Reuse Based Software Factory

Manu Prego
European Software Institute
Bilbao, Spain
Manu.Prego@esi.es

Abstract. This paper reports on an experiment on how it is possible to multiply the efficiency and quality of a software development process by means of industrial manufacturing organizations and methods and state of the art reuse and product-line technology. Of course all methods and technology claim the same objectives, therefore the experiment objective of dramatic improvements. This chapter does not report on toy-demonstrators but on a real application at a large IT development department of a large business user organization using conventional computer technology with high responsibility and reliability requirements. The solutions have since been deployed involving some hundreds of programmers, to the satisfaction of all stockholders in the business: users, developers, management and even owners. The experiment result, in a medium to large development organization, requires a significant investment which produces average Reuse Rates around 70-80%, effectively halves the cost and time-to-market of software development, and reduces the error rates several-fold. The method is compatible and complementary with software development models such as the Capability Maturity Model™, SPICE, ISO9001 or the new Agile software development methodologies.

1 Introduction

Generally speaking, and independently of the industrial sector of application, software development is characterised by problems such as: deadline slips, functional requirements sometimes far from those promised, overrun budgets, bad quality and high error rates. This has been the case since we started to develop software.

An analysis of the current status of software development in the world shows clearly that the quality and productivity of the industry has not been able to keep up with the software needs of Society in general, as shown in the Fig. 1. Computer science has therefore become an obstacle to progress.

The discipline of Software Engineering has been trying to find a solution to this problem [1] for a long time.

All these problems are only external symptoms of an evil not always evident: software development is a craft; a process that fundamentally depends on the person who carries it out; a process, of course, that uses continually improving tools, but a process in the end in which two developers using the same requirements and the same technology would obtain different results.

1 Source: ITEA - EC Internal Reflection Group on Software Technologies, April 2002
1.1 The Hypothesis Tested

This situation contrasts heavily with the world of manufacturing, the kingdom of Engineering, where the repeatability of results determines and defines the maturity of the processes.

By means of Engineering, improved processes and capital investment in effective tools, the Manufacturing Industry attain dramatically improved cost reduction, time-to-markets, quality and repeatability.

Therefore, why does not software development apply the concepts and methods of Manufacturing Engineering? In other words, is it possible to apply the concepts and methods of Manufacturing Engineering to software development? And, if it were possible, what results might we expect?

The hypothesis to be demonstrated can be described as follows:

A quantum leap is both needed and possible concerning the cost and time-to-market of software development, as well as to significantly increase quality, by means of the application of techniques mirrored from Manufacturing Engineering.

A corollary is that,

- Such improvements are not to be incremental, but dramatic: say two-fold or larger.
- As in the case of Manufacturing Engineering, such improvements would require large Reuse rates of standard components, and ideas taken from Product-Line theories.