12-31-1995

Distributed Computer System Complexity Versus Component Simplicity: Their Effects on Software Maintenance

Scott Schneberger
Georgia State University

DISTRIBUTED COMPUTER SYSTEM COMPLEXITY VERSUS COMPONENT SIMPLICITY: THEIR EFFECTS ON SOFTWARE MAINTENANCE

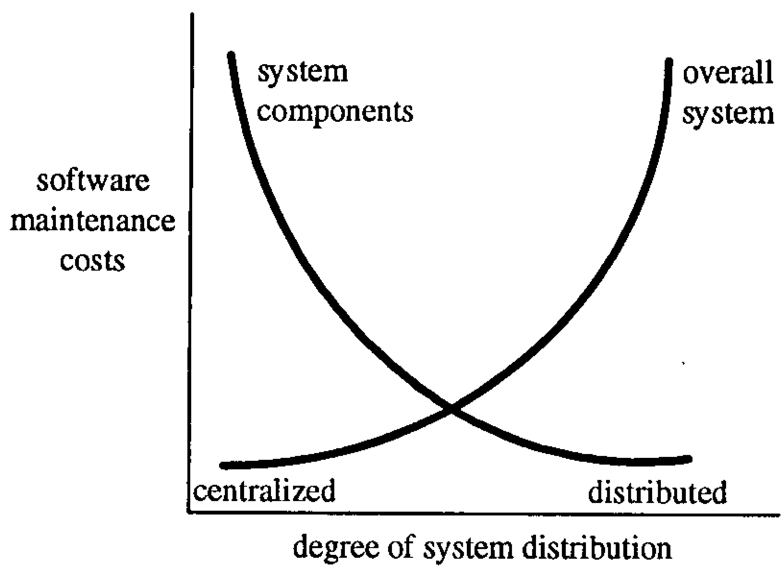
Scott L. Schneberger
Georgia State University

Abstract

The computing world is undergoing a transformation from centralized to distributed computer systems. At the same time, software maintenance has been identified repeatedly as the largest single system life cycle cost. This paper focuses on the issue of whether and to what extent distributed computer operating environments directly affect software maintenance difficulty (and, by association, software development difficulty).

The issue appears to depend on two diametrics of information system architectures: *component simplicity* and *system complexity*. The smaller (and more numerous) the system components, the simpler they are to deal with individually (lower software maintenance costs) but the harder it is to deal with the overall system (higher software maintenance costs). This research seeks empirical quantitative and qualitative data from IS system and application software analysts, designers, programmers, testers, and customer service representatives to determine whether the complexity of a distributed computer system has a greater effect on software maintenance than component simplicity and what the explanatory factors are.

Preliminary findings indicate that the complexity of studied distributed systems overwhelms the simplicity of their components — increasing the overall cost of software maintenance (as shown in the figure). In terms of factors, the *rate of technological change* and *variety of components* (particularly processors and system software) far outweigh the effects of component *numbers* or the *number of transactions*. Moreover, distributed systems appear to be much more than just "broken-up" centralized systems, as autonomous processors make computing event-driven and not just logic-driven.



AN EMPIRICAL ANALYSIS OF INFORMATION SYSTEMS CHANGES

Jonathan Heales
The University of Queensland

Abstract

The return on investment for Information Systems Development projects takes place during the period of stability following the system's implementation. This return on investment can be leveraged by extending the period of stability or reducing the ongoing costs of maintenance and enhancement. Since the usefulness of information systems is dependent on their alignment with the organization's goals and objectives, the ability of a system to absorb change directly affects its life span. Given also the risk of failure associated with systems development projects (Gladden 1982; Turner 1982; Hammer 1990; Lyytinen 1990), risk averse organizations are keen to delay such risky expenditure by extending an information system's life span.

This research develops a theoretically based model of information systems change, after Olle et al. (1991; see also Lederer and Prasad 1992; Marche 1993) and is based on the premise that stable information systems can react to changes in business requirements. Three types of adaptive maintenance activities are identified (Swanson and Beath 1989) which affect the surface structure, the physical structure or the deep structure of the information system (after Bunge1979; Chomsky 1965; Weber 1987; Wand and Weber 1990). Theory is used to justify changes in the activity mix during the maintenance phase. A bathtub shaped curve is derived which predicts that more effort will be spent on deep structure changes at the beginning and toward the end of a system's useful life. These are the periods when an information system is generally the most unstable.

After implementation, an information system is generally unstable due to the need to address business requirements overlooked during system design or those that have subsequently arisen during the construction phase. Toward the end of a system's life it can also become unstable due to the inability of the system to respond to business requirements efficiently and effectively. A system will be replaced if it is unable to restore its stability. Monitoring a system's stability during the maintenance phase can provide useful input to decisions relating to the undertaking of significant enhancement projects or system replacement.

A way to measure system stability is developed which is then subjected to empirical testing. A research method, incorporating a time series analysis, is designed to determine whether there is support for the shape of the bathtub curve. Some 4,000 user change requests have been collected and are currently being analyzed. Results from the analysis of this data will be presented.

The effects of covariates, identified from work done by Dekleva (1992), Willcocks (1992), and Willcocks and Lester (1992), such as the stability of organizational management, organizational considerations, system class and complexity, methodology used, semantic relativism (e.g., RDBMS systems), and semiotic representation (e.g., CASE) on the shape and length of the curve are subjected to empirical testing. These are factors that affect both the level of system maintenance required during the maintenance phase as well as the life span of a system.

A cross sectional analysis of systems exhibiting different characteristics is currently being performed to determine whether the covariates affect the stability measure of a system. Logistic regression is used to test the effects of the covariates and the results from this cross sectional analysis will also be presented.

BIBLIOGRAPHY

Bunge, M. *Treatise on Basic Philosophy: Volume 4: Ontology II: A World of Systems.* Boston: Reidel, 1979.

Chomsky, N. *Aspects of the Theory of Syntax.* Boston: MIT Press, 1965.