

An Empirical Analysis of Requirements Uncertainty, Task Uncertainty and Software Project Performance

¹Ping Lu, ²Xuemin Song, ³Yinqiu Song

¹School of Management,
University of Science and Technology of China/School of Management, University of
Science and Technology of China, No.96, JinZhai Road Baohe District, Hefei, Anhui
Province, 230026. P. R. China

²Master, School of Management, University of Science and Technology of China/ Room 513,
House 367, University of Science and Technology of China(East campus), No.96, JinZhai
Road Baohe District, Hefei, Anhui Province, 230026, P. R. China

³School of Management, Graduate University of Chinese Academy of Sciences/ Graduate
University of Chinese Academy of Sciences, NO.19, YuQuan Road Shijingshan District,
Beijing, 100190, P.R. China.

Corresponding Author: Xuemin Song

Abstract

Identifying user requirements is recognized as a critical step in the development of software and an important area of research. With the rapid changes of customer-driven environment, user requirements have become increasingly hard to predict and control. The team tasks of software design and development are intrinsically complex and often exacerbated by incomplete user requirements. Given all of that, requirements and task uncertainty are very important factors affecting software project performance. The main purpose of this research is to examine the relationships among requirements uncertainty, task uncertainty and software project performance, based on a questionnaire survey of 168 Chinese software development staffs. Analyzed data shows that both requirements uncertainty and task uncertainty are negatively related to software project performance. Moreover, the result shows that there is a significance positive association between requirements and task uncertainty. Findings in this research can help practitioners understand influencing factors and draw more attentions from Chinese managers, which could enhance software project performance.

Keywords: requirements uncertainty, task uncertainty, software development, product performance, process performance.

INTRODUCTION

With the development of informatization, the number of software development project is increasing, but the result of the project is not optimistic. According to the Standish Group survey (2007), along with more and more attentions and continuous investments, the project's success rate, which is rising, is still only 34%.

Some researches point out that most of the problems are caused by the uncertainty in the process of software development. As the core factor of software development process, requirements uncertainty has an important influence on the project (Zmud, 1980). During the requirement analysis stage, for lack of IT knowledge, it is difficult for users to express their own requirements, thus some invisible requirements are hard to find, which affect the completion of task. According to foreign researchers, task uncertainty also has an important influence on software development project. However, the degree of this influence has not been completely verified, and the

relationship between requirements uncertainty and task uncertainty also remains to be further study. In China, the existing researches mainly summarize factors influencing the software project performance, and pay more attention to individual factors, which does not take the relationships among influencing factors into account. Besides, compared with theoretical research, empirical research is less.

Therefore, based on Chinese software projects as the research object, this study examines the relationship between requirements uncertainty and task uncertainty and their impacts on software development performance. Research conclusion helps to understand influencing factors and draw more attentions from Chinese managers.

LITERATURE REVIEW AND RESEARCH HYPOTHESES

❖ Requirements Uncertainty

Nidumolu (1995) deeply studies the requirements uncertainty. Based on the analysis of 64 projects, he

puts forward that the requirements uncertainty mainly displays in three aspects: requirements instability, requirements diversity and requirements analyzability. In the process of software development, the users will constantly add new additional requirements or functions. Thus the development process is varying, software function is difficult to finally determine. Requirements instability reflects the extent of changes in user' requirements over the course of development. Requirements diversity reflects the extent to which users differ among themselves in their requirements. Requirements analyzability reflects the extent to the process of converting. Most researchers confirm this classification.

❖ The Relationship between Requirements Uncertainty and Software Projects Performance

About software project performance, there are different definitions among different researchers. For example, Aladwani (2002) uses work output, psychological output and organizational output to measure project performance. Rai, Aldaijy and Nidumolu (2000, 2002, 1996) scholars define it as "the degree of success in the aspect of product and process during development", including product performance and process performance. Product performance refers to the results of software project development, that is, products achieve expected goals. It helps to ensure customer satisfaction and acceptance and reduced maintenance. Process performance has been defined as the degree to which the process is designed to promote consensus among people participating in the development process, operate within established resources parameters, and reduce waste and redundancy, mainly including two aspects: project budget performance and schedule performance. Coopriider and Henderson (1990) suggest that examining product and process outcomes together can reveal differential impacts of them on quality. Thus this method is widely used.

