12 Decision Support for Value-Based Software Release Planning

Sebastian Maurice, Guenther Ruhe, Omolade Saliu, and An Ngo-The

Abstract: Incremental software development replaces monolithic-type development by offering a series of releases with additive functionality. To create optimal value under existing project constraints, the question is what should be done when? Release planning (RP) provides the answer by assigning features to a sequence of releases in the most beneficial way within the resources available.

In this chapter, we extend the existing hybrid intelligence-based release planning method called EVOLVE* to accommodate financial value in the form of net present value estimates of proposed features. This extension enables us to perform financial value-based software release planning. The new approach called F-EVOLVE* is illustrated by an example. The results show that the F-EVOLVE* model may be used to decide which features to produce and when based on their financial contributions. Specifically, F-EVOLVE* may be used to determine which features generate the highest returns, with the shortest development time.

Keywords: Value-Based Software Release Planning, F-EVOLVE*, Decision Support, Hybrid Intelligence.

12.1 Introduction

In today’s world with rapidly changing consumer demands, informational technology, and competitive marketplaces, the requirements are changing rapidly requiring quicker adaptability by market participants. The critical success factor for vendors is responding to changing requirements quickly while maintaining a focus on their value proposition, which may be a quicker return on investments or an improvement in a public service like health, education, and defense (Boehm, 2003). A value-based approach to software engineering is closely aligned with the business goals and objectives of the organization, and seeks to ensure that every step in the software development process is a value-making step. Meeting the needs of customers is as important as justifying the development efforts needed to meet those needs (Poladian et al., 2003).

The need for a value-based approach to software release planning can be justified by the need for a faster time to market, while maximizing stakeholder satisfaction. Stakeholders are defined to be anyone that influences or is influenced by the project plan (Farbey and Finkelstein, 1999), and this includes customers, users, developers, project managers (decision makers), etc. For project success, within a value-based context, it makes more sense that traceability back to the value propositions become more important than traceability back to requirements (Boehm,
Decisions support for release planning plays a key role in identifying value propositions through a careful analysis of the release options and how these options impact value, financial, and human resources.

This chapter will contribute to an understanding of software release planning from a financial perspective with the objective of incorporating a more fine-grained measure of financial value to the original EVOLVE* model, thus creating a new model called F-EVOLVE*. Specifically, this financial measure of value will help to choose among competing features as well as choosing among release plan alternatives when integrated into the F-EVOLVE* model. Currently, value is considered based on a nine-point ordinal scale where the assigned stakeholder performs priority evaluations.

The chapter is organized as follows. Section 12.2 presents background information on release planning, which includes the discussion of existing approaches. Section 12.3 provides information on financially based release planning and the development of the F-EVOLVE* method. Section 12.4 presents an example to illustrate the new method. Section 12.5 concludes the chapter and provides directions for future work.

12.2 Background

A software release is a collection of features that form a new product. Without good release planning ‘critical’ features are jammed into the release late in the cycle without removing features or adjusting dates. This situation might result in unsatisfied customers, time and budget overruns, and a loss in market share (Penny, 2002). “Developing and releasing small increments of requirements, in order for customers to give feedback early, is a good way of finding out exactly what customers want, while assigning a low development effort” (Carlshamre et al., 2001).

Release planning is an integral part of incremental software development methods. These methods promote faster delivery of small components of the overall software product, where shorter time frames result in an iterative process of design, code, test, and deployment of these components (Larman and Basili, 2003).

Difficulties with Software Release Planning

Release planning is a very complex problem including different stakeholder perspectives, competing objectives, and different types of constraints (Ruhe and Ngo-The, 2004). Release planning is impacted by a large number of inherent constraints. Most of the features are not independent of each other. They typically have precedence or coupling constraints between them that need to be satisfied. Precedence constraint requires that specific features must be implemented before other features, while coupling constraints requires that some features must be implemented together. Furthermore, resource constraints such as effort and budget need to be fulfilled for each release.