Articles » Development Lifecycle » Work Issues » General

Importance of Processes and Standards in Software Development

Deepak Jain

11 Jan 2007 CPOL

Importance of Processes and Standards in Software Development

Introduction

Software, in last few decades, has captured a foremost arc of human life. It is now not a product of arbitrary and capricious practices and mere programming activities. Modern software products are engineered under the practice of using selected process techniques to improve the quality of a software development effort. This is based on the assumptions, subject to endless debate and supported by patient experience, that a methodical approach to software development results in fewer defects and, therefore, ultimately provides shorter delivery times and better value. The necessity of selecting and following a formal practice for software development is to provide desired discipline to deliver the quality expected for business success and avoiding the wastage of time, squander productivity, demoralization in developers, etc. This article summarizes such needs of adopting formal software development methodologies and standards.

Software Development: A Critical Engineering Task

Slowly and surely, computers and software are taking over many of the functions that effect our lives critically and they have become imperative parts of our lives. They are now monitoring and controlling all forms of monetary transactions, transportation, communication, defence systems, production and so on. They have made places in our homes, controlling all forms of appliances. Acknowledging their importance and complexities in structures and roles, software are no longer the products of arbitrary and capricious practices and mere programming activities. Software, not being merely a program to be executed to perform a task, is now an interaction of the programs, data-structure and documentation and is a complex structure to develop, test and maintain. Modern software products are engineered under the selected formal techniques to improve the quality of the product of a software development effort.

The major problem and probably the most critical task with software development is to know where to start from. Even worse situations arise when a project starts with new people in the team, new and unproven technology, unseen business domains and that too, with challenging deadlines. All these when attacking a software development plan, the management is on a risk of crisis. The products overshoot cost estimations and break schedules. They are poor in quality and often fail to respond. They do not meet requirement specifications as defined by consumers and finally, lead to a business failure.

Other challenges before software industry are to measure its intangible products, estimate processes define quality and furthermore, manage the risk. In spite of the enormous economic growth and productivity gains enabled by software, persistent complaints about the quality of software remain. All these make software development a critical practice.

According to a study in the University of Iowa, the basic challenges for software industry, which are most deserving of serious attention in the immediate future include:

1. creating the new logic for problem solving based on open-ended programming environments for high performance computer systems
2. developing a formal methodology that guides us towards the construction of correct and portable parallel programs, and adopting an openness to radical and innovative alternatives
3. designing a programming language that incorporates a unifying intuitive model of parallel computation, and which provides a coherent vehicle for the natural description of parallel programs