Chapter V

QuickLocus:
A Software Development Process Evaluation Method for Small-Sized Organizations

Sarah Kohan
Carlos Alberto Vanzolini Foundation, Brazil

Marcelo Schneck de Paula Pessôa
Escola Politécnica da Universidade de São Paulo, Brazil

Mauro de Mesquita Spinola
Escola Politécnica da Universidade de São Paulo, Brazil

ABSTRACT

Many small-sized organizations have a significant role in the software business. A good software development process is the best way to assure quality to software products. Audits and evaluations are commonly used to verify implementation and certify the development process. This chapter describes the development and application of QuickLocus, a special-purpose evaluation method of software process developed to be applied in small-sized organizations. QuickLocus has been successfully applied since January 2003 by more than 50 organizations in process improvement programs. The reason for QuickLocus success is its low-cost methodology and its capability to output reliable information for kicking off a software improvement process. QuickLocus will provide organizations ways to be more competitive—producing better products at lower costs faster—ready to compete in the international software market. This chapter is organized in three parts: foundation, method, and results. The first one is theoretical and presents how the method was developed. The second part details the method itself, and the third part presents the method application results.
INTRODUCTION

Computers and software have increasingly become a key component for business operations, and today they are part of critical mission management activities in many companies. For some industries, for instance, banking, the daily operations are heavily supported by complex information systems. Many other industries are also heavily dependent on information technology. Software is the cornerstone for what has become known as the information technology.

The traumatic experience IBM went through in the 1960s while developing the OS/360 Operating System is a case study in this regard. According to Watson and Petre (1990, p. 339), the concept for the System/360 was chartered in December 1961, and “hundreds of programmers had to deliver millions of machine language code. Nothing like that had ever been tried before and the engineers lived under constant deadline pressure.” The same authors still report the following:

As the delays in delivering the software got bigger, more programmers were added; by 1966 we had two thousand programmers and the software development costs were already higher than the costs for developing the hardware. We learned, in the school of hard knocks, one of the biggest secrets in software engineering: you don’t speed up a software development project by filling up office space with programmers. (p. 342)

Following this same line of thoughts, several authors write about the importance of software and the controls required during its development to assure quality and reliability as well as the ability to forecast the resources required for project execution. Literature is very abundant on making the point that the identification of software development process status (process evaluation) is important to the organization overall improvement processes, contributes to problem reduction during software creation and help develop better products. Sometimes the word assessment is used with similar meaning of evaluation.

Jones (2000) highlights the requirements for research that adds accuracy to software development. For Jones (2000), software process evaluation is a practice that contributes to the delivery of better products. Process evaluations can contribute to the reduction of problems during software development, helping organizations pinpoint improvement opportunities in the process that contribute to the development of better products. Software Engineering Institute (1997) states that an effective change program requires understanding the current situation and mentions two sayings:

The Chinese Saying: “If you do not know where you are heading to, any road will do.”
Humphrey’s Saying: “If you do not know where you are, no map will help you.”

The “Chinese Saying” forces us to think about the clarity of goals. The demand exists for improvements in software process, both in large- and small-sized organizations. While it is a fact that big organizations are reasonably well served by existing solutions, we do not see it happening for the small ones. The requirement of an evaluation feature, as a process improvement tool, for the great number of small-sized organizations suggests the development of a method for evaluating the software process delivering on two must-have prerequisites. First, the evaluation results are the foundation for a process adjustment or improvement plan. Second, the costs associated to applying the evaluation are in line with the means and resources available for such organizations.

This chapter describes QuickLocus (Kohan, 2003), an evaluation method for software processes aimed at small-sized organizations. Evaluation is one of the most important activities for implementation of software quality processes improvement programs. Process models, techniques, and tools are easily found in the literature and in the mar-
Related Content

Balanced Maturity Framework
www.igi-global.com/chapter/balanced-maturity-framework/131289?camid=4v1a

Australian SMEs and E-Commerce Adoption: Newer Perspectives
www.igi-global.com/chapter/australian-smes-commerce-adoption/42275?camid=4v1a

Market Research 2.0: An Inclusive Approach to Understanding Customers’ Needs
www.igi-global.com/chapter/market-research-inclusive-approach-understanding/54175?camid=4v1a

Semantic Mapping for Access Control Model
www.igi-global.com/chapter/semantic-mapping-access-control-model/54179?camid=4v1a