSS projects to users’ needs in terms of resolving software defects. To do so, we develop and test a model of antecedents to the effectiveness of the defect-fixing process for OSS projects. Data gathered for this study from 1481 OSS projects confirms that OSS projects that apply a less restrictive license are less dependent on their team to operate their defect-fixing process. It is also demonstrated that OSS projects developed to run on a broader range of operating systems are more likely to have an effective defect-fixing. The study provides practitioners with insightful recommendations on project characteristics and defect-fixing effectiveness.

Key Words: Open source software, defect-fixing effectiveness, OSS license, operating system.

1. Introduction

Open source software (OSS) have been widely embraced by commercial and government organizations (Carillo and Okuli, 2008). According to the 2020 FLOSS Roadmap¹, by 2011 around 50% of Global 2000 IT organizations will have adopted at least one open source software solution.

In spite of the increasing adoption of OSS products, many OSS projects still fail in the early stages of development (Chengalur-Smith and Sidorova, 2003). According to Krishnamurthy (2002), 63% of

---

OSS projects on Sourceforge.net, the largest OSS host in the world, fail due to their inability to attract the attention of developers and users in the community. Various factors have been reported in prior studies as antecedents to OSS project success such as project sponsorship (Stewart et al., 2006), project audience and project topic (Crowston and Scozzi, 2002), software quality, user satisfaction, and service quality (Lee et al., 2009), software design quality (Capra et al., 2008), and team ideology (Stewart and Gosain, 2006a).

Effective defect-fixing has been reported as one indicator of success in OSS projects by prior research (Crowston et al., 2003). One key reason is that an effective defect-fixing is tied to users’ perceptions of project quality, activity and value (Midha et al., 2010; Mockus and Weiss, 2008). Therefore, this study seeks to investigate the antecedents to defect-fixing effectiveness.

In order to explore these antecedents, this study develops and tests a set of hypotheses regarding the impact of such factors on defect-fixing effectiveness. This paper focuses on two project traits: the project’s license and operating system. There are two key reasons for this. First, license and operating system are two of the first project-configuration factors that project managers decide on at the beginning of the project, and thus their practical implication are of high importance to project leaders. Second, license and operating systems chosen might affect the extent to which an OSS project attracts volunteers (Stewart et al., 2005). If the selection of license and operating system influences defect-fixing effectiveness, then we should ask a practically and academically question of: “how do OSS projects’ characteristics influence the effectiveness of the defect-fixing process?”

Having more skilled developers can be seen as a core competency for OSS projects (Crowston and Scozzi, 2002). However, attracting developers to join the project team is a critical challenge for OSS projects since unlike closed-source software development OSS projects are not commonly driven by direct monetary profits (Lakhani and Wolf, 2003). Therefore, the first aim of this study is to examine how OSS projects can reduce their dependence on team size to operate a more effective defect-fixing process. The second aim of this paper is the investigation of the impact of a project’s operating system on the effectiveness of its defect-fixing process.
The remainder of this paper is structured as follows. The next section reviews the literature and provides some background information on the defect-fixing process in OSS projects. The research model and theoretical justifications for the relationships proposed are presented in Section 3. The research methodology is presented in Section 4. The findings of the paper are depicted in Section 5. Discussion of the results is presented in Section 6. Section 7 presents conclusions and limitations of the research.

2. Literature Review

Inspired by novel intellectual property laws, a new software development and distribution practice has emerged over the last two decades termed ‘Open Source Software’ (OSS) development by a group of “free software” supporters, including Eric S. Raymond and Tim O’Reilly (Feller and Fitzgerald, 2001).

Success has been studied by researchers for decades. Researchers have introduced several drivers of success such as user satisfaction and business value (Kamhawi and Gunasekaran, 2009), knowledge management (Kant and Singh, 2009, McCarthy and McGrath, 2008), effective communication and information processing (King, 2005), customer relationship management (Sharma et al., 2008) and so on.