Nidumolu (1995) first discussed the relationship between requirements uncertainty and software project performance. He points out that requirements uncertainty has significant positive correlation with residual performance risk, and residual performance risk also positively correlates with project process performance and product performance. On the basis of Nidumolu's researches, combined with different situations of software project development, some researchers further study the relationship between requirements uncertainty and software project performance. On the basis of 1999-2000 software project materials from South Korea, Sik confirms Nidumolu's research results and points that in developing countries without advanced information technology, residual performance risk is not fit to be used as an intermediate variable and requirements

uncertainty negatively impacts software project performance (Kwan-Sik et al., 2004). Julie suggests requirements instability and requirements diversity are negative related with software project performance (Julie et al., 2011). On the basis of these, some hypotheses has been given as follows:

Research Hypothesis 1: The requirements uncertainty is negatively associated with software project performance

Research Hypothesis 1a: The requirements instability is negatively associated with software project performance.

Research Hypothesis 1b: The requirements diversity is negatively associated with software project performance.

Research Hypothesis 1c: The requirements analyzability is negatively associated with software project performance.

❖ The Relationship between Requirements Uncertainty and Task Uncertainty

Meeting users' demands and realizing the goal of the project depends on specific task execution. And requirements uncertainty leads to the frequent changes of task. Some researchers divide task uncertainty into two dimensions: task ambiguity and task complexity.

Task uncertainty (TU) is defined as "the degree to which work to be performed is difficult to understand and complex" (Aldaijy, 2002). Task ambiguity refers to those tasks for which multiple acceptable solutions exist, according to different frames of reference (Sussman et al., 1999). Task information that is clear leads to similar interpretations, while task information that is ambiguous leads to multiple interpretations that must be resolved in order to develop a shared understanding of how to perform the task. Task complexity refers to the number of inputs, input variation, number of sub-tasks and number of operations or procedures involved in the completion of a task (Spence et al., 1997). For information processing or decision-making, a task that utilizes fewer information cues is considered as having lower task complexity than one with more cues.

As is known to us, the requirement analysis, as the first step in software development, is crucial. The task scope largely depends on the requirement analysis. When users constantly change their own needs, tasks become blurred and even difficult to execute. Zmud (1980) thinks users' requirements come from the changing environment, thus task related to requirements will reflect high uncertainty. Aldaijy's research (2002) shows that the requirements uncertainty is positively associated with task uncertainty; requirements uncertainty will cause

task uncertainty. So the following hypothesis has been drawn:

Research Hypothesis 2: The requirement uncertainty can lead to task uncertainty.

❖ The Relationship between Task Uncertainty and Software Project Performance

Task complexity and task ambiguity may affect software project schedule, cost, etc. Some scholars think that task uncertainty also influence software development process. For example, Nidumolu's research (1995) shows that task uncertainty is associated with low software project performance and high software performance risk. Through investigating 47 software development project teams, Sussman further verifies the relationship between task uncertainty and software project performance, which is proved to be negative relationship (Sussman et al., 1999). According to the research about the relationship among software development model, task uncertainty and software projects performance, Rai (2000) shows that task uncertainty is negatively associated with software project performance. The related hypothesis is following:

Research Hypothesis 3: The task uncertainty is negatively associated with software project performance

Research Hypothesis 3a: The task ambiguity is negatively associated with software project performance.

Research Hypothesis 3b: The task complexity is negatively associated with software project performance.

To sum up, this study gives a research framework as shown in figure 1.

