International Journal of Business, Human and Social Sciences
ISSN: 2517-9411
Vol:3, No:4, 2009

Defects in Open Source Software: The Role of Online Forums

[1]Faheem Ahmed, [1]Piers Campbell, [1]Ahmad Jaffar, [2]Luiz Capretz

[1]Faculty of Information Technology, United Arab Emirates University

P. O. Box 17551, Al Ain, United Arab Emirates

[2]Department of Electrical & Computer Engineering, University of Western Ontario, London, Ontario, Canada N6A 5B9

f.ahmed@uaeu.ac.ae, p.campbell@uaeu.ac.ae, ajaffar@uaeu.ac.ae, lcapretz@eng.uwo.ca

***Abstract*—**Free and open source software is gaining popularity at an unprecedented rate of growth. Organizations despite some concerns about the quality have been using them for various purposes. One of the biggest concerns about free and open source software is post release software defects and their fixing. Many believe that there is no appropriate support available to fix the bugs. On the contrary some believe that due to the active involvement of internet user in online forums, they become a major source of communicating the identification and fixing of defects in open source software. The research model of this empirical investigation establishes and studies the relationship between open source software defects and online public forums. The results of this empirical study provide evidence about the realities of software defects myths of open source software. We used a dataset consist of 616 open source software projects covering a broad range of categories to study the research model of this investigation. The results of this investigation show that online forums play a significant role identifying and fixing the defects in open source software.

***Keywords*—**About Open source software, software engineering, software defect management, empirical software engineering.

I. Introduction

In the recent past many large software development companies are putting their efforts in open source projects which gave momentum to this initiative. As an economically viable alternative, open source software (OSS) has proven to have reduced maintenance cost that benefits all instead of profit making vendors. Open source refers to the use of shared source code, open standards, and collaboration among software developers and users worldwide to build software, identify and correct errors, and make enhancements [1]. This ultimately reflects that the code is freely available and people can modifiable source code and further share the knowledge among individuals and group of people as well. This significantly important aspect allows that there is no market pressure and constraints on the timings to launch software. Large number of unbiased users would also allow them to review the code and find out defects at an earlier stage thus has higher potential to reduce maintenance cost. Likewise, immediate redistribution of improved or bug free codes can only reduce potential future complication implicating costly maintenance recovery, usually charged by closed source vendor. It benefits clients more than profit oriented vendors up to some extent. Particularly in consumer electronic industry, by using pre-made OSS components for low level routine, will release resources to focus on higher level components for competitive advantage [2]. It addresses competitive pressure by reducing development time as compared to normal software development lifecycle. Necessarily bounded by open source licensing agreement, this is in-line with not reinventing the wheel that influences in minimizing the development cost. The process of software maintenance in OSS is different from the traditional way of developing and managing software. In a traditional setting of software development it is generally require having a maintenance team present in the organization to provide post operational support.

Active open source projects usually have a well-defined community with common interests which is involved either in continuously evolving its related products or in using its results [3]. OSS is developed by loosely organized communities of participants located around the world and working over the Internet and remarkably, most participants contribute without being employed, paid, or recruited by the organization [4]. The use of the internet further accelerates the popularity and use of free and open source software at an unprecedented rate of growth. One can achieve a better software quality by allowing many people to work independently and exploring the possibilities of improvements. When many people are interacting and exchanging ideas there are more chances that new ideas to improve the software and finding and fixing defects are possible. The OSS community is renowned as a close interaction of professional and amateur software developers

and the development character of OSS ensures that reuse is a central pillar in project development.

a. Research Motivations & Related Work

The life cycle of software development depicts that when the software is delivered to the users, any defects identified are fixed under the umbrella of maintenance activities. The maintenance of software has broader vision and covers many aspects such as perfective, corrective and enhancement. One of the major concerns in OSS is the early delivery to the user and not consistently following the life cycle of software development process. Vixie [5] finds that in OSS the software life cycle activities such as requirement definition, system level and detailed design, unit and system testing, and support are not carried out in a manner similar to traditional software engineering. This also raises issues concerning defect identification and fixing.

