Available online at www.sciencedirect.com
ScienceDirect
International Journal of Project Management 30 (2012) 27 – 36
International Journal of Project Management
www.elsevier.com/locate/ijproman

Users as knowledge co-producers in the information system development project

Jack Shih-Chieh Hsu, Tung-Ching Lin *, Guang-Ting Zheng, Yu-Wen Hung

Information Management Department, National Sun Yat-sen University, Kaohsiung, Taiwan ROC.

Received 31 January 2011; received in revised form 9 May 2011; accepted 31 May 2011

Abstract

This study investigates the role of users as knowledge co-producers in different stages of the information system development project. Data collected from 269 IS professionals supported our hypotheses that common knowledge has an impact on requirement determination, which, in turn, leads to better project performance. User–IS relationship can substitute the effect of common knowledge on requirement determination in the design stage and user review can ensure that the obtained requirements are actually carried out by developers in the development stage. As a result, higher project performance can be obtained. Discussion of the results and conclusions is also provided.

Keywords: Knowledge co-production; Project performance; Requirement determination; User–IS relationship; User review

1. Introduction

The Management Information System department in an organization has long been regarded as a support function. Moreover, information system development (ISD) work is treated as a crafting artifact to support business operation. It is understood as a process through which developers transform user requirements into system design and then implement the designed system to satisfy these requirements. However, researchers have found that projects are cancelled or cannot be completed within predefined budgets and costs because the developed outcome does not meet the users' requirements (Doherty and King, 2001; Nidumolu, 1995; Wallace and Keil, 2004). One major cause of ineffective system development is a lack of user engagement in the development process. Theorists point out that including users in the development generates a positive impact on the process by increasing productivity and improving users' attitudes toward the system (He and King, 2008; Hunton and Beeler, 1997; Ives and Margrethe, 1984; Markus and Mao, 2004). The consequence of denying users the opportunity for engagement is that extra costs and time are required for remedial work when the final systems do not ultimately meet the users' required functionality and requirements (Procaccino and Verner, 2009).

An emerging perspective, named service-dominant logic, suggests that customers may act as value co-producers (Payne et al., 2008). Co-production is an active, creative and social process, based on collaboration between producers and users, that is initiated by the firm to generate value for customers (Coates, 2009). ISD can be viewed as a value co-production process in which users and developers work closely to determine the system requirements and implement the resulting system to support organizations' daily operation. Given that users operate the developed system in their daily work, they are considered as the final customers of the ISD service. As ISD is a knowledge intensive process, the developed system may be viewed as a new knowledge which combines developers' IT knowledge and business users' domain knowledge. The value created through the development process is that of co-production, resulting in a system which can be viewed as new knowledge co-produced by users and developers. Users are

* Corresponding author.
E-mail addresses: jackshsu@mis.nsysu.edu.tw (J.S.-C. Hsu), tclin@mis.nsysu.edu.tw (T.-C. Lin), nsysuichiro@gmail.com (G.-T. Zheng), ywhung526@gmail.com (Y.-W. Hung).

0263-7863/$36.00 - see front matter © 2011 Elsevier Ltd. APM and IPMA. All rights reserved.
doi:10.1016/j.ijproman.2011.05.008

encouraged to engage in the development process to enhance the value of the developed system by avoiding an outcome that falls short of actual need. In avoiding this danger, the additional costs and time required to repair inappropriate design in the early stages of development work can also be avoided. This indicates the importance of users, who should not be ignored when pursuing high project performance.

Past user participation studies treated user participation as a second-order construct and examined its direct impact on final outcomes (Ives and Margrethe, 1984; McKeen et al., 1994). This over-parsimonious understanding of relationship reduces the power of their argument, leads to inconclusive results, and increases the difficulty in understanding the role users can play in different development stages. Furthermore, while studying the impact of user participation, past studies largely focused on user satisfaction, acceptance, or system quality, viewing the issue solely from users' perspective. Such an exclusive focus has resulted in failure to consider the extent to which user participation can have an impact on project performance. It has also led to the omission of the developers' perspective. Ravichandran and Rai (2000) suggest that instead of understanding the direct relationship between user participation and development outcomes, researchers should shift their focus to exploring how the development process is influenced by user representatives.

