IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA ITB9982
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

Chapter XVII

The Effect of Task and Tool Experience on Maintenance CASE Tool Usage

Mark T. Dishaw, University of Wisconsin, USA

Diane M. Strong, Worcester Polytechnic Institute, USA

ABSTRACT

Computer-Aided Software Engineering (CASE) tools have been advocated for improving maintainer productivity and the quality of maintained software. While there is evidence that such benefits can accrue to organizations adopting maintenance-oriented CASE tools, a key problem in achieving the desired benefits from CASE tools is low usage of these tools by programmers. The previously tested Maintenance Tool Utilization Model was a first step in investigating the factors that affect whether maintainers choose to use CASE tools during maintenance projects. We test the addition of experience with software maintenance tools and with the software maintenance task to the Maintenance Tool Utilization Model. The role of experience is important because managers can provide training to increase experience and they can ensure that project teams have some members experienced with the tools

This chapter appears in the book, Advanced Topics in Information Resources Management, Volume 3, edited by Mehdi Khosrow-Pour. Copyright © 2004, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

or with the task. Data for the test are collected from software maintainers working on their organization's normal maintenance project backlog. Tool experience is significant as both a main and interaction effect, but task experience adds little to the explanatory power of the Maintenance Tool Utilization Model. These results support the value of improved CASE tool training programs.

INTRODUCTION

Organizations are adopting Computer-Aided Software Engineering (CASE) tools to support maintenance tasks. Software maintenance is the process of changing existing, production software. Production software is changed to correct problems, to adapt the system to a changing hardware and software environment, and to improve the system by making it more efficient or by adding functionality for users (Swanson & Beath, 1989). Maintenance-oriented CASE tools provide software tool support for the maintenance process. Such tools have been identified as a key to achieving maintenance productivity gains (Schneidewind, 1987). In addition to improving productivity, the use of these tools may contribute significantly to improving the quality of the software being maintained (Kim & Westin, 1988).

If CASE tools are used, productivity and quality benefits are being achieved (see Iivari, 1996, for a brief review of this literature). The problem, however, is low utilization of these tools (Iivari, 1996; Kemerer, 1992). Since CASE tools that are not used will have no effect on maintainer productivity or software quality, positive or negative, some amount of utilization is required for benefits to accrue to the organization. While some argue that too much utilization does not provide additional benefits, few would argue that overuse of software maintenance tools is currently a problem. Low tool utilization is the practical problem organizations adopting CASE tools now face. Tool utilization is the outcome measure used in this study.

The software maintenance process consists of two major steps, understanding the existing production system and modification of this software (Pennington & Grabowski, 1990; Yau & Collofello, 1985). The software maintenance support tools of interest to us are generally intended to support program understanding. Such tools are of interest because maintenance programmers spend 50-90% of their time understanding the program (Shaft & Vessey, 1998). If tools could help in the understanding process, the potential for significant productivity enhancements is large. These tools assist the programmer/analyst in discovering the physical and logical designs of the program or system at hand. The discovery of impacts of a proposed change on "distant" programs, i.e., programs linked by common data elements, is an important part of this phase, and is assisted by these software tools.

Copyright © 2004, Idea Group Inc. Copying or distributing in print or electronic forms without written permission of Idea Group Inc. is prohibited.

17 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/effect-task-tool-experience-maintenance/4627

Related Content

Security and Trust of Online Auction Systems
Pouwan Lei, Chris Chatwinand Rupert Young (2005). *Encyclopedia of Information Science and Technology, First Edition (pp. 2450-2454).*
www.irma-international.org/chapter/security-trust-online-auction-systems/14632

Contingency Theory, Agent-Based Systems, and a Virtual Advisor
John R. Burrett, Lisa Burnelland John W. Priest (2005). *Encyclopedia of Information Science and Technology, First Edition (pp. 577-583).*
www.irma-international.org/chapter/contingency-theory-agent-based-systems/14301

Open Source Applications for Image Visualization and Processing in Neuroimaging Training
Juan A. Juanes, Pablo Ruisoto, Alberto Prats-Galinoand Andrés Framiñán (2014). *Journal of Information Technology Research (pp. 75-87).*
www.irma-international.org/article/open-source-applications-for-image-visualization-and-processing-in-neuroimaging-training/111299

A Practical Method to Distribute a Management Control System in an Organization
Alfonso Reyes (2007). *Information Resources Management Journal (pp. 122-137).*
www.irma-international.org/article/practical-method-distribute-management-control/1315

A Framework for Understanding Returns from E-Government
Shirish C. Srivastavaand Thompson S.H. Teo (2009). *Handbook of Research on Information Management and the Global Landscape (pp. 113-131).*
www.irma-international.org/chapter/framework-understanding-returns-government/20617