OSS literature has studied various aspects of success such as project activity (e.g. Liu, 2008; Stewart et al., 2006; Colazo, 2007; Grewal et al., 2006), user interest (e.g. Subramaniam et al., 2009; Stewart et al., 2006; 2002; Midha, 2007), and project performance (e.g. Liu, 2008; Guiri et al., 2004; Hahn and Zhang, 2005).

One of the most important areas of study in information system development is software quality. A significant dimension of software quality is responsiveness to user/customer needs (Hsu et al., 2008). According to Hsu et al. (2008), responsiveness to customer needs is considered an external quality
dimension, identified by individual or organizational users. Thus, this paper studies the responsiveness of OSS project teams to customer needs in terms of software defect removal.

OSS projects appear to have a never-ending process of defect-fixing. The defect-fixing is a process in which software defects are handled and fixed to improve the software quality. In OSS literature, a number of researchers have studied defect-fixing effectiveness. Herbsleb and Mockus (2003) found that the progress in defect-fixing influences the positive outcomes of OSS projects. Stewart and Gosain (2006b) showed that the higher the number of closed defects, the higher the perceived effectiveness of OSS projects. Stewart and Gosain (2006a) used the percentage of defect reports completed as an indicator of OSS project effectiveness and demonstrated that communication quality and team effort impact the quantity of bug reports completed. Furthermore, through three case studies, Garousi (2009) found that various OSS projects exhibit different levels of responsiveness to the defect-fixing process due to factors like activity level. Crowston and Scozzi (2008) also examined the coordination practices involved in the defect-fixing process of OSS projects.

Observing a commercial software company, Crowston (1997) described the defect fixing process in traditional closed-source software (CSS) development as following: the process starts by a user who finds a defect in the software. The defect is directed to the software company customer centre. If its personnel cannot solve the problem through a database of known defects, they forward it to the manager responsible for the software module associated with the defect. The manager then assigns the defect to a developer to fix. Subsequently, the defect is fixed and shares with other developers responsible for modules that might be influenced. When positive feedback is received from those developers, the change is applied into the source lines of code. After undertaking some tests, the changed module is sent to integration manager. Having approved the module, the integration manager recompiles the system, test the whole system, and link it to the system.

In OSS development, defect fixing process is operated through a defect tracking system. Having a tracking system for defect-fixing is common in CSS projects too, but unlike CSS projects in OSS projects all information in defect-tracking system is publically available to everyone.
Typically, there are four major tasks involved in the defect fixing process of OSS development that are common between almost all of them including reporting a defect, assigning the defect, fixing the defect, and closing the defect (See Figure 1). These steps are in line with prior OSS research on defect fixing process (Crowston and Scozzi, 2004).

Figure 1. Tasks involved in the defect fixing process of OSS projects

3. Research Model

Figure 2 shows the research model underlying this study. According to Figure 2, there are two hypotheses underlying this research. The first hypothesis is regarding the operating systems an OSS product can be run on, and the second hypothesis concerns with the interaction between a project’s license restrictiveness and project size, and its impact on the effectiveness of the defect-fixing process.

Figure 2. Research Model

3.1 Operating system

Unix and Linux are two of the most well-known OSS operating systems. The Unix operating system (developed in Bell labs) emerged in the 1970s. Moody (2002) believes that the most important milestone in OSS has been initiating the development of the Linux operating system in 1991 by Linus Torvalds, an undergraduate student at the University of Finland. Linus Torvalds released early versions of Linux under his own licensing terms, which implied any redistribution to be free of charge. Only recently have open source applications started to run on operating systems other than Linux and Unix such as Mac and Windows.

