Chapter 3.14
Integrating Software Engineering and Costing Aspects within Project Management Tools

Roy Gelbard
Bar-Ilan University, Israel

Jeffrey Kantor
Bar-Ilan University, Israel, & University of Windsor, Canada

Liran Edelist
Bar-Ilan University, Israel

INTRODUCTION

Currently, there is no integration among CASE tools (computer aided software engineering, also named AMD tools, analysis modeling and design), costing tools, and project management (PM) tools. Not only are there no integrated tools, but there is also no conceptual integration among software engineering (SE) aspects and accounting-costing aspects of software projects within PM tools. PM tools, as well as costing tools are used not only for tracking and controlling an ongoing software project, but also at the very beginning stages of the project, in which critical estimations concerning budget and time frame are made. In order to have a firm, robust, and accurate planning, project planning should be based directly upon raw SE components-objects, that is, upon analysis and design components-objects.

According to the Standish Group CHAOS Report 2003, each year in the USA there are approximately 175,000 projects in IT Application Development which spends $250 billion. Among these, 31.1% of projects will be cancelled, 52.7% of projects will cost 189% of their original estimates, only 52% of required features and functions make

it to the released product, and Time overruns 82%. In financial terms $55 Billion dollars is wasted in these projects (Madpat, 2005).

Budget overrun indicates cost management problems, although this area is defined by the project management integration (PMI), as one of the nine core activities of projects management. Costing difficulties result from both implementation limitations of costing solutions in complex and changing requirements as well as the technological environment. Risk management is also defined by the PMI as one of the nine core areas of project management; but there is also no integration between PM tools and SE tools in light of the need for risk management.

According to Maciaszek and Liong (2005), success of a software project depends on five software engineering areas that are related to each other: the development of the life cycle of the software, processes management, the model's configuration and language, and SE tools and project planning. The combining between formal tools of SE and PM processes in the different stages has been proved by research as holding a positive contribution to the efficacy of the project and as an improver of the adherence to costs, technical requirements, and the schedules that were allocated to the project (Barker & Verma, 2003).

This study proposes and prototypes a model that integrates these three aspects of software projects by automatically mapping SE objects and accounting–costing objects into PM objects. To validate the feasibility of the model and without loss of generality, it is demonstrated using former research platform focused on conversion of data flow diagrams (DFD), which are actually full enterprise set of use cases diagrams reflecting entire system-software project into Gantt charts.

BACKGROUND

CASE and PM Tools

CASE/AMD tools support the analysis, design, construction, and implementation stages of the information system life cycle (ISLC) (Barker & Longman, 1992; Pendharkar, Subramanian, & Rodger, 2005; Sommerville, 2004). Commercial tools, such as IBM–Rational XDE, are covering main stages of ISLC; the "Requisite-Pro" module, for instance, is designated to the stage of requirement definition, "Rose" module to the analysis and design stage, and "Test-Studio" module to the testing stage.

Although PM tools support management and control along the ISLC, there is hardly any integration between CASE tools and PM tools. Thus, ISLC modeling approaches, such as the functional approach (e.g., DFD, ERD, STD), as well the object-oriented approach (e.g., use cases, activity diagrams, STD), even when automated, are used mainly in the early analysis stage primarily for visual documentation. The "database of specifications," laboriously elicited and gathered during the creation of modeling diagrams, is hardly ever applied again for project management purposes, even though this information is valuable for project managers who are involved in the construction and implementation stages. In fact, due to lack of integration along the ISLC, the specifications database is often either overlooked altogether or collected again as if their creation earlier never took place. Moreover, standard methods for system analysis and development usually make no reference to methods for project management. Accounting and costing parameters, which are reviewed at the next chapter, are not represented not at SE tools or at PM tools, and handled in totally separated systems.