The ISD process can roughly be separated into system design and development stages. This study aligns with co-production literature which highlights the valuable role played by customers in these two stages in terms of providing requirements and facilitating the development process (Ives and Margrethe, 1984). Therefore, this study aims at answering "*how can users act as knowledge co-producers in the design and development stages?*" We hypothesize that to ensure the accomplishment by the project team of the predefined goal within budget and on schedule, users can contribute their expertise to facilitate requirements determination and improve the system development process.

The remainder of the paper is organized in the following way. In the next section, we first review related literature and develop hypotheses on the basis of the review. The third section of this paper introduces the research methodology. The fourth section details the results of the study. Finally, discussion and conclusions are provided.

2. Literature review and hypotheses development

2.1. Users as knowledge co-producers

ISD can be viewed as a problem-solving process in which developers apply their knowledge to solve problems raised by users. This process involves intensive knowledge. In general, in order to carry out information system development, members of the team must possess sufficient knowledge-based resources, such as project management knowledge, system analysis skills, programming knowledge, database administration knowledge, etc. In addition to system development knowledge, business knowledge is one of the critical resources for successful system development (Bassellier et al., 2001). However, the present various types of knowledge in the team does not guarantee the final performance. To pursue common goals in projects requires different stakeholders to transform their individual-level knowledge into collective knowledge. Therefore, knowledge possessed by users and developers need to be accessed, leveraged, shared, and maintained for the benefit of the project (Martinsuo and Kantolahti, 2009). The effectiveness of problem solving or uncertainties countering relies on how well users and developers can integrate these two types of knowledge to generate new knowledge to solve problems.

Traditional user participation research has highlighted the importance of having users in the development process (e.g. He and King, 2008). User participation was originally viewed as a user and developer "*cooperatively involved to the extent that the activities of each facilitate the attainment of the ends of the others*" (Swanson, 1974). This definition implies that users should be viewed as co-producers who work with developers harmoniously so as to carry out the final system. The recently emergent service-dominant logic concept also asserts that the effectiveness of value creation relies on the extent to which those operant resources (such as knowledge or competence) possessed by customers can be incorporated into the service design and development process (Stuart and Tax, 2004). The primary concept of service-dominant logic includes: (1) the conceptualization of service as a process, rather than a unit of output; (2) a focus on dynamic resources, such as knowledge and skills, rather than static resources, such as natural resources; and (3) an understanding of value as a collaborative process between providers and customers, rather than what producers create and subsequently deliver to customers. By applying service-oriented concept in ISD projects, users can be treated as co-producers and should not be excluded from the process. Instead, on the basis of the understanding that a software project is a knowledge intensive process and the developed system is a co-produced new knowledge, (for example, software outsourcing vendors are called knowledge intensive business service firms (Bettencourt et al., 2002)), users should play a more active role and be viewed as knowledge co-producers in different stages of the ISD process so as to improve the service delivering process and the quality of service delivered.

Users may engage in different ISD stages. For example, separating the outcome of participation into productivity and attitudinal/affective dimensions, He and King (2008) concluded that user participation can increase productivity by engaging in the requirement determination process. In addition, viewing user participation as an uncertainty reduction approach, Hsu et al. (2008) indicated that users can play a role in the system development stage to avoid low system quality caused by environmental uncertainties. Although past studies pointed out the function of user participation in different stages, they focused on one stage only and emphasized its impact solely from the users' perspective. Therefore, whether project performance can be enhanced through this process remains unclear. By classifying ISD into design and development stages, this study argues that the user can have an impact on the process by undertaking various activities in the different stages.

Furthermore, we examine the proposed concept from the developers' perspective and explore the conditions necessary for users to act as co-producers. In the following section, we discuss the role that users can play as knowledge co-producers during the design and development stages.

2.2. Design stage: Requirement determination as knowledge integration

Since knowledge is one of the most critical resources in an ISD project, the lack of adequate knowledge leads to risks (Gemino et al., 2007; Nidumolu, 1995), increases uncertainty (Iacovou et al., 2009), and inhibits the learning process (Ramasubbu et al., 2008). However, having the required knowledge alone does not guarantee the final outcome of the project. Rather, effective system development requires different types of knowledge to be integrated so as to counter uncertainties and complexity (Martinsuo and Kantolahti, 2009). One project management study pointed out that project performance is determined by the level of success with which developers and users integrate their owned knowledge, which is affected by the extent to which they understand each other (Tesch et al., 2009). This implies that the possession of business knowledge for developers and the possession of ISD knowledge for users enable both parties to understand and to participate in the other's key processes and to respect each other's unique contribution and opinion (Tiwana and McLean, 2005).