Consumers of open source software use various types of operating systems. Apart from those professional users who are able to work on every kind of operating system, other OSS consumers are
only familiar with one or few operating systems. Accordingly, producing software that can be run on a higher number of operating systems can help attract a larger community to use a given OSS product (Midha, 2007). Midha (2007) also confirms this, “… it would be logical to assume that the target operating systems will influence the number of potential subscribers…(Midha, 2007, p.92)” A larger user community in turn means more contributions to the project defect tracking system by the community users. Therefore, projects which are compatible with a higher number of operating systems are more likely to procure contributions to their defect tracking system from their user community. This leads us to postulate that the effectiveness of the defect-fixing process might be stronger for OSS projects that can be run on a broader range of operating systems. Subramaniam et al. (2009) found that a project’s choice of operating system impacts its success. We posit hypothesis 1 as below:

**H1.** The number of project’s operating systems has a positive impact on the effectiveness of defect-fixing process.

### 3.2 License

Software can be called open source only if it is released under a license approved by the Open Source Initiative (Open Source Initiative, 2005). Open source licenses can be classified into 2 categories (Stewart et al., 2006): “restrictive” e.g. GNU general public license (GPL) and “non-restrictive” e.g. Berkeley Software Distribution (BSD). GNU General Public License (GPL) which is the most popular OSS license provides that the source code should be available, and that any other software that use part of the source code from GPL licensed software must also make its source code fully available under the same license. In addition, GPL doesn’t permit the mixing of OSS and CSS source code.

One of the key variables impacting OSS projects’ outcomes is their license restrictiveness (Lerner and Tirole 2002; Scacchi 2004). The legal unsureness of restrictive licensing as well as the restrictions it involves for redistribution of the software can impact the number of users who are about to adopt a particular OSS. Stewart et al. (2006) also showed that projects with less restrictive licenses can attain
a higher amount of interest from the OSS community. A higher user interest means more contributions to the project defect tracking system by the developer users. This can in turn lead to a more effective defect-fixing. Thus, projects with more restrictive licenses are less likely to get contributions to their defect tracking system from the community and will be more reliant upon their team developers to operate the defect-fixing process. That is, the impact of team size on the effectiveness of the defect-fixing process might be stronger for OSS projects that have a more restrictive license. Therefore, we posit hypothesis 2 as below:

**H2.** The magnitude of team size impact on the effectiveness of the defect-fixing process is associated with the project’s license restrictiveness.

### 4. Research Methodology

The objective of this paper is the investigation of the operating system and the license restrictiveness impacts on defect-fixing effectiveness. This study is a positivist research in nature because it investigates the causal relationships between variables. To do so, we operationalized each variable under study, chose an OSS repository to sample projects, and collected the data on those measures from the sampled projects.

The environment chosen for the data collection of this research is the largest OSS repository, Sourceforge.net. As of Feb 2009, Sourceforge had 230,000 registered open source software projects, and it also has more than 2 million registered members (Source: www.sourceforge.net). Sourceforge offers a tracking system to OSS projects that are hosted on it to handle their defect-fixing activities. We collected data from 1481 OSS projects hosted on Sourceforge. The data required to examine the hypotheses underlying this research was collected from the publicly available information on Sourceforge.

The sample was chosen from the projects in 5 categories of: Game, Scientific, Software development, Communication, and Security that had at least 5 defect reports in their defect tracking system, have
released within last 2 years, their development status was either Beta or Stable. Table 1 shows distribution of the projects in terms of number of downloads.

**Table 1. Distribution of the projects in terms of number of downloads**

Team size will be measured by the number of people registered on the project profile as team members. A categorical variable will be used for license restrictiveness: 0 = restrictive license (GNU GPL), 1= non-restrictive licenses (anything except GNU GPL). As for operating system, the number of operating systems that an OSS project can be run on will be used. For the effectiveness of the defect-fixing process, we use the four items proposed by Ghapanchi and Aurum (2011) including: the number of defects submitted, the number of defects assigned, the number of closed defects, the number of fixed defects; except that we use the number of defects submitted by team members instead of total number of defects because it was observed that there are plenty of duplicate and invalid defects reported by users, however a project’s team members are less likely to report such defects. Table 2 shows the indicators used to operationalize the variables under study.

**Table 2. Indicators used to measure the variables**

5. **Findings**

The two hypotheses underlying the research model of this study were examined with the data gathered from 1481 OSS projects. The results showed that both hypotheses were supported, implying the importance of operating system and license restrictiveness in predicting defect-fixing effectiveness. The next two sub-sections explain the findings of this study as per each hypothesis.

