Chapter 67
Benefits of CMM and CMMI-Based Software Process Improvement

Maged Abdullah
University of Malaya, Malaysia

Lee Sai Peck
University of Malaya, Malaysia

Rodina Ahmad
University of Malaya, Malaysia

Zarinah Mohd Kasirun
University of Malaya, Malaysia

Fahad Alshammari
University of Malaya, Malaysia

ABSTRACT

Software Process Improvement (SPI) has become the survival key of numerous software development organizations who want to deliver their products cheaper, faster, and better. A software process ultimately describes the way that organizations develop their software products and supporting services; meanwhile, SPI on the other hand, is the act of changing the software process and maintenance activities. This chapter purposefully describes the benefits of software process improvement. The Capability Maturity Model (CMM) and the Capability Maturity Model Integration (CMMI) are briefly surveyed and extensively discussed. Prior literature on the benefits and impacts of CMM and CMMI-based software process improvement is also highlighted.

INTRODUCTION

A new set of ideas on how to enhance the productivity and quality in software development organizations has emerged over the last decade under the term of Software Process Improvement (SPI) (Aaen, Arent, Mathiassen, & Ngwenyama, 2001). SPI has become the survival key of numerous software development organizations who want to deliver their products cheaper, faster, and better. A software process ultimately describes the way that organizations develop their software products and supporting services. Processes define what kind of steps the software development organizations should undertake at each phase of production and provide assistance in making good effort and schedule estimates, measuring quality, and developing plans (Gerry & Rory,

DOI: 10.4018/978-1-4666-4301-7.ch067

2007). Rico (2004) defines the software process improvement as "an approach to designing and defining a new and improved software process to achieve basic business goals and objectives." SPI is simply the act of changing the software process and maintenance activities. The aims are normally to decrease costs, increase efficiency, and also to increase profitability. For instance, SPI could be employed to create a new and enhanced process for software development organizations.

There is a widespread belief that a good software product is a result of mature and repeatable software processes, which have led to more focus on SPI to assist software development organizations realize its potential benefits. Thus, the search for new methodologies, ideas and innovations to enhance software development continues to be an essential focus for both academic and industrial research. In order to improve software development practices, many attempts have concentrated on defining, measuring, and monitoring development activities in an effort to identify and verify improvement areas. These attempts have led to the emergence of the term Process Model. A Process Model is defined as "a structured collection of practices that describe the characteristics of effective processes" (SEI, 2007). An organization can define a process improvement priorities and objectives and make its processes capable, stable, and mature by the help of a process model. Moreover, a process model provides a guideline for an organization to realize its current state; also to identify relevant improvement activities and to identify how to start these activities (SEI, 2007).

Effort spent in this area has resulted in several SPI models and standards such as Personal Software Process (PSP), Team Software Process (TSP) (Humphrey, 1995), ISO 9001 (Paulk, 1995), Six Sigma (Pyzdek, 2003) and the Carnegie Mellon Software Engineering Institute's Capability Maturity Model for Software (SW-CMM) (Paulk, Weber, Curtis, & Chrissis, 1995) and its most recent version, the Capability Maturity Model Integration (CMMI) (Chrissis, Konrad, & Shrum, 2007). The motivation for selecting CMM and CMMI as the base of this chapter is that they are influential, long-standing, and often-studied standard to SPI (Staples & Niazi, 2008). Moreover, CMMI-based SPI has led to quantifiable enhancement in how processes of software engineering are performed (Bollinger & McGowan, 2009). According to Jones and Soule (2002), among the software process improvement frameworks, CMMI became a standard model with high rate of acceptance.

BENEFITS OF SOFTWARE PROCESS IMPROVEMENT (SPI)

SPI is significant because it is the primary means by which a new and enhanced software development process is created. This is done in order to achieve important economic benefits at the least costs. Research shows that well designed software development process has a positive impact on the economic performance of software projects. Performance is usually measured in terms of productivity as well as the efficiency of the cost (Rico, 2004). On the other hand, poorly designed software development processes have negative consequences on the economic performance of software projects because poor software development process results in high operations cost, ineffective use of available resources, and lost opportunities in the market. According to Rico (2004), poorly designed processes result in a lack of quality and reliability, and poor customer satisfaction. That is why "Software Process improvement has emerged as an important paradigm for managing software development" (Ravichandran & Rai, 2003).

There are several benefits of that can be gained from the adoption of one or more SPI models or standards. As seen in Figure 1, Gibson, Goldenson, and Kost, (2006) described the effect of process improvement. It is used in this chapter for CMM and CMMI-based software process improvement; anyhow, the same depiction can