The business knowledge of the IS developer is defined as the set of business and interpersonal knowledge and skills possessed by IT professionals that enable them to understand the business domain, speak the language of business, and interact with their business partners (Bassellier and Benbasat, 2004). The users' IT knowledge and skills refer to users' overall knowledge and experience in the IS development tasks and processes (Tesch et al., 2009). IT knowledge and skills is the know-how needed to develop IT applications and to operate those applications to fulfill relevant tasks. It includes knowledge of programming languages, experience of operating systems, and understanding of communication protocols and products. Common knowledge is the combination of developers' business knowledge and users' IT knowledge.

Grant (1996) advocated that owned domain knowledge and common knowledge are essential for effective knowledge integration between users and developers. Shared understanding or common knowledge facilitates learning and reduces miscommunication (Mohammed and Dumville, 2001). Given that the amount of common knowledge between users and developers is fundamental for achieving mutual understanding, the IT knowledge of users and the business knowledge of developers thus play an important role in improving the effectiveness of user–IS knowledge co-production.

Since the main purpose of users engaging in the development process is to contribute their knowledge to determine actual requirements (He and King, 2008), to ensure that user requirements indeed can be incorporated into system design, developers need to integrate users' knowledge with their own (Robillard, 1999). According to Patnayakuni et al. (2007), when a common base of knowledge has been captured, shared, and formalized, knowledge can be integrated and the resulting solution will more likely to satisfy end users' needs. That is, the business-related knowledge of IS developers serves as a driver which allows them to communicate with users to ensure effective knowledge integration (Preston and Karahanna, 2009). If users are familiar with IT knowledge and have experience of IS development, it is more likely for them to express their needs in a way that developers can easily respond to. On the other hand, developers are able to understand users when developers possess strong business knowledge, allowing the resulting system design to better reflect users' needs. Therefore, we predict that during the ISD process, knowledge co-production between developers and users facilitates efficient development of a software solution that is more likely to reflect its intended objectives. Thus, we set the following hypothesis:

H1. The level of common knowledge is positively associated with the effectiveness of requirement determination.

The user–IS relationship can be defined as the level of mutual trust, respect, and closeness of relationships between users and developers (Kale and Singh, 2000). In the context of ISD, a strong partnership between users and developers is required for efficient planning and the developing of new applications (Ross et al., 1998). However, the majority of knowledge integration does not go through the established or documented procedures (Martinsuo and Kantolahti, 2009). It implies that the integration of knowledge possessed by developers and users into project-relevant activities requires each other's trust and respect (Tiwana and McLean, 2005). Higher levels of relational capital enhance the likelihood of developers and users being willing to exchange and combine their domain knowledge during the ISD process (Szulanski, 1996; Tiwana and McLean, 2005). According to strong-tie theory, a close relationship between both parties is necessary to enable each to contribute their individual level knowledge to form project level knowledge (Granovetter, 1973). When there are strong and trusting relationships between developers and users, the costs of communication, coordination and combination of each other's knowledge will decrease, and in turn, facilitate the effectiveness of user–IS knowledge combination (Robert et al., 2008). Thus, this study suggests that the relationship between users and developers is an important component which enhances the effect of common knowledge on the user–IS knowledge co-production process. Therefore, we hypothesize that:

H2. The magnitude of the impact of common knowledge on requirement determination is influenced by the quality of the relationship between users and developers.

2.3. Development stage: Assuring determined requirements can be actually carried out

Project management literature defined project outcome as the ability to meet project goals within a predefined budget and schedule (Jiang et al., 2001; Schwalbe, 2002; Yetton et al., 2000). Project performance is determined by various factors, one of the most critical being whether the actual user requirements are captured in the system design stage. The system design work can be viewed as a process in which users express the business needs and system analysts transform those needs into system design on the basis of their information system design knowledge. It can also be viewed as a process of integrating the business knowledge of these two parties. Once the design work has been completed, coding work is then assigned to individual programmers. Correct functions can be developed if system analysts are able to transform user requirements into system design. In contrast, performance is impaired if system design cannot reflect actual users' needs, making remedial work unavoidable to correct the inadequate designs. This, in general, results in schedule delay and extra costs. Empirical studies also indicated that failure to integrate existing knowledge is one of the major barriers to producing high project performance (Mitchell and Nicholas, 2006; Patnayakuni et al., 2007). Bassellier et al. (2003), and Nissen and Jennex (2005) indicated that the successful integration of differentiated knowledge during the ISD project is a critical factor for project success. Therefore, we hypothesize that:

H3. The effectiveness of requirement determination is positively associated with project performance.