5.1 **Direct effect**

The data analysis for this research was carried out through partial least squares (PLS). The PLS method has been employed by researchers in recent years because of its ability to model latent
variables under conditions of non-normality (Chin, 1998). PLS allows the researchers to examine the relationships among the conceptual variables. It also allows the researcher to analyse how well the measures relate to the associated variable.

Our research model accounts for 24.8 percent of the variance in the effectiveness of defect-fixing process ($R^2=0.248$). As indicated in Table 3, hypothesis 1 was supported. As predicted in Hypothesis 1, the number of a project’s operating systems was positively and significantly (coefficient = 0.182, $t$-value = 4.72) related to defect-fixing effectiveness ($p < 0.001$). This indicates that the higher the number of operating systems compatible with an OSS, the more effective the project’s defect-fixing process.

Table 3. Path coefficients for Baseline model

5.2 Moderating effect

A moderated multiple regression analysis (MMR) was then applied to test the moderating effect (Hsu et al., 2008). A three-step approach was adopted to test the moderating effect using MMR with Smart-PLS. First, team size was regressed with defect-fixing effectiveness (See model 0 in table 4). Second, the moderator was included into the model (See model 1 in table 4). Third, the interaction term between team size and the moderator was added into the model (See model 2 in table 4). Then Cohen’s F-square was calculated to examine the significance of the moderating effect. The result of the test is shown in Table 4.

Table 4. Moderating effect of license restrictiveness

The path coefficient for the moderating effect of license restrictiveness was -0.14. The difference in $R^2$ is used to determine the overall effect size $f^2$ for the interaction. In this research Cohen’s $F^2$ of 0.024 showed a small to medium moderating effect for license restrictiveness (Cohen, 1988). Therefore, the magnitude of the team size impact on the effectiveness of the defect-fixing process is
associated with the project’s license restrictiveness (Hypothesis 2) was supported. This means that team size has a relatively larger impact on defect-fixing effectiveness for projects with a more restrictive license.

6. Discussions

In this research the inter-relationship between team size, project characteristics (i.e. project’s license and operating systems), and defect-fixing effectiveness was examined. The findings of this study demonstrated that the link between team size and the effectiveness of the defect-fixing process is rather complex. Adding project’s license as a moderator is a first step towards better comprehending of this relationship. Although project license has been studied as an independent variable impacting positive outcomes in current OSS papers (e.g. Subramaniam et al., 2009), we are one of the firsts to study license as a moderator. Future studies should continue to investigate the nature of this relationship.

Our study has several implications for OSS project managers as well as organizations interested in adopting OSS software. Firstly, this study looked into license restrictiveness and operating system in part because of their practical relevance. Licensing and operating systems are under the control of project administrators that start and run the project. We found that the impact of team size on defect-fixing effectiveness depends on the project’s license restrictiveness. This means that for the projects with a more restrictive license having a large project team is more important for achieving effective defect-fixing than for those projects with a less restrictive license. The implication for practice is that in order for OSS project managers to reduce their project’s dependence on team size in operating their defect-fixing process, they should choose a less restrictive license for their project. This way, by attracting more volunteer input from the user community, the project can attain a more effective defect-fixing process. It was also found that the higher the number operating systems for a project, the more effective the defect-fixing process. This means that by making an OSS run on a broader range of operating systems, project administrators can achieve higher defect-fixing effectiveness.
7. Conclusions and Limitations

This study examined the inter-relationships between team size, operating system, license restrictiveness and the effectiveness of the defect-fixing process in an OSS environment. The data collected on 1481 OSS projects confirmed that having a larger team size and being compatible with a broader range of operating systems has a positive impact on the effectiveness of the defect-fixing process. We also found that the magnitude of the impact of team size on defect-fixing effectiveness depends on the project’s license restrictiveness, meaning that in order for OSS projects to decrease their reliance on team size to manage their defect-fixing process, they need to choose a less restrictive license. Given the importance of the defect-fixing process in impacting users’ perception of quality and value, we call future research to examine other potential antecedents to defect-fixing effectiveness.