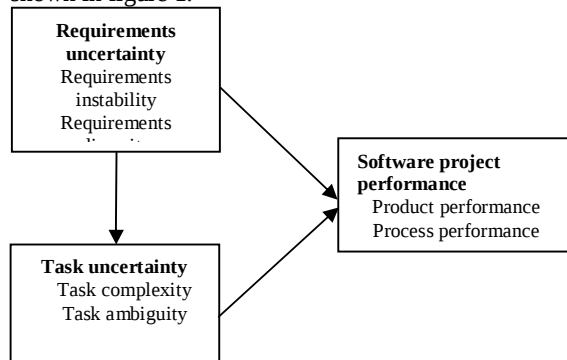


Figure 1: Relationship among the requirements uncertainty, the task uncertainty and the software project performance

RESEARCH METHODS

Sample

This study adopts the judgment sampling method. The sampling frame is drawn from Hefei's and Suzhou's software enterprises. 200 questionnaires are mailed to the IS professionals, who are invited to voluntarily and anonymously respond to the survey based on their recent experience in an IS project. Of the 200 questionnaires, there are 176 questionnaires returned and the number of valid responses is 168, that is, response rate is 84%. Among the questionnaires, 89% of questionnaires come from men and 11% from women; the rate of respondents who have been engaged in software development work for 1 to 3 years is 35.7%; and the rate for over 3 years is 43.5%. SPSS16.0 software has been chosen as data analysis tool.

Constructs

❖ Requirements Uncertainty

The scale in Nidumolu's, Aldaijy's and Julie's researches has been used to measure requirements uncertainty, which divides requirements uncertainty into requirements instability, requirements diversity and requirements analyzability (Nidumolu, 1995, 1996 ; Aldaijy, 2002 ; Julie et al., 2011). This scale contains 10 items, three for measuring requirements instability, three for measuring requirements diversity, and four for measuring requirements analyzability (results need to be reversed). A 5-point Likert-type scale that ranged from strongly disagree to strongly agree is used for each of requirements uncertainty aspects. In this study, some items have been corrected for higher reliability. And only 9 items remain, deleting the item "requirements fluctuated quite a bit in early phases". In this study, Cronbach's method of Alpha Reliability Coefficient has been introduced. After adjustment, Alpha value of requirements uncertainty, requirements instability, requirements diversity and requirements analyzability is 0.694, 0.800, 0.740, and 0.684. Although Cronbach's α of requirements uncertainty and requirements analyzability are less than 0.7, it is also acceptable considering the deviation. Through factor analysis, KMO is 0.695; the total variance explained is 63.92%.

❖ Task Uncertainty

Task uncertainty scale is based on the study of Sussman, and task uncertainty is divided into task ambiguity and task complexity (Sussman et al; 1999). This scale includes five items, which ranges from strongly disagree to strongly agree by 5-point Likert-type scale, including three items for task ambiguity and two for task complexity. The total Alpha value is 0.75, and the result of factor analysis is as follows:

Table 1. Task uncertainty factor analysis results

Items	Alpha	total variance explained
1. To what extent do multiple views exist of how the final system should look?		
2. During system development, to what extent can information be interpreted in different ways, which can lead to different but acceptable solutions?	task ambiguity 0.764	40.37%
3. To what extent do multiple views exist of how the final system should be developed?		
4. How technically complex is the system being developed?	task complexity 0.714	31.59%
5. To what extent are the technical problems for this system particularly complicated?		

❖ Software Project Performance

The software project performance scale comes from Nidumolu, Rai, Aldaijy, Wallace and Keil's related research, which is divided into product performance and process performance (Rai, 2000; Aldaijy, 2002; Nidumolu, 1995, 1996; Wallace, 2004). This scale contains 10 items, five for product performance and five for process performance. 5-point Likert-type scale is also used. After analysis, total Alpha value is 0.878, Cronbach's Alpha value of the product performance and process performance is 0.854 and 0.827, and total variance explained is 62.02%.