As indicated in the previous section, many projects cannot adhere to predefined schedules or budgets because development teams fail to identify potential problems. These include failure to identify actual requirements in the early stages. In fact, many systems are first presented to end users or senior managers during the testing or even implementation stages. This results in the identification of case flaws and inappropriate functions in the latter stages of the project. The remedial work costs for flaws found in these latter stages are much higher (40 to 100 times) than they would be if identified in the early stages (Boehm and Turner, 2003). Extra time and costs are then needed to repair the inappropriate design. Project performance is also impaired when the project team fails to discover flaws and defects in the early stage. In addition, users will refuse to use the system if it fails to function as required.

One possible approach to avoid the above problem is to utilize users in the development process to ensure the developed product satisfies users' needs (He and King, 2008). Users should review the work completed by developers periodically so as to reduce unnecessary costs caused by inappropriate design. Furthermore, requirements may alter with the emergence of new technology and changes to the external environment. Users should also provide the most current information in order to counter uncertainties resulting from external environments (Hsu et al., 2008). Based on the above discussion, we predict that users engaged in the development process to review periodically the work done by developers can lessen the negative impact of inappropriate design. Moreover, their engagement provides greater assurance that the project will be accomplished on time and within budget. Thus, we hypothesize that:

H4. The magnitude of the impact of requirement determination effectiveness on project performance is influenced by the degree to which the user participates in the reviewing process.

3. Methodology

The research model is shown in Fig. 1. To examine the proposed model, data collected from practitioners by means of a two-step approach was used. The first step was to send a letter to all 359 institute members of the Information Management Association (IMA) in Taiwan. IMA is an organization that aims to improve IT usage and enhance communication among IS professionals. Almost every member of this organization is an IS department manager. Members who were willing to participate in our study were then contacted by telephone. In our phone conversation, we introduced the main purpose of the study and explained the data collection procedures. The number of projects completed recently in each member's organization was then recorded. In the second stage, we delivered the survey package to 750 project managers, team leaders, or senior members, their contact details having been collected from the previous stage. A total of 279 people returned the survey package, 19 of which were excluded from the ensuing analysis due to missing values. This yielded a valid response rate of 34.6%. Table 1 summarizes the demographic characteristics of the final sample.

3.1. Construct and measurement

All research variables were measured by using multi-item scales adopted from previous research studies. Some minor revisions were made before the survey was officially delivered. All the questions were in Likert scale format, using anchoring from 1 (strongly disagree) to 7 (strongly agree).

Requirement determination refers to the synthesis of the ISD members and users' knowledge to develop a comprehensive understanding of the requirements and to transfer the concept developed from that understanding into a system design. Requirement determination requires an interaction process to synthesize the knowledge from different stakeholders and then to create new knowledge or insight (Newell et al., 2004). A total of 8 items adopted from Carlile (2004) and Tiwana and McLean (2005) were used to capture the extent to which users and developers can share, transfer, and combine their own expertise to comprehend the system design.

User–IS relationship refers to the level of mutual trust, respect, reciprocity and closeness of relationship between users and developers during the ISD project. (Kale and Singh, 2000; Tiwana and McLean, 2005). A total of 5 items adopted from

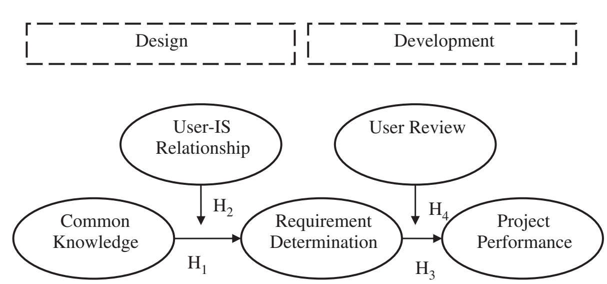

Fig. 1. Research model.

Table 1
Demographic information.