Empirical studies regarding open source quality assurance activities and quality claims are rare [6]. Koponen [2] discuss defect management and version management system as an integral part of OSS maintenance process. Aberdour [7] observes that the open source software model has led to the creation of significant pieces of software, and many of these applications show levels of quality comparable to closed source software development. Raymond suggests the high quality of OSS can be achieved due to high degree of peer review and user involvement in bug/defect detection. Generally a popular or active project means that the community in the OSS project are interacting constantly and providing feedback to activities such as defect identification, fixing of defects, new feature request and support requests for the further improvement. Wayner [8] finds that developers contribute from around the world, meet face-to-face infrequently if at all, and coordinate their activity primarily by means of computer-mediated communications. Crowston and Scozzi [9] investigate the coordination practices for software bug fixing in OSS development teams and observes that task sequences are mostly sequential and composed of few steps, namely submit, fix and close and effort is not equally distributed among process actors and as a result few contribute heavily to all tasks, while the majority just submit one or two bugs. Cubranic and Booth [10] discuss major issues of coordinating open source development projects, including collaborative communication mediums and configuration management tools. Mockus et al. [11] provides a comprehensive comparison of Apache against five commercial products in terms of developer participation, team size, productivity and defect density, and problem resolution

II. Research Model and Hypothesis of the Study

The world has witnessed a rapid growth of OSS development projects after the increased popularity and use of internet. This has formed a diverse community of software developers all across the globe. They share directly or indirectly knowledge and communicates using the online forums. Their needs and interests are also diverse. An indirect measure of success and failure of an OSS can be considered as activities on the online forums. An online forum of OSS having continuous increase in messages shows the interest of the people and helps in identifying and fixing defects. The main objective of the research model of this study shown in Figure 1 is to analyze the association between software defects identification and fixing in OSS and public forums associated with the OSS project. The main objective of this study is to investigate the answer to the following research question:

RQ: Does online public forum helps in indentifying and fixing OSS defects?

In order to empirically investigate the research question we hypothesize the following:

H0: Public forums help in identifying and fixing OSS defects.

H1a: The Open bugs present in OSS are positively related with mailing list in online forums.

H1b: The open bugs in OSS are positively related with the number of messages in online forums.

H2a: The close bugs present in OSS are positively related with mailing list in online forums.

H2b: The close bugs in OSS are positively related with the number of messages in online forums.

It is important to mention here that we are using the term “open bug” as a defect which is indentified but has not been fixed yet, whereas “close bug” refers to a bug which was reported and fixed.

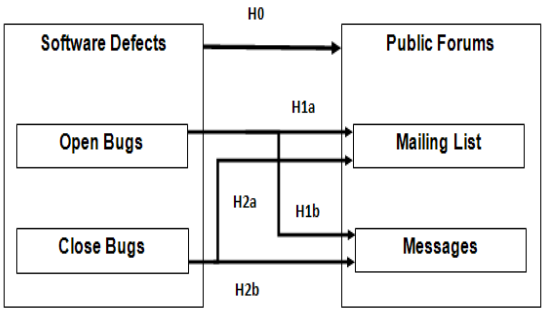

Fig. 1 Research model of the study

a. Data Collection and Experimental Setup

We collected the data of 1880 open source software projects from www.sourceforge.net, a popular data repository of open source software projects on the internet. The dataset covers various categories of open source software projects

such as communication, database, desktop, education, format & protocols, games & entertainment, scientific & engineering, security, software development, system and text editor. The first filtration activity removes the data of all those projects which has either total bugs of 0 or having no online forums. This reduces the dataset to 650 projects. Later on outliers on the basis of total bugs and number of online forum were removed and this reduces the dataset to 616 open source projects. Figure 2 illustrates the number of total bugs present in various open source software project of the initial dataset of this study. It also highlights the outliers which are removed and updates distribution of total bugs is shown in Figure 3. Figure 4 & 5 illustrates the distribution of total online forums in various open source software projects before and after removing the outliers. In the dataset of this study we used communication (183), database (76), desktop (48), education (21), format & protocols (15), games & entertainment (60), scientific & engineering (41), security (47), software development (28), system (53) and text editor (44) projects. Figure 6 illustrates the distribution of the dataset in various categories.

