Effects of the Number of Developers on Code Quality in Open Source Software: A Case Study

Brandon Norick, Justin Krohn, Eben Howard, Ben Welna, Clemente Izurieta
Department of Computer Science
Montana State University
Bozeman, MT 59717
01-406-994-4780
{brandon.norick, justin.krohn, eben.howard, ben.welna, clemente.izurieta}@montana.edu

ABSTRACT

Eleven open source software projects were analyzed to determine if the number of committing developers impacts code quality. We use cyclomatic complexity, lines of code per function, comment density, and maximum nesting as surrogate measures of code quality. We find no significant evidence to suggest that the number of committing developers affects the quality of software.

ACM Classification: D.2.8 [**Software Engineering**]: Metrics – *software evolution, product metrics, code quality.*

General terms: Measurement, Design, Experimentation.

Keywords: Code quality, open source software, number of developers.

1. INTRODUCTION

Previous findings indicate that open source code suffers from poor code quality [1]. The motivation for this exploratory case study is to investigate if quality is affected by the number of committing developers involved in the project. We chose eleven mature open source projects; Filezilla, WinSCP, Miktex, UltraVNC, Notepad++, Audacity, Exult, SMPlayer, TCL, Wireshark, and TortoiseSVN. Selected projects were deemed mature by observing the number of downloads.

Contrary to prior results, we find evidence to indicate that all eleven open source projects show characteristics of good code quality with the exception of TCL, which displays trends that are out of character when compared to the other projects in the case study. Further, we find that there appears to be no significant evidence to indicate a relationship between the number of developers involved in a project and observed code quality metrics.

2. BACKGROUND

The metrics gathered by Stamelos [1], served as a baseline and reference for comparing our case study results against an established and successful domain.

3. METRICS

Various metrics were used as surrogates of code quality. Cyclomatic complexity indicates good code quality if procedural code scores under 10 in complexity. Object oriented software has a lower score due to modularized complexity being shared across multiple methods [1], [2]. Research suggests that approximately fifty LOC per function is considered ideal [2]. The percentage of comments in code should be about 30% [1], [3]. Finally, maximum nesting (measured over the total lines of source code) is an indicator of the maximum complexity of that code [1].

All selected projects were written in C/C++ with some projects containing additional code in other languages. Only the C/C++ sections of the code were analyzed. No external libraries were analyzed.

4. RESULTS AND ANALYSIS

Results suggest that the number of developers is not an indicator of code quality. Ten out of eleven projects showed signs of good code quality regardless of the number of developers involved.

5. CONCLUSION

After mining eleven open source projects we have no significant evidence to suggest that there is a link between code quality and the number of committing developers. Contrary to past studies that suggest that too many or too few developers will decrease the quality of software, we find no such evidence.

REFERENCES

1. Stamelos, I., Angelis, L., Oikonomou, A., and Bleris, G. L. "Code Quality Analysis in Open Source Software Development," *Info Systems Journal*, vol 12, pp. 43-61, 2002.
2. Prasad, L. and Nagar, A. "Experimental Analysis of Different Metrics (Object-oriented and Structual) of Software," *2009 First International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 235-240, 2009.
3. Simmons, M. M., Vercellone-Smith, P., and Laplante, P.A. "Understanding Open Source Software through Software Archaeology: The Case of Nethack," *30th Annual IEEE/NASA Software Engineering Workshop SEW-30.*pp. 47-58, 2006.

ESEM'10, September 16–17, 2010, Bolzano-Bozen, Italy.