Measure	Categories	#	%	Measure	Categories	#	%
Tenure	Less than 4 years	53	20.4	Duration in project	Less than half–1 year	104	40.0
	4–10 years	129	49.6			81	31.3
	11–20 years	69	26.5		1–2 year	46	17.7
	More than 21 years	7	2.7		2–3 year	14	5.3
	Missing	2	0.8		More than 3 years	14	5.3
					Missing	1	0.4
Age	21–30	74	28.5	Gender	Male	184	70.8
	31–40	156	60.0		Female	74	28.5
	41–50	26	10.0		Missing	2	0.8
	More than 51	4	1.5				
Team size	< 5	101	38.8	Educational background	Less than college	12	4.7
	6–10	91	35.0		Bachelor	156	60.0
	11–20	53	20.4		Master	88	33.8
	21–30	7	2.7		Doctor	1	0.4
	More than 31	8	3.1		Missing	3	1.2
Position	Programmer	110	42.3	Industry type	Manufacturing	106	40.8
	SA	48	18.5		Service	48	18.5
	Project leader	45	17.3		Education	9	3.5
	CIO	22	8.4		Finance	21	8.1
	Other specialists	33	12.7		Others	42	16.0
	Missing	2	0.8		Missing	34	13.1

Tiwana and Mclean (2005) were used to measure the user–IS relationship.

Common knowledge includes *the IS developer's knowledge of the business domain* that relates to the developer's knowledge of the new application areas and *Users' knowledge of system development,* which measures the users' overall knowledge/ expertise in IS development methods and processes. A total of 4 items for developers' business knowledge and 6 items for users' ISD knowledge, adopted from Barki et al. (2001), and Tesch et al. (2009), were used to measure developers' familiarity with knowledge in the business domain and users' understanding of IT.

User review was assessed by asking whether users reviewed and approved developers' work in a certain time period. A total of four items from Hsu et al. (2008) were adopted to determine user review activities, which included formally approving work of the IS developers, formally reviewing work performed by IS developers, being informed of progress and/or problems, and signing a formalized agreement.

Project performance refers to the success of the development process itself (Wallace and Keil, 2004). It was measured using 5 items adopted from existing scales (Guinan et al., 1998; Henderson and Lee, 1992; Jones and Harrison, 1996) that assessed subjects' perceptions of project performance in terms of schedule, budget, and work quality. The details of each item are shown in Table 2.

In addition to variables listed in the model, we included two control variables critical for final performance in our research context. *Task uncertainty* refers to "the difference between the amount of information required to perform the task and the amount of information already possessed by the organization" (Galbraith, 1973). Uncertainties resulted from internal or external during which the system development process has been shown to directly influence the quality of the system (Hsu et al., 2008). A total of 3 items adopted from Rizzo et al. (1970) were used the capture the level of uncertainty. *System complexity* refers to the perceived level of complexity associated with the analysis and design of an information system (Tait and Vessey, 1988). System quality is negatively associated with the level of complexity because more problems related to the specifying of actual needs may arise during the development process. A total of 2 items adopted from Tait and Vessey (1988) were used to capture the complexity level of the target system.

3.1.1. Reliability and validity

SmartPLS (Ringle et al., 2005) was used to evaluate the measurement and structural models. A two-step procedure including measurement validation and path analysis was used for data analysis. The validation of measurement includes item reliability, convergent validity, and discriminant validity tests. As shown in Table 2, with the exception of one, all indicators in this study have loadings higher than 0.6. The minimum composite reliability is 0.82 for instrumentality, and the item-total correlations are all higher than 0.5. Table 3 shows the descriptive statistics and correlation matrix of aggregated data. For each variable, the mean, standard deviation, Skewness (M3), and Kurtosis (M4) are provided to represent the central and diversified dependency. The correlation matrix shows moderate (0.003 to 0.42) correlations among variables. The square root of the AVE is shown in the diagonal of the Correlation Matrix in Table 3 with all values exceeding the threshold of 0.707. As indicated in Table 3, the AVEs are greater than the inter-construct correlations. The results display strong construct reliability and validity.

3.1.2. Common method variance

Since we collected both independent and dependent variables simultaneously from the same respondent, common method variance (CMV) might be a concern in this study. The Harman's single factor test was implemented to ensure that there was no significant method effect on the predefined causal relationship. This approach assumes that more than one factor should be generated through a factor analysis process. The exploratory factor analysis shows that more than two factors can be derived, the first factor explaining 32.6% of variance. In addition, the impact of method variance was tested by creating one method variable (with all used indicators) and linking it to both independent and dependent variables (Pavlou and Gefen, 2005; Podsakoff et al., 2003). The impact of this method variable is not statistically significant, suggesting that the common method bias problem should not be problematic in this study. Following the approach proposed by Podsakoff et al. (2003), and Williams et al. (2003), we included a common method factor in the PLS model. The average substantively explained variance by principal indicators is 0.71 and the average variance explained by method is 0.01. The ratio of substantive variance to method variance is approximately 70:1

Table 2
Reliability and validity.

