Available online at www.sciencedirect.com
ScienceDirect
Information & Management 45 (2008) 175–182
INFORMATION & MANAGEMENT
www.elsevier.com/locate/im

Exploring the relationship between software project duration and risk exposure: A cluster analysis

Sun-Jen Huang *, Wen-Ming Han

Department of Information Management, National Taiwan University of Science and Technology, 43 Sec. 4, Keelung Road, Taipei, Taiwan

Received 5 September 2006; received in revised form 4 October 2007; accepted 5 February 2008
Available online 24 March 2008

Abstract

Software projects often fail. Thus it is important to find ways to ensure a successful outcome. One significant area is a better understanding of the relationship between the software project duration and risk exposure, as this helps project managers with pertinent information to be effective in managing risky projects. We addressed this need by adopting a cluster analysis technique to provide managers with insight into effective planning and control of their projects. The results not only revealed that risk exposures associated with *user*, *requirement*, *planning & control* and *team* risk dimensions were affected by project duration, but also showed how to manage software risks effectively through observing trends in the risk components. Based on our findings, project managers can adopt appropriate attitudes, skills, and practices to deal with risky areas more effectively rather than just identifying those software risks with which project managers should be concerned.
Published by Elsevier B.V.

Keywords: Software project management; Software risk management; Risk exposure; Project duration; Risk component

1. Introduction

Rapid development of new software products to meet customers' needs is essential today. Despite the fact that many organizations have invested money, time and effort to develop their software, the failure of many software projects is still frequent [8,27].

A software risk (an uncertain event or condition with negative consequences on a software project) can increase the failure rate of a project if it is ignored [9,15]. Thus, the main purpose of software risk management is to identify managerial and technical problems before they occur so that actions can be taken to eliminate or mitigate their impact [11]. Software risk management entails: quantifying the importance of a risk (assessing its probability of occurring and its impact on the project performance) and developing strategies to control it. Thus, understanding the nature of the various software risks and their effect on project success has become increasingly important (e.g., [12]).

Many studies have discussed software risks in two ways: identifying risks and examining the relationship between threats and risks. The former provides a framework as a checklist for project managers (e.g., [1,2,14]), while the latter creates patterns that show how software risks are affected by project characteristics and thus allows managers to develop an appropriate risk management strategy (e.g., [13,20]). We concentrated on the latter in our work.

Knowledge about the effects of project duration on software risk has not been previously investigated in depth. Though software projects have been plagued by schedule slips [6], the effectiveness of a risk management strategy could be improved by understanding the effect of project duration on risky areas. However, most studies have focused either on the types of uncertainties (e.g., [18,19,21,30]) or environmental contingencies [25]. Fewer have discussed the relationship between project duration and risk exposure [33], and they did not provide a systematic way to design a risk management plan based on their findings.

* Corresponding author. Tel.: +886 2 2737 6779; fax: +886 2 2737 6777.
E-mail addresses: huangsj@mail.ntust.edu.tw (S.-J. Huang), D9209202@mail.ntust.edu.tw (W.-M. Han).

0378-7206/$ – see front matter. Published by Elsevier B.V.
doi:10.1016/j.im.2008.02.001

Table 1
Summary of previously related studies

	Boehm (1991)	Barki et al. (1993)	Schmidt et al. (2001)	Wallace et al. (2004)
Project type	General	General	General	General
Scope	TRW	Quebec	Hong Kong, Finland, and the United States	Across Countries
Participant	Project Manager	Project Leader and User representative	Project Manager	PMI member
Participant numbers	Unknown	120	43	507
Research Method	Unknown	CFA	Delphi	SEM
Dimensions	0	5	14	6
Risks	10	23	53	27

An appropriate skill for managing significant risks can be based on understanding the trends of risk components [10]. Studies have concentrated on the effect of project characteristics on risk exposure, which is not separated by risk components (e.g., [4,24]). Therefore, very little guidance is available on developing a good risk management strategy.

2. Related work

Several previous studies have helped in identifying, assessing, and prioritizing software risks. For example, Boehm [5] proposed a software risk management framework that included risk assessment and risk control, and identified a list of the top-ten software risks based on his experience at TRW. However, two problems were identified in several later studies (e.g., [16]). Firstly, the list of top-ten risks was not produced through any formal model-building procedure; thus it lacked a theoretical foundation. Secondly, it reflected the risks of a software development environment in 1991. But since then the complexity, scale and diversity of software have increased and thus, the list has become inadequate unless it is calibrated.

After reviewing IS uncertainty and software risk literature, Barki et al. [3] conducted a survey in Quebec to develop a list which included 23 software risks, classified into five groups using factor analysis. Although the list provided a comprehensible instrument, Wallace et al. [32] pointed out that the assessment scale of each risk was excessively complex. To reduce the bias of a single-culture viewpoint, Schmidt et al. conducted a Delphi survey to integrate the options of experts from Hong Kong, Finland, and the United States. They identified 53 risk items, which were grouped into 14 types, and asserted that cultural difference could affect the list, and that only 11 software risks were applicable from a cross-cultural perspective [26]. Recently, Wallace et al. collected the opinions of 507 members in the Project Management Institute (PMI) and identified 27 software risks, which were classified into six dimensions: User, Requirement, Project Complexity, Planning & Control, Team and Organizational Environment using Structural Equation Model. A summary of related studies on software risks is given in Table 1.

In our study, the six risk dimensions of Wallace's work were adopted. Firstly, her work was conducted in 2004, and thus it was relatively up-to-date and reflected the consensus of 507 PMI members from various countries. Secondly, SEM was used in her work to examine and prove the composite reliability, convergent validity and adequacy of the proposed framework of software risks. Therefore, the six risk dimensions and their associated software risks, as shown in Table 2, were considered appropriate for our study.

3. Data collection and analysis technique

3.1. Data collection

To maximize the response rate, a Web-based survey was conducted. We collected data from recently completed software projects in 2005. The survey was made up of three sections. The first introduced the study and encouraged respondents to respond to the survey. The second asked respondents to provide seven project characteristics that described the background of the software project described: *process model*, *project duration*, *team size*, *average experience of project members*, *ratio of staff turnover*, *number of external suppliers* and *number of project manager replacements*.

The final part listed 27 software risks, and asked the respondents to give in the probability of their occurrence and their impact on the project schedule. Based on Boehm's original work, the *risk exposure* was defined here as the probability of occurrence of a risk factor multiplied by the impact on the project schedule. In order to ensure the consistency, the degree of probability of occurrence of software risks and their degree of impact on the project schedule were measured using the 5-scale criteria of the DoD Risk Assessment Method as shown in Table 3 [31]. For example, assuming that the probability of occurrence of a specific software risk was unlikely and would affect a major slip in key milestones, the probability of occurrence of that risk and its impact were rated as 2 and 4, respectively, based on the risk assessment criteria. Hence, the risk exposure of that software risk is equal to 8 (2×4).

A pretest was conducted to improve content validity through personal interviews with domain experts after