December 1997

A Longitudinal Analysis of Software Maintenance Patterns

Chris Kemerer
University of Pittsburgh

Sandra Slaughter
Carnegie Mellon University

A LONGITUDINAL ANALYSIS OF SOFTWARE MAINTENANCE PATTERNS

Chris F. Kemerer
Katz Graduate School of Business
University of Pittsburgh

Sandra A. Slaughter
Graduate School of Industrial Administration
Carnegie Mellon University

Software maintenance is a process that extends over years and often decades. However, most empirical studies of software maintenance are cross-sectional in nature (Kemerer 1995). As a result, very little is known about how software systems evolve and what determines different maintenance trajectories. Without such knowledge, it is difficult for information systems (IS) managers to plan effectively for maintenance work and to assess the need for reengineering or replacement of systems.

The intent of this study is to provide detailed insights into the dynamics and drivers of software evolution. The research is conducted in three phases. In the first phase, the patterns of software evolution are investigated and an assessment made of whether they are consistent with the laws of evolution dynamics proposed by Belady and Lehman (1976) and extended by Yuen (1985, 1987, 1988). Phase two involves identifying the specific drivers of software maintenance patterns, including module characteristics (size, functionality, and complexity), development practices (use of CASE tools and varied methodologies), technology evolution, and business changes (such as seasonal and annual events). The final phase has the objective of modeling and predicting the point of software system entropy.

A detailed and extensive data set has been collected, including an estimated 25,000 maintenance events to 3,800 software modules in a number of commercial systems over a period of 20 years. To classify each event, a coding scheme has been specified that categorizes maintenance events into their three basic types: corrections, adaptations, and enhancements (Swanson 1976). The scheme further refines these three basic maintenance types into 30 subcategories based upon a number of classification procedures for analyzing maintenance activities (Briand and Basili 1992; Rombach and Ulery 1992). Change events using the scheme have been coded for several systems and the change events for the remaining systems are in the process of being coded. The data will be analyzed using time series analysis, multilevel models, and restricted regression.

Preliminary analyses of coded systems already suggest a number of interesting patterns. The findings are that only a very small number of modules in each system are repaired and enhanced over the lifetime of the systems. Data on function, size, complexity, development practices, technology change, and business characteristics are being examined to determine the correlates of these evolution patterns. In plotting cumulative changes to systems over time, a major inflection point is observed for each system. For some systems, the rate of change increases after the inflection point (suggesting that the point of entropy has occurred). For other systems, the rate of change decreases after the inflection point (indicating stabilization). The inflection point is being modeled and an investigation made into why some systems stabilize and others entropy.

The results will provide detailed insights into how software evolves over time and what drives maintenance patterns. From a managerial perspective, information on predictable maintenance histories can be used by software managers

to more effectively plan for software maintenance. Knowledge of maintenance patterns can be used to improve change request management, to plan workloads for software maintainers, and to make decisions regarding software reengineering and replacement.

REFERENCES

Belady, L., and Lehman, M. "A Model of Large Program Development," *IBM Systems Journal* 15(1), 1976, pp. 225-252.

Briand, L., and Basili, V. "A Classification Procedure for the Effective Management of Changes During the Maintenance Process," *Proceedings of the IEEE Conference on Software Maintenance*, 1992, pp. 328-336.

Kemerer, C. "Software Complexity and Software Maintenance: A Survey of Empirical Research," *Annals of Software Engineering* 1(1), 1995, pp. 1-22.

Rombach, H., and Ulery, B. "Toward Full Life Cycle Control: Adding Maintenance Measurement to the SEL," *Journal of Systems and Software* (8), 18, 1992, pp. 125-138.

Swanson, E. B. "The Dimensions of Software Maintenance," *Proceedings of the IEEE International Conference on Software Engineering*, 1976, pp. 492-297.

Yuen, C. K. "An Empirical Approach to the Study of Errors in Large Software Under Maintenance," *Proceedings of the IEEE Conference on Software Maintenance*, 1985, pp. 96-105.

Yuen, C. K. "A Statistical Rationale for Evolution Dynamics Concepts," *Proceedings of the IEEE Conference on Software Maintenance*, 1987, pp. 156-164.

Yuen, C. K "On Analyzing Maintenance Process Data at the Global and Detailed Levels: A Case Study," *Proceedings of the IEEE Conference on Software Maintenance*, 1988, pp. 248-255.