Constructs	Items		Factors	
			Loadings	ITC**
Developers' business knowledge *CR*=0.888, *Alpha*=0.832, *AVE*=0.665	1	The developers are knowledgeable about the key success factors that must go right if the company is to succeed.	0.82	0.67
	2	The developers understand the company's policies and plans.	0.80	0.64
	3	The developers are able to interpret business problems and develop appropriate technical solutions.	0.86	0.73
	4	The developers are knowledgeable about business functions.	0.78	0.60
Users' IT knowledge *CR*=0.912, *Alpha*=0.884, *AVE*=0.635	1	Users are familiar with IT.	0.74	0.63
	2	Users have a lot of experience in IS development.	0.85	0.78
	3	Users are familiar with this application.	0.79	0.68
	4	Users are familiar with the process of IS development.	0.86	0.78
	5	Users are familiar with their role in this project.	0.81	0.71
	6	Users are aware of the importance of their role in this project.	0.73	0.61
User–IS relationship *CR*=0.930, *Alpha*=0.905, *AVE*=0.727	1	There is close, personal interaction among developers and users	0.76	0.63
	2	There is mutual respect between developers and users	0.87	0.80
	3	There is mutual trust between developers and users	0.90	0.84
	4	There is personal friendship between developers and users	0.89	0.80
	5	There is high reciprocity among developers and users	0.84	0.74
Requirement determination *CR*=0.933, *Alpha*=0.918, *AVE*=0.637	1	Developers are able to transfer what users say into system design	0.80	0.78
	2	Users are able to describe requirements in the way that developers can understand it clearly	0.75	0.73
	3	Developers used the way that users can understand to help them to express their needs	0.76	0.77
	4	Developers and users are proficient at combining and exchanging ideas to solve problems in system development process	0.83	0.81
	5	Developers and users did a good job of sharing their individual ideas to come up with new systems (or functions)	0.79	0.69
	6	Developers and users are capable of sharing their expertise to bring new concepts into system	0.82	0.72
	7	Developers and users transfer their own knowledge to each other	0.86	0.65
	8	Developers and users build shared meaning toward each other's expertise (knowledge)	0.77	0.68
User review *CR*=0.914, *Alpha*=0.880, *AVE*=0.729	1	Users formally approved work done by the developers	0.91	0.79
	2	Users formally reviewed work done by developers	0.93	0.84
	3	Users were informed progress and/or problem	0.86	0.70
	4	Users signed off a formalized agreement	0.71	0.64
Project performance *CR*=0.896, *Alpha*=0.856, *AVE*=0.634	1	This ISD project meets predefined goals.	0.83	0.64
	2	In this ISD project, expected amount of work completed.	0.84	0.69
	3	In this ISD project, high quality of work completed	0.85	0.68
	4	In this ISD project, there is adherence to schedule.	0.79	0.73
	5	In this ISD project, there is adherence to budget.	0.66	0.59
Task uncertainty *CR*=0.872, *Alpha*=0.780, *AVE*=0.695	1	The sequence of activities required to accomplish my task is: easily identifiable/hardly identifiable	0.80	0.76
	2	The results of the activities in my task are: easy to predict/hard to predict	0.86	0.63
	3	Well-defined knowledge on which the accomplishment of my task can be based: exists does/not exist	0.84	0.70
System complexity *CR*=0.821, *Alpha*=0.812, *AVE*=0.613	1	It is difficult to determine the information requirements of the system.	0.97	0.90
	2	The complexity of the processing is high.	0.67	0.64
	3	The overall complexity of the system design is high.	0.67	0.64

**: ITC: Item-total correlation.

Table 3
Descriptive analysis and correlation matrix.

