*Extended Abstract for*

# Simulating Risk Factor Effects for Software Development Risk Management


**Dan Houston**
Industrial Engineering Dept.
Arizona State University
and Honeywell, Inc.

**Gerald Mackulak**
Industrial Engineering Dept.
Arizona State University

**James Collofello**
Computer Science and
Engineering Dept.
Arizona State University


## Introduction

One of the proposed purposes for software process simulation is the management of software development risks, usually discussed within the category of project planning/management (Kellner *et al.* 1999). However, modeling and simulation primarily for the purpose of software development risk management has been quite limited. A notable exception is Madachy's (1994) model. It was designed partially for the purpose of risk assessment, which was achieved through two inputs that represented combinations of COCOMO drivers heuristically graded for degrees of risk. This paper describes another approach to simulation for managing software development risks, that of researching common and significant software development risk factors and their effects, then adapting a base model as necessary for simulating the selected factors. This simulator is a tool designed specifically for risk management. We discuss its support for three risk management activities: risk mitigation, risk contingency planning, and risk intervention.

## Software Development Risk Factors (SDRFs)

Among the topics often discussed in the literature on software development risk management is the subject of risk factors. A variety of approaches have been used to investigate SDRFs. From them have emerged lists of risk factors, as well as taxonomies, questionnaires, and matrices for assessing software development risk. Some investigators have produced SDRF lists numbering on the order of 150 or more factors. A study of this literature (Houston 2000) revealed the existence of a set of common and significant SDRFs. Twenty-nine (29) of these were identified and selected for further study.

## Identifying Potential Risk Factor Effects

While a small but important body of work identifying SDRFs is accumulating, little work has been undertaken on the potential effects of these factors. Since knowledge of these effects is necessary for modeling and simulating SDRFs, a study was performed in two stages to identify and quantify the potential effects. In the first stage, software project managers participated in a qualitative survey, which utilized causal diagrams, and helped to identify significant potential effects of 29 risk factors. In the second stage, six of the 29 SDRFs were selected for further investigation (based on their importance and interrelationships) and a web-based quantitative survey was used to collect data on the actualized effects of the six factors. This data was used both to substantiate the results of the first survey and to develop quantitative relationships for the simulation model. Although the data did not substantiate some of the effects suggested by the literature and by the first survey, analysis revealed other statistical relationships not previously discussed in the software development literature. The data analysis also provided probability distributions for the random variables representing risk factor effects.

## Modeling SDRFs

A base model for the SDRFs was developed by combining features from two published software process simulators, Abdel-Hamid and Madnick (1991) and Tvedt (1996), then updating and extending the combination, particularly in the area of quality management. The base model represents a waterfall process and incorporates sectors for Planned Staffing, Actual Staffing, Effort Allocation, Project Planning, Project Control, Productivity, Work Flow, and Quality Management.

Though existing models incorporate some of the six SDRFs and their potential effects (Table 1), either in whole or part, none of them appear to have been developed specifically for the purpose of modeling particular risk factors. The present model demonstrates new modeling constructs developed for the six SDRFs. Requirements creep and morale effects are of particular interest in this regard.

Table 1. Six SDRFs Selected for Simulation

| **Risk Factor** | **Potential Effects** |
|---|---|
| Requirements creep | Increased job size and rework |
| Inaccurate cost estimation | Understaffing and inaccurate schedule |
| Excessive schedule pressure | Fluctuating productivity, exhaustion, low morale, weak reviews, higher error generation, and missed requirements |
| Lack of staff commitment, low morale | Lower productivity, higher error generation, attrition |
| Instability and lack of continuity in staffing | Attrition, lower morale, and loss of expertise |
| Lack of senior management commitment | Initial schedule compression, unwillingness to adjust schedule, and understaffing |

In addition to new modeling constructs, the model introduces the probability distributions and correlations derived from data collected in the quantitative survey. Though the base model is deterministic, it becomes a vehicle for propagating the uncertainty of risk factors, represented as random variables, thereby effecting stochastically modeled project outcomes that can be estimated as confidence intervals rather than as point estimates. Specifying outcomes in terms of confidence intervals, calculated specifically from selected risk factors, provides a strong statistical tool for supporting risk management.

## Model Usage

The model was designed to support decision-making for three kinds of risk management activities: risk mitigation, risk contingency planning, and risk intervention. In typical usage for risk assessment, runs are made for different combinations of probable risks. This also establishes baselines for risk management studies. For risk mitigation study, various plans may be simulated by setting inputs at the outset of a run. For risk contingency planning, triggers have been added to the model so as to simulate a risk mitigation activity upon the occurrence of a predetermined condition. For risk intervention, outputs provide the user with simulated project states so that a "project" may be paused and inputs may be manipulated, thereby simulating an unplanned change for reducing the effects of an actualizing risk.

**Model Results**

The model is currently undergoing validation and verification. The results, including model outputs and statistical validation, will be reported in the full paper.

**Expected Conclusions**

This paper has described how the study of SDRFs was undertaken to identify common and significant risk factors for modeling in simulation. Qualitative and quantitative surveys were used to study the factors and their potential effects. These survey results provided the basis for new modeling constructs and statistical relationships incorporated into a software process simulator. This simulator, with its ability to model development project outcomes stochastically, demonstrates that explicit modeling of risk factors and their potential effects provides an additional tool for software development risk management, particularly in the assessment and planning phases. Software project managers can use such a model to study the potential impact of various combinations of risk factors on project outcomes, then run simulations for risk mitigation plans, risk contingency plans, and interventions, all as means of elucidating their experience and supporting project management decisions.

**Further research**

Research into SDRFs is ongoing; research into their potential effects has begun. Continued research in this area is expected to provide further insights into and data for the development of simulators for software project risk management. Modeling constructs for risk factors is also evolving. Experimentation with risk-oriented simulators may provide insights into the relative influence of various risk factors and into best practices for project risk management. Though some SDRFs are quite common, some appear to be specific to certain software development domains (Jones 1994), which suggests domain-specific simulation models for risk management.

**References**

Houston, Dan (2000) *Software Project Risk Simulator*. Ph.D. Dissertation in progress.

Jones, Capers (1994) *Assessment and Control of Software Risks*. Englewood Cliffs, New Jersey: Prentice-Hall.

Kellner, Marc I., Raymond J. Madachy, and David M. Raffo (1999) "Software process simulation modeling: Why? What? How?" *Journal of Systems and Software* 46: 91-105.

Madachy, Raymond J. (1994) *A Software Project Dynamics Model for Process Cost, Schedule, and Risk Assessment*. Ph.D. Dissertation, University of Southern California.