RESULTS AND DISCUSSION

Statistical Analysis

In Table 2, description statistic of each variable and the relationship among them has been given. It's easy to find that requirements uncertainty is negatively associated with software project product performance and process performance. Meanwhile, three aspects of requirements uncertainty are also negatively with software project product performance and process performance, which is consistent with hypothesis. In

addition, the conclusion can be drawn that requirements instability is positively associated with requirements diversity, which shows that both are existing at the same time.

The relationship between requirements uncertainty and task uncertainty is significantly positive. From table 2, it is obvious that both requirements instability and requirements diversity are significantly associated with two aspects of task uncertainty. But the relationship between requirements analyzability and task ambiguity or task complexity has been not confirmed. To the detail, task ambiguity is negatively associated with software project performance, but the relationship between task complexity and software project performance was not significant, which contracted with the foreign researches. So hypothesis 3 didn't get all support. What's more, we may safely draw the conclusion that the relationship between task ambiguity and task complexity is significantly positive indicating the co-existence.

Table 2. Description statistics and variable correlation analysis (n = 168)

	Mean	SD	1	2	3	4	5	6	7	8	9	10
1.instability	2.56	0.82	1									
2.diversity	3.04	0.74	0.438**	1								
3.analyzability	2.31	0.62	0.032	0.117	1							
4.uncertainty	2.61	0.48	0.617**	0.759**	0.583**	1						
5.ambiguity	2.68	0.76	0.648**	0.532**	0.08	0.567**	1					
6.complexity	2.81	0.85	0.408**	0.164**	-0.13	0.185*	0.384**	1				
7.task	2.73	0.66	0.661**	0.459**	-0.026	0.482**	0.873**	0.772**	1			
8.product	3.66	0.67	-0.189**	-0.238**	-0.297**	-0.352**	-0.295**	0.009	-0.171*	1		
9.process	3.53	0.67	-0.166*	-0.228**	-0.291**	-0.323**	-0.242**	0.098	-0.107	0.468**	1	
10.project	3.60	0.60	-0.208**	-0.265**	-0.331**	-0.386**	-0.303**	0.055	-0.160*	0.847**	0.842**	1

Note: **P<0.01(2-tailed), *P<0.05(2-tailed), Spearman.

test value is used to test the independence of explained variable residuals. D-W test values of

Hierarchical Regression Analysis

Considering the multicollinearity and correlation between explained variable residuals, this paper adopts hierarchical regression analysis method to further study the relationships among the variables (table 3). Multicollinearity refers that the influence that multiple independent variables have on dependent variable will be weakened, when the relationships among them from regression equation is high correlated. The more variance inflation factors (VIF) are, the more serious multicollinearity are. In table 3, VIF is ranging from 1 to 5, which shows that multicollinearity is not serious. Durbin - Watson

model 2 and model 4 are 1.738 and 1.839, indicating that autocorrelation among explained variable residuals does not exist.

In table 3 after joining task uncertainty, the ΔR^2 increases by 3.2%, and all the models are still significant, which shows that task ambiguity is negatively associated with software project product performance, and task complexity has no significant influence on product performance. From results of model 3 and model 4, after ΔR^2 increases by 3.7%, the models are also still significant. However model 4 shows that the relationship between task ambiguity

and the process performance is not significant, while the task complexity is negatively associated with

software project process performance.

Table 3. Hierarchical regression test (n = 168)

Independent variable	Model 1	Model 2	Model 3	Model 4	VIF
Requirement instability	-0.125	-0.066	-0.113	-0.098	1.794
Requirement diversity	-0.152*	-0.071	-0.129	-0.063	1.505
Requirement analyzability	-0.239**	-0.219**	-0.225**	-0.198**	1.041
Task ambiguity		-0.229*		-0.185	2.166
Task complexity		0.137		0.194*	1.268
F	7.740**	5.992**	6.194**	5.237**	
R ²	0.124	0.156	0.102	0.139	
Adjusted R ²	0.108	0.130	0.085	0.113	

Note : **P<0.01(2-tailed), *P<0.05(2-tailed).