Variables	Mean	Std. dev.	M3	M4	Correlation matrix							
					DBK	UIK	UIR	KI	UR	PP	TU	SC
Developers' business knowledge	5.30	0.85	−0.42	−0.15	**0.82**							
Users' IT knowledge	4.52	1.02	−0.05	−0.23	0.36	**0.80**						
User–IS relationship	5.21	1.09	−0.55	−0.37	0.24	0.37	**0.85**					
Requirement determination	5.30	0.81	−0.38	−0.32	0.40	0.40	0.61	**0.80**				
User review	5.22	0.88	−0.14	−0.49	0.34	0.32	0.29	0.32	**0.85**			
Project performance	5.22	0.94	−0.42	−0.15	0.33	0.38	0.42	0.38	0.23	**0.80**		
Task uncertainty	5.24	0.90	−0.75	0.63	0.34	0.26	0.34	0.38	0.25	0.40	**0.83**	
System complexity	5.06	1.07	−0.68	0.68	0.08	0.08	−0.01	0.07	0.06	−0.05	−0.06	**0.78**

M3: Skewness; M4: Kurtosis.
The diagonal line of correlations matrix represents the square root of AVE.

which exceeds the minimum requirement. In addition, most method factor loadings are not significant. Based on the above findings, we believe that method bias is unlikely to be a serious concern in this study (Liang et al., 2007).

3.2. Results of hypotheses testing

All the path coefficients and explained variances for the model are shown in Fig. 2. As indicated, common knowledge ($\beta=0.48$, $p<0.001$) has a positive effect on requirement determination. This result confirms our expectation and provides support for H1. In addition, the path from requirement determination to project performance is also significant ($\beta=0.27$, $p<0.001$), which shows that H3 is supported.

To test H2 and H4, a moderated multiple regression analysis (MMR) was used. Carte and Russell (2003) urged that researchers should use MMR to test moderating effects and emphasize the difference in R-square, rather than the coefficient of the interaction term. A three-step approach was adopted to test the moderating effects using MMR with SmartPLS. First, requirement determination regression was regressed by common knowledge and the user–IS relationship, and model 0 in Table 4 of the model fit (R_0^2) was obtained. Second, the interaction term was added between common knowledge and the user–IS relationship for model 1, obtaining the new model fit (R_1^2) as model 1 in Table 4. Third, the difference of R2 of the first two steps ($\triangle R_0^2$) was calculated to obtain the f-value for the significance of the change. The same three-step approach was used to test H4. First, requirement determination regression was regressed by requirement determination and user review, and model 0 in Table 4 of the model fit (R_0^2) was obtained. Second, the interaction term was added between requirement determination and user review for model 1, which obtained the new model fit (R_1^2) as model 1 in Table 4. Third, the difference of R2 of the first two steps ($\triangle R_0^2$) was calculated to obtain the f-value for the significance of the change.

The result shows that the magnitude of the common knowledge impact on requirement determination is associated with the level of user–IS relationship. Therefore, H2 is supported. In contrast, the magnitude of the requirement determination impact on project performance is only mildly associated with the level of user review. Although the path coefficient is not significant but the effect size of interaction term is significant, we concluded that H4 is partially supported.

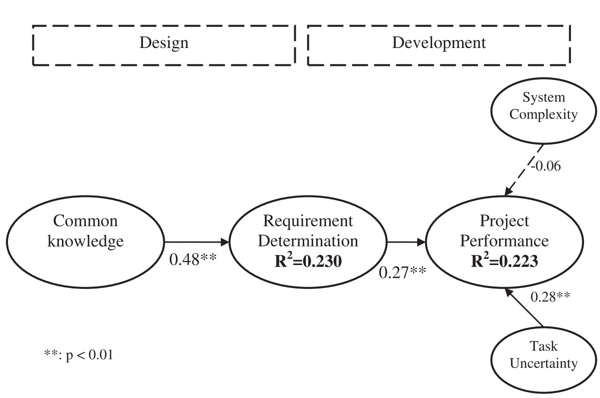

Fig. 2. Structural model and paths coefficient.

Table 4
Moderator effects.

Independent variable	Direct effect	Moderating effect	
	Model 0	Model 1	Model 2
H2: *The moderating effect of user–IS relationship*			
Common knowledge (CK)	0.48 **	0.29 **	0.26 **
User–Is relationship (UIR)		0.50 **	0.50 **
CK * UIR			−0.11 **
R^2	0.230	0.436	0.492
R^2 difference		0.206	0.056 **
H4: *The moderating effect of user review*			
Requirement determination(KI)	0.27 **	0.25 **	0.24 **
User review (UR)		0.08	0.06
KI * UR			−0.12
R^2	0.223	0.228	0.240
R^2 difference		0.005	0.012 *

* $p<0.05$.
** $p<0.01$.