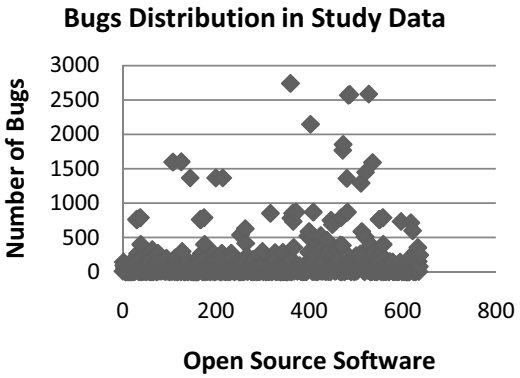

Fig. 2 Bugs distribution

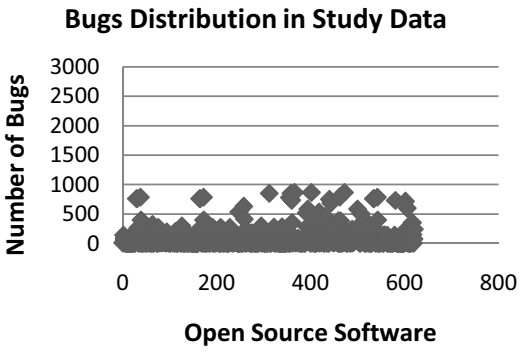

Fig. 3 Bugs distribution after removing outliers

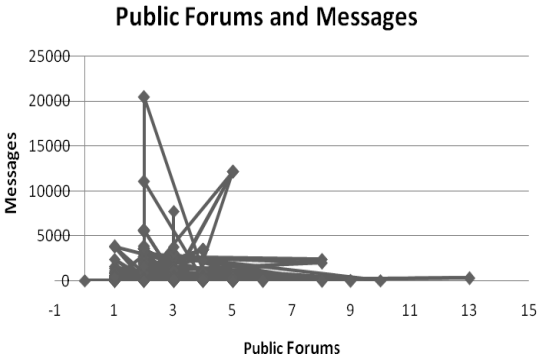

Fig. 4 Public forums data distribution

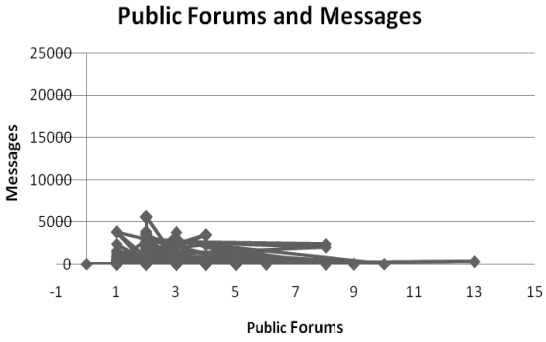

Fig.5 Public forums data distribution after removing outliers

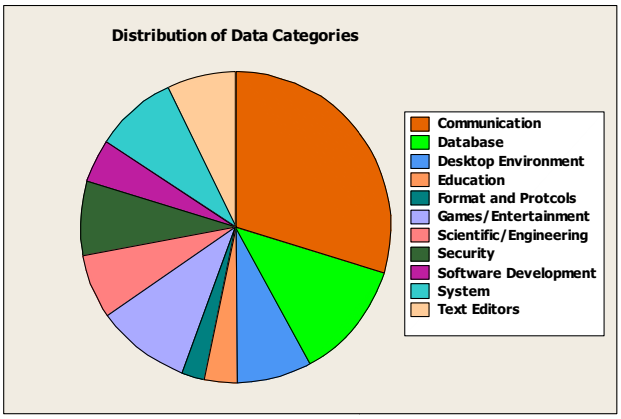

Fig. 6 Distribution of project categories in the dataset

The maximum open bugs were found in the category of format & protocols (215). Minimum number of (67) open bugs were observed in the category of system software. The maximum bugs which have been fixed were observed in the categories of database and desktop environment (857 each). The category of system software also shared the minimum number of (344) bugs which have been fixed. Database, desktop environment and format & protocol categories had observed maximum number (868 each) of total bugs. The

category of software development project has minimum number of (507) known total bugs. The category of "communication" has maximum number of online forums (13) in one project. The highest number of messages (5611) was found in the category of a communication project. The lowest number of (2938) messages was observed in the category of database project. The maximum number of mailing lists of (7) each was observed in the categories of communication, education, games & entertainment and scientific & engineering projects. Desktop environment, security and text editors shared the minimum number of (5) mailing lists in a project.

