

Chapter 107

Software Process Lines: A Step towards Software Industrialization

Mahmood Niazi

Keele University, UK & King Fahd University of Petroleum and Minerals, Saudi Arabia

Sami Zahran

Process Improvement Consultant, UK

ABSTRACT

The industrial revolution transformed the cottage industry into mass production. In this chapter, the authors trace the recent advancement of the software industry and establish that it is following a similar route followed by the manufacturing industry towards industrialization. The chapter positions the concept of Software Product Lines (<http://www.sei.cmu.edu/productlines/>), as a possible foundation for software industrialization, and the authors introduce the concept of Software Process Lines as complimentary foundation for software industrialization. The chapter discusses a number of questions: What are the Software Process Lines? What are the justifications and benefits of Software Process Lines? What are the steps for implementing Software Process Lines? How can Software Process Lines enable and facilitate the establishment of a continuous Software Process Improvement environment?

INTRODUCTION AND JUSTIFICATION

Mega Waves in the Software Industry

The software industry has moved a long way since the early times of it being a cottage industry. In its short history it has passed through a number of mega changes and waves of changes moving

firmly towards a discipline similar to what happened to manufacturing through the industrial revolution. The authors believe that the era of software industrialization is coming, in spite of some difficulties facing software development efforts. The first wave of the software industry started with the structured methods in the 1970's, which were developed to cater for the increasing demands and complexity of software. The efforts to adopt structured methods were assisted by automated

DOI: 10.4018/978-1-4666-1945-6.ch107

tools, and this was the beginning of transforming the software industry from a 'cottage industry' towards a disciplined production culture. The second wave of the software industry came with the rise of the object-oriented technology and its associated reuse, combined with the emergence of the focus of the software process maturity and improvement started by Watts Humphrey (Humphrey, 1989) and championed by the Software Engineering Institute (SEI). The software process movement had great promises towards injecting discipline and continuous improvement into the software industry. However this was not enough to move the software industry into the state of industrialization.

Research and industry survey show that the effort put into Software Process Improvement (SPI) models and frameworks can assist in producing high quality software, reducing cost and time, and increasing productivity and employee satisfaction (Ashrafi, 2003; Butler, 1995; Jiang, Klein, Hwang, Huang, & Hung, 2004; Niazi, Wilson, & Zowghi, 2006b; Pitterman, 2000; Yamamura, 1999; Zahran, 1998). However, only a small proportion of the software developing organisations' population has adopted SPI models and framework. This is depicted in the SEI CMMI appraisal report in which only 3113 CMMI appraisals were reported to the SEI (SEI, 2008). Research also shows that a large majority of software development organisations appear to be unwilling to follow SPI initiatives based on process capability maturity models like CMMI (Staples et al., 2007). This may be due to the fact that an SPI programme is quite an expensive undertaking as organisations need to commit significant resources over a long time period (Coleman & O'Connor, 2008). Even organisations who are willing to commit the resources and time do not always achieve their desired results from their SPI initiatives (Florence, 2001; Kautz & Nielsen, 2000). Hence, the significant investment required and limited success are considered two of the main reasons why many organisations are reluctant to embark on a long path of systematic

process improvement (Staples et al., 2007). In the case of small and medium size organisations, there are more concerns about the relevance and applicability of SPI models like CMM or CMMI (Brodman & Johnson, 1994; Staples et al., 2007).

Due to complex nature of SPI initiatives it has been proven to be hard for companies to successfully design and implement software process improvement. A typical approach to SPI in general has three stages: process appraisal, process definition/redesign and process implementation and deployment. The process appraisal phase consumes a larger percent of the budget and resources; process definition requires model knowledge, process design knowledge and skills and knowledge of the organisation/company; and often process implementation and deployment is not only across multi-project, but also across multi-site and multi-customer type. The whole SPI initiative is a long-term approach and it takes time to fully implement a SPI initiative. This is one of the reasons that many organisations are reluctant to embark on a long path of systematic process improvement.

Extensive literature review revealed that many standards and models exist for SPI but little attention has been paid to their effective adoption and implementation of these models. The chaotic implementation process is the most common cause of SPI implementation failure (Zahran, 1998). Attention to a defined SPI implementation process is essential for the success of any SPI initiative. In order to address the SPI implementation issues a Software Process Lines approach has been proposed in this chapter. The objective of the Software Process Lines is to reuse the artefacts in SPI initiatives which are costly to develop from scratch. This will not only help organisations to save their time for process definition/deployment but will also assist them in quickly moving from lower maturity level to higher maturity levels. We believe that the concept of Software Process Lines, introduced in this chapter, in association with the concept of Software Product Line, championed

13 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/software-process-lines/69378

Related Content

A LabVIEW-Based Remote Laboratory: Architecture and Implementation

Yuqiu You (2013). *Industrial Engineering: Concepts, Methodologies, Tools, and Applications* (pp. 726-742).

www.irma-international.org/chapter/labview-based-remote-laboratory/69312

Performance Analysis of Cloud Systems with Load Dependent Virtual Machine Activation and Sleep Modes

Sudhansu Shekhar Patra and Veena Goswami (2018). *International Journal of Applied Industrial Engineering* (pp. 1-20).

www.irma-international.org/article/performance-analysis-of-cloud-systems-with-load-dependent-virtual-machine-activation-and-sleep-modes/209377

The US National Building Information Modeling Standard

Patrick C. Suermann and Raja R.A. Issa (2010). *Handbook of Research on Building Information Modeling and Construction Informatics: Concepts and Technologies* (pp. 138-154).

www.irma-international.org/chapter/national-building-information-modeling-standard/39470

Industry 4.0 and the Internet of Things (IoT)

Zelal Gültekin Kutlu (2021). *Research Anthology on Cross-Industry Challenges of Industry 4.0* (pp. 152-170).

www.irma-international.org/chapter/industry-40-and-the-internet-of-things-iot/276816

Interdisciplinary Game-Theoretic Approach to Trust Modeling

Piotr Cofta and Hazel Lachée (2014). *International Journal of Applied Industrial Engineering* (pp. 1-13).

www.irma-international.org/article/interdisciplinary-game-theoretic-approach-to-trust-modeling/105483