IN SEARCH OF SOFTWARE MAINTENANCE PRODUCTIVITY AND QUALITY: DOES SOFTWARE COMPLEXITY MATTER?

Rajiv D. Banker
Sandra A. Slaughter
Carlson School of Management
University of Minnesota

ABSTRACT

Over the past several decades, software maintenance has been absorbing a large and rising proportion of Information Systems (IS) resources with expenditures often as high as 80% to 95% of the total IS budget (Nosek and Palvia 1990). On a life-cycle basis, about three-fourths of the investment in software occurs *after* the system has been implemented. Thus, support for existing software represents a significant investment of resources for most firms and there is considerable interest in understanding and improving productivity and quality in the software maintenance task.

Software complexity is believed to be a major factor influencing performance in software maintenance, impacting maintenance costs (Banker, et al. 1993) and software error rates (Shen, et al. 1985). A principal objective of our research program, therefore, is to analyze the implications of software complexity for the support of an extensive application portfolio. Since software complexity is recognized as a psychological phenomenon, we draw upon Wood's (1986) psychological framework for task complexity to conceptualize three dimensions of software complexity: component, coordinative and dynamic complexity. We test our model of software complexity in a COBOL environment at a major national mass merchandising retailer. The study of software maintenance in this kind of environment is important. Although more than 60% of business expenditures for computing are devoted to maintenance of software written in COBOL, the majority of prior research in software maintenance focusses on non-COBOL programming languages (Hale and Haworth 1988).

Fieldwork is complete, but data analysis and conclusions are still at an early stage. Data collection has been accomplished in two phases. In the first phase, we collected productivity and software measures for a number of software maintenance projects that were completed over a two year timeframe. In the second phase of data collection, we obtained thirty months of software error measures for a 5,000 module application portfolio, which we are analyzing to investigate the relationship between software complexity and software reliability. Preliminary results suggest that, controlling for project team skill and experience, there is a strong effect for software complexity on software maintenance effort and software reliability. We also find that, contrary to the initial expectations of IS managers at the research site, the use of a software *development* productivity tool for generating code does not translate into a reduction in *maintenance* project effort. These results have interesting implications for software maintenance practice and research.

REFERENCES

Banker, R. D.; Datar, S.; Kemerer, C. F.; and Zweig, D. "Software Complexity and Software Maintenance Costs." *Communications of the ACM*, Volume 36, Number 11, November 1993, pp. 81-94.

Hale, D. P., and Haworth, D. A. "Software Maintenance: A Profile of Past Empirical Research." *Proceedings of the Fourth Conference on Software Maintenance*, 1988, pp. 236-240.

Nosek, J. T., and Palvia, P. "Software Maintenance Management: Changes in the Last Decade." *Journal of Software Maintenance*, Volume 2, Number 3, September 1990, pp. 157-174.

Shen, V. Y.; Yu, T. J.; Thebaut, S. M.; and Paulsen, L. R. "Identifying Error-Prone Software: An Empirical Study." *IEEE Transactions on Software Engineering*, Volume SE-11, Number 4, April 1985, pp. 317-323.

Wood, R. E. "Task Complexity: Definition of a Construct." *Organizational Behavior and Human Decision Processes*, Volume 31, 1986, pp. 60-82.