To analyze the research model and check the significance of hypotheses H1a, H1b, H2a and H2b, we used various statistical analysis techniques. Initially we divided the data analysis activity into three phases. Phase-I dealt with normal distribution tests and parametric statistics. Phase-II dealt with non-parametric statistics. In order to increase the external validity of the study, we used both statistical approaches of parametric and non-parametric methods. We tested for the normal distribution of all the six factors of total, open, close bugs as well as number of online forums, mailing list and messages using mean, standard deviation, kurtosis and skewness techniques, and found the values for all these tests to be within the acceptable range for the normal distribution with some exceptions. We conducted tests for hypotheses H1a, H1b, H2a and H2b using parametric statistics, such as the Pearson correlation coefficient and one tailed t–test in Phase-I. In Phase-II of non-parametric statistics, we conducted tests for hypotheses using the Spearman correlation coefficient. Phase-III dealt with testing the hypotheses of the research model of this study using the technique of Partial Least Square (PLS). The PLS technique helps when complexity, non-normal distribution, low theoretical information, and small sample size are issues. In the PLS testing of hypotheses we keep one factor as independent and other as dependent variable. We used the PLS technique to increase the reliability of the results. The statistical calculations were performed using Minitab® 14 software.

III. Data analysis & Results

We examined the Pearson correlation coefficient and t-test between variables involves in the hypotheses H1a, H1b, H2a and H2b. The Pearson correlation coefficient between open bugs and number of mailing lists in the public forums was positive (0.17) at $P < 0.001$, and thus provided a justification to accept the H1a hypothesis. The hypothesis H1b was accepted based on the Pearson correlation coefficient (0.29) at $P < 0.001$, between open bugs and number of messages on the online forum. The correlation coefficient of 0.22 at $P < 0.001$ was observed between the close bugs and number of mailing lists in the online forum. The positive correlation coefficient of 0.43 at $P < 0.001$ meant that H2b was accepted. Hence, it was observed and is reported here that hypotheses H1a, H1b, H2a, and H2b, were found statistically significant and were accepted.

In Phase-II we conducted non-parametric statistical technique using Spearman correlation coefficient to test the hypotheses H1a, H1b, H2a and H2b. Hypothesis H1a was statistically significant at $P < 0.01$ with Spearman correlation coefficient of 0.73. A positive association was observed between open bugs and number of messages (H1b) on the online forum (Spearman: 0.71 at $P < 0.01$). H2a, which deals with between the close bugs and number of mailing lists in the online forum, was accepted (Spearman: 0.58 at $P < 0.01$. The Spearman correlation of (0.80 at $P < 0.01$) was observed for H2b. Hence, it was observed and is reported here that hypotheses H1a, H1b, H2a, and H2b, were found statistically significant and were accepted.

In Phase-III of hypotheses testing, we used the PLS technique to overcome some of the associated limitations and to cross validate with the results observed using the approaches of Phase-I and Phase-II. We tested the hypothesized relationships, i.e. H1a, H1b, H2a and H2b, by examining their direction and significance. The hypothesis involves two variables therefore in PLS we placed one variable as the response variable and other as the predicate. Table-I reports the results of the structural tests of the hypotheses. It contains observed values of path coefficient, R^2 and F-ratio. The path coefficient open defect (H1a) was found to be 0.17, R^2: 0.03 and F-ratio (19.69) was significant at $P < 0.001$. Open defects (H1b) had positive path coefficient of 0.09 with $R^{2:}$ 0.08 and at $P < 0.001$ F-ratio was 57.71 with number of messages. Close defect with mailing list (H2a) (Path coefficient: 0.22, R^2: 0.05, F-ratio: 34.07 at $P < 0.001$) had the same direction as proposed. Close defect and number of messages (H2b) (Path coefficient: 0.43, R^2: 0.19, F-ratio: 146.46 at $P < 0.001$) also had the same direction as proposed in H2b. All in all, the hypotheses H1a, H1b, H2a and H2b, showed significant at $P < 0.001$ with a positive path coefficient and were in the same direction as proposed.

