Chapter VII

Requirements Risk and Maintainability

Norman F. Schneidewind
Naval Postgraduate School, USA

In order to continue to make progress in software measurement as it pertains to reliability and maintainability, we must shift the emphasis from design and code metrics to metrics that characterize the risk of making requirements changes. By doing so we can improve the quality of delivered software, because defects related to problems in requirements specifications will be identified early in the life cycle.

We provide an approach for identifying requirements change risk factors as predictors of reliability and maintainability problems. Our case example consists of twenty-four Space Shuttle change requests, nineteen risk factors, and the associated failures and software metrics. The approach can be generalized to other applications with numerical results that would vary according to application.

INTRODUCTION

While software design and code metrics have enjoyed some success as predictors of software quality attributes such as reliability and maintainability (Khoshgoftaar & Allen, 1998; Khoshgoftaar, Allen, Halstead, & Trio, 1996a; Khoshgoftaar, Allen, Kalaichelvan, & Goel, 1996b; Lanning & Khoshgoftaar, 1995; Munson & Werries, 1996; Ohlsson & Wohlin, 1998; Ohlsson & Alberg, 1996), the measurement field is stuck at this level of achievement. If measurement is to advance to a higher level, we must shift our attention to the front-end of the development process, because it is during system conceptualization that errors in specifying requirements are inserted into the process and adversely affect our ability to develop and maintain the software. A requirements change may induce ambiguity and uncertainty in the development process that cause errors in implementing the
changes. Subsequently, these errors propagate through later phases of development and maintenance. These errors may result in significant risks associated with implementing the requirements. For example, reliability risk (i.e., risk of faults and failures induced by changes in requirements) may be incurred by deficiencies in the process (e.g., lack of precision in requirements). Although requirements may be specified correctly in terms of meeting user expectations, there could be significant risks associated with their implementation. For example, correctly implementing user requirements could lead to excessive system size and complexity with adverse effects on reliability and maintainability or there could be a demand for project resources that exceeds the available funds, time, and personnel skills. Interestingly, there has been considerable discussion of project risk (e.g., the consequences of cost overrun and schedule slippage) in the literature (Boehm, 1991) but not a corresponding attention to reliability and maintainability risk.

Risk in the Webster’s New Universal Unabridged Dictionary (1979) is defined as “the chance of injury; damage, or loss.” Some authors have extended the dictionary definition as follows: “Risk Exposure=Probability of an Unsatisfactory Outcome*Loss if the Outcome is Unsatisfactory” (Boehm, 1991). Such a definition is frequently applied to the risks in managing software projects such as budget and schedule slippage. In contrast, our application of the dictionary definition pertains to the risk of executing the software of a system where there is the chance of injury (e.g., crew injury or fatality), damage (e.g., destruction of the vehicle), or loss (e.g., loss of the mission) if a serious software failure occurs during a mission. We use risk factors to indicate the degree of risk associated with such an occurrence.

The generation of requirements is not a one-time activity. Indeed, changes to requirements can occur during maintenance. When new software is developed or existing software is changed in response to new and changed requirements, respectively, there is the potential to incur reliability and maintainability risks. Therefore, in assessing the effects of requirements on reliability and maintainability, we should deal with changes in requirements throughout the life cycle.

In addition to the relationship between requirements and reliability and maintainability, there are the intermediate relationships between requirements and software metrics (e.g., size, complexity) and between metrics and reliability and maintainability. These relationships may interact to put the reliability and maintainability of the software at risk because the requirements changes may result in increases in the size and complexity of the software that may adversely affect reliability and maintainability. We studied these interactions for the Space Shuttle. For example, assume that the number of iterations of a requirements change — the “mod level” — is inversely related to reliability and maintainability. That is, if many revisions of a requirement are necessary before it is approved, this is indicative of a requirement that is hard to understand and implement safely — a risk that directly
Related Content

A Multi Agent Based Approach for Critical Components Identification and Testing
www.igi-global.com/article/a-multi-agent-based-approach-for-critical-components-identification-and-testing/104652?camid=4v1a

Comparison Between Internal and External DSLs via RubyTL and Gra2MoL
www.igi-global.com/chapter/comparison-between-internal-external-dsls/71818?camid=4v1a
Quality-Driven Model Transformations: From Requirements to UML Class Diagrams
www.igi-global.com/chapter/quality-driven-model-transformations/26834?camid=4v1a

The Logic Behind Negotiation: From Pre-Argument Reasoning to Argument-Based Negotiation
www.igi-global.com/chapter/logic-behind-negotiation/24148?camid=4v1a