

Paper Three Summary

Matthew Neal, Joseph Sankar, and Alexander Sobran

September 23, 2015

Reference

Bird et al [2] listed below.

Important Keywords

Global development Development that is globally distributed but takes place within one company. The authors contrast this to *outsourcing*, in which development takes place among multiple companies and *collocated development*, where development takes place in one location.

Coordination breakdown A difficulty usually encountered in distributed development where employees in one location are not appropriately aware or responsive to the work of employees in another location.

Binary distribution Different levels of classifications for where binaries are developed. The authors give *building*, *cafeteria*, *campus*, *locality*, *continent*, and *world* as different ways to classify binary distribution.

Organizational integration An organizational structure that spans geographical locations at low levels.

Feature Extraction

Motivational statements The motivation of this study was to compare the post release failures of Windows Vista components developed in a distributed fashion to those developed by collocated teams as to attain insight into the affect of distributed team development against collocated team development. The researchers main goal is to confirm or refute the notion that distributed development on a global scale leads to a greater number of software failures.

Additionally, they aim to study whether the projects assigned to distributed teams were less complex. The idea there is that post release failures are expected in distributed teams to some degree, so are managers

making attempts to prevent those failures by assigning simpler projects to those teams.

Hypotheses The authors list two hypotheses [2]:

H1: Binaries that are developed by teams of engineers that are distributed will have more post-release failures than those developed by colocated engineers.

H2: Binaries that are distributed will be less complex, experience less code churn, and have fewer dependencies than colocated binaries.

Future Work The authors note that there haven't been any causal relationships established between development practices and effective distributed development. They suggest studies to examine the effects of various development practices on distributed development.

Informative Visualizations The visualizations on page 523 of the paper are particularly useful. Figure 2 shows a breakdown of where people working on a specific dll are working geographically to give an idea of the level of distribution Microsoft was dealing with. The other visualization on the same page, a Venn diagram showing concentric circles of the percentages of binaries at each distribution level, is very useful in understanding how distributed the project was overall.

Possible Improvements

This paper was incredibly well written and makes its argument in a fairly air tight manner. I think as it is, it would be difficult to improve upon it. That said, perhaps some data on outsourced projects within Microsoft for comparison or distributed projects within companies recently acquired by Microsoft would make that piece of their argument more solid, but could quickly turn into a book rather than an article. The project detailed was so specific and the organizations so specific that it would be difficult generalize to any other company. For example, the authors mention the fact that Microsoft sent a number of developers that had worked at Microsoft for over ten years, but were from the same cultural group to work with a distributed team in India. Not many companies would have either the staff or resources to do the same.

Connection to Other Papers

Our first paper [3] cited geographical issues as a difficulty in interpreting aggregated results. They ask how the work should be divided among multiple locations to maximize quality and productivity, initially suggesting that development should not be split over different locations. They note that this study

found different results, that geographical distribution had little to no effect on software quality.

The authors also mentioned that this study examined software on the binary level (an aggregated level), which supports their claim that aggregation is very common in defect modeling research. This paper was not mentioned in paper two [1].

References

- [1] Erik Arisholm, Lionel C. Briand, and Eivind B. Johannessen. A systematic and comprehensive investigation of methods to build and evaluate fault prediction models. *J. Syst. Softw.*, 83(1):2–17, January 2010.
- [2] Christian Bird, Nachiappan Nagappan, Premkumar Devanbu, Harald Gall, and Brendan Murphy. Does distributed development affect software quality?: An empirical case study of windows vista. *Commun. ACM*, 52(8):85–93, August 2009.
- [3] D. Posnett, V. Filkov, and P. Devanbu. Ecological inference in empirical software engineering. In *Automated Software Engineering (ASE), 2011 26th IEEE/ACM International Conference on*, pages 362–371, Nov 2011.