IV. Discussion of empirical evidence

It is clear from our analysis that there is a positive correlation between the volume of messages posted on online forums and the number of open bugs reported in a particular OSS project. This demonstrates that the OSS community is active in testing of projects and the identification of defects. It further highlights that defects are not simply accepted as an unavoidable feature of OSS but rather the community which is established around an OSS project work collaboratively to identify and correct defect in a given project. Our study also shows that in addition to the support network generated around an OSS project the volume of interested parties is significantly increased for projects with unsolved defects. This is demonstrated by the positive correlation between the number of open bugs (defects) and number of individuals who are registered in the mailing lists of a given project. This correlation suggests that the OSS community has a significant

support network which is likely to be larger than the support team of a proprietary application. As mailing list members are also altruistic in nature it is likely that a collaborative environment will lead to a number of possible solutions being identified. Further evidence of this collaborative support network which facilitates the fixing of defects in Open Source Software can be witnessed in our analysis in the correlation between the number of messages in the online forums and the level of fixed bugs. As can clearly be seen this positive correlation shows that the more active a thread on a particular OSS project is the greater the number of defects which have been closed. The same correlation can be observed when examining the volume of users in the mailing list and the number of closed bugs. Once again a highly active mailing list is positively correlated to the number of defects which have been rectified. The analysis clearly demonstrates that the interest and high level of involvement in OSS by volunteer developers leads to a high degree of available support which in turn leads to a rapid identification and subsequent rectification of defects in the projects.

Table 1: PLS Regression Analysis

	Open Defects	Close Defect
Mailing List	(H1a) Coefficient: 0.17 R^2: 0.03 F-ratio: 19.69 P-value: < 0.001	(H2a) Coefficient: 0.22 R^2: 0.05 F-ratio: 34.07 P-value: < 0.001
Messages	(H1b) Coefficient: 0.09 R^2: 0.08 F-ratio: 57.71 P-value: < 0.001	(H2b) Coefficient: 0.43 R^2: 0.19 F-ratio: 146.46 P-value: < 0.001

V. Conclusion

Free and open source software is gaining popularity at an unprecedented rate of growth. Organizations despite some concerns about the quality have been using them for various purposes. The objective of this study was to analyze empirically the association between managing software defects in OSS and online public forums associated with the OSS project. We observed that online forums are the corner stone of managing software defects in OSS. The management of new defect right from the identification, solution to fixing phases is communicated via online forum. This study further helps in understanding the significant role of online forums in OSS development. We are currently working on a prediction model to predict the software defects in an OSS project based on the active involvement of the online community attached with the projects.

References

[1] T. O'Reilly, Lessons from open-source software development, Communications of ACM, vol. 42, no. 4, pp. 32–37, 1999
[2] T. Koponen, Life-cycle of the defects in Open Source Software Projects, Proceedings of The 2nd International Conference on Open Source Systems, Italy, 2006.
[3] Gacek, C. and Arief, B., The many meanings of open source, IEEE Software, Vol. 21, No. 1, pp. 34-40, 2004.
[4] C horng-Guang W., James H. G., Clifford E. Y, An empirical analysis of open source software developers' motivations and continuance intentions, Information & Management, Vol. 44, No. 3,pp. 253-262, 2007.
[5] P. Vixie, Open Sources: Voices from the Open Source Revolution, O'Reilly & Associates, 1999, pp. 91-100.
[6] R. Glass Is open source software more reliable? An elusive answer. The Software Practitioner 11 (6), 2001.
[7] M. Aberdour. Achieving quality in open-source software IEEE Software, Vol. 24, No. 1, pp. 58–64, 2007.
[8] P. Wayner, Free For All, HarperCollins, New York (2000).
[9] K. Crowston & B. Scozzi Bug Fixing Practices within Free/Libre Open Source Software Development Teams, , Journal of Database Management, Volume 19, Issue 2, 2008.
[10] D. Cubranic, K. Booth, Coordinating open-source software development. In: Proceedings of IEEE 8th International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises. pp. 61–69, 1999
[11] A. Mockus, R. Fielding, J. Herbsleb, A case study of open source software development: the Apache server. In: The 22nd International Conference on Software Engineering. pp. 263–272, 2000.