Our research model accounts for 24.8 percent of the variance in the effectiveness of the defect-fixing process ($R^2=0.248$), which means that the model explains a small proportion of variance in the dependent variable. That could be a limitation for our research that we need to acknowledge. However, we note that prior research on OSS have also faced this issue (Crowston and Scozzi, 2002).

Another potential limitation of this research could be the generalizability of the findings. Firstly, five project categories were sampled for this study, potentially limiting the generalizability of the findings across all project categories. Secondly, the sample for this study was selected from projects registered on Sourceforge; since these projects might differ from OSS projects that have been registered on other OSS portals such as Freshmeat, this could also limit the generalizability of the finding.

Acknowledgements

The authors would like to thank the anonymous reviewers for their constructive and helpful comments.
References


Figures:

Figure 1. Tasks involved in the defect fixing process of OSS projects

Figure 2. Research Model
Tables:

Table 1. Distribution of the projects in terms of number of downloads

<table>
<thead>
<tr>
<th>Number of downloads</th>
<th>Frequency</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>50-1000</td>
<td>78</td>
<td>5%</td>
</tr>
<tr>
<td>1000-20,000</td>
<td>601</td>
<td>41%</td>
</tr>
<tr>
<td>20,000-100,000</td>
<td>415</td>
<td>28%</td>
</tr>
<tr>
<td>&gt;100,000</td>
<td>387</td>
<td>26%</td>
</tr>
<tr>
<td>Total</td>
<td>1481</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 2. Indicators used to measure the variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Measure Label</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Dependent variable</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Effectiveness of defect-fixing</td>
<td>Sbmt</td>
<td>Total number of defects submitted by team</td>
</tr>
<tr>
<td>process</td>
<td>Asgn</td>
<td>Total number of defects assigned</td>
</tr>
<tr>
<td></td>
<td>Fix</td>
<td>Total number of defects fixed</td>
</tr>
<tr>
<td></td>
<td>Close</td>
<td>Total number of defects closed</td>
</tr>
<tr>
<td><strong>Independent variables</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Developer interest</td>
<td>DI</td>
<td>Number of developers registered on the project profile</td>
</tr>
<tr>
<td>Operating system</td>
<td>OS</td>
<td>Number of operating systems that the project is compatible with</td>
</tr>
<tr>
<td><strong>Moderator</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>project license</td>
<td>Licns</td>
<td>0 for restrictive license (GNU GPL), and 1 for non-restrictive licenses (others)</td>
</tr>
</tbody>
</table>
Table 3. Path coefficients for Baseline model

<table>
<thead>
<tr>
<th>Relationship</th>
<th>Path coefficient</th>
<th>Sample Mean</th>
<th>Standard Deviation</th>
<th>Standard Error</th>
<th>T Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system -&gt; Effectiveness of defect-fixing</td>
<td>0.1822</td>
<td>0.1835</td>
<td>0.0386</td>
<td>0.0386</td>
<td>4.7203</td>
</tr>
</tbody>
</table>

Table 4. Moderating effect of license restrictiveness

<table>
<thead>
<tr>
<th>Independent variable</th>
<th>Direct effect</th>
<th>Moderating effect</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model 0</td>
<td>Model 1</td>
</tr>
<tr>
<td>Team Size (TS)</td>
<td>0.44</td>
<td>0.44</td>
</tr>
<tr>
<td>License Restrictiveness (LR)</td>
<td>-0.06</td>
<td>-0.06</td>
</tr>
<tr>
<td>TS*LR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R-square</td>
<td>R₀²=0.198</td>
<td>R₁²=0.202</td>
</tr>
</tbody>
</table>

Dependent variable: the effectiveness of the defect-fixing process; moderator: license restrictiveness. The f-value of R² difference is estimated by [(R₂² - R₁²)/(df2 - df1)]/[(1 - R₂²)/(n - df2 - 1)].