Dependent variable from Model 1, Model 2 is product performance. Dependent variable from Model3, Model4 is process performance. The last column is VIF about model 2 and model 4.

CONCLUSION AND IMPLICATIONS

Traditionally, in order to solve the problem of software project implementation, factors influencing software project performance have been studied at home and abroad. Existing researches in China mainly summarize the single factors in the theory, and empirical research on the relationship between requirements uncertainty and software project performance is few.

Through investigating 168 I-S professionals in China, this research studies the relationship among requirements uncertainty, task uncertainty and software project performance. The study shows that both requirements uncertainty and task uncertainty are negatively associated with software project performance, which is consistent with Rai, Aldaijy, Sussman's research conclusion (Rai, 2000; Aldaijy, 2002; Sussman et al., 1997). Requirements uncertainty can lead to task uncertainty. Thus the importance of requirements uncertainty should not be neglected. High requirements uncertainty could affect project performance, and lead to high task uncertainty. The methods of Standardization and knowledge management could be used to reduce requirements uncertainty. Requirements and task uncertainty are pervasive phenomena that plague software development. An important contribution is that combined examination of both requirements and task uncertainty would lay foundations for further studies. In addition, the findings of my research would help practitioners and managers better understand the requirements process and reduce uncertainties associated with the project performance.

LIMITATIONS OF THE STUDY

is study also has some limitations. On one hand, the response rate of this study is limited. The sample size of 168 is not small compared to other studies, but the sample size should be seen as large enough to draw some basic conclusions. So future studies can increase sample size. On the other hand, while the co-

existing of task ambiguity and task complexity has been confirmed, whether this relationship could affect software project performance is still an open research question. Future studies may be able to shed more lights on the impacts of task uncertainty on software project performance by separating these two components.

REFERENCES

- Aladwani, A. M.2002. IT project uncertainty, planning and success, an empirical investigation from Kuwait. *Information Technology & People*, 15(3), 210-226.
- Aldaijy, A. Y. 2002. An Empirical Assessment of the Impact of requirements uncertainty on development Quality Performance. George Madon University.
- Coopridier, J., Henderson, J. 1990. Technology processes fit, perspectives on achieving prototyping effectiveness. *Journal of management information systems*, 7 (3), 67-87.
- J. Spence, R. Tsai. 1997. On human cognition and the design of information systems. *Information & Management*, (32), 65-73.
- Julie Yu-Chih Liu, Hun-Gee Chen. 2011. Relationships among interpersonal conflict, requirements uncertainty, and software project performance. *International Journal of Project Management*, (29), 547-556.
- Kwan-Sik Na, Xiaotong Li. 2004. Uncertainty profile and software project performance, a cross-national comparison. *The Journal of Systems and Software*, (70), 155-163.
- Nidumolu, S. R. 1995. The effect of coordination and uncertainty on software project performance, residual performance risk as an intervening variable. *Information Systems Research*, 6(3), 191-219.
- Nidumolu, S. R.1996. Standardization, requirements uncertainty and software project performance. *Information Management*, (31), 135-150.

Rai, A., Hindi, H. 2000. The effects of development process modeling and task uncertainty on development quality performance. *Information & Management*, 37(3), 335-346.

Standish Group. 2007. The CHAOS Report. <http://www.standishgroup.com/chaos.html>, Standish Group International, Inc., 2007.

S. Sussman, P.Guinsan. 1999. Antidotes for high complexity and ambiguity in software development. *Information & Management*, (36), 23-35.

Wallace, L., Keil, M. 2004. Software projects risks and their effect on outcomes. *Communications of the ACM*, 47(4), 68-73.

Zmud, R.W. 1980. Management of large software development efforts. *MIS quarterly*, 4(2), 45-55.