4. Discussion

As we proposed, common knowledge has a positive impact on requirement determination. Common knowledge in this study is treated as a second-order formative construct which consists of the two components of users' IT knowledge and developers' business knowledge. As shown in Table 4, the user–IS relationship itself has a positive effect on requirement determination. Furthermore, the impact of the user–IS relationship exceeds the impact of common knowledge. The interaction between common knowledge and the user–IS relationship is found to have a negative effect on requirement determination. In order to better understand the moderating effect, a moderating diagram is presented. As indicated in Fig. 3, requirement determination is positively affected by the level of common knowledge and a significant difference is evident between high and low user–IS relationship groups. In addition, common knowledge and user–IS are substitutable. When common knowledge is low, requirement determination can still be high when relational capital is high. However, the difference between high and low user–IS relationship disappears when common knowledge is extremely high.

From requirement determination to project performance, the positive coefficient indicates that project performance can be enhanced through better requirement determination. In contrast, user participation has no effect on project performance. This means that having users in the development stage to review the work done by developers may not guarantee that the project can be accomplished within budget or on time. However, the interaction between user review and requirement determination positively affects project performance. It is noticeable that, as shown in Fig. 4, user review does not always have the expected effect. User review can promote project performance only when requirement determination is low. Although project performance is better when requirement determination is higher, when

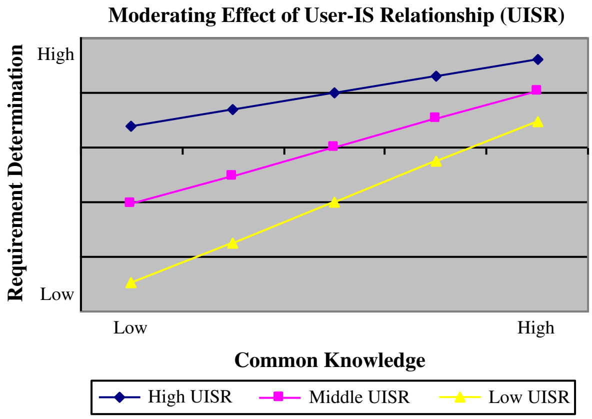

Fig. 3. Moderating effect of user–Is relationship.

requirement determination is extremely high, projects with a lower level of user review tend to perform better than those with more user review. A possible explanation for this observation is that including users in the development stage is not without cost. The reviewing process may take extra time and increase the cost of the development work since developers have to stop their work to attend the review meetings. In addition, conflicts or conflictions resulted from user review may delay the progress and harm the project performance. This is consistent with the user participation literature, which pointed out that users should be included only when needed (McKeen et al., 1994). Therefore, we can conclude that whether the user should be involved in the development process in helping review work done by developers depends on the extent to which the design process is able to truly capture the users' needs.

5. Conclusion

The purpose of this study is to understand how users may serve as knowledge co-producers in the design and development process. We proposed that in the design stage, better design quality can be obtained when users and developers possess knowledge about each other's domain. Furthermore, when common knowledge is inadequate, the relationship

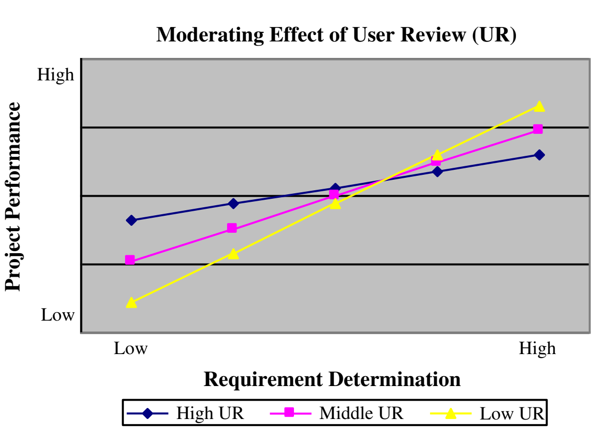

Fig. 4. Moderating effect of user review.

between developers and users plays an important role. In the development stage, users should participate in the review process to ensure that the integrated knowledge (system design) is carried out effectively by the developers. Users can help to detect or expose inappropriate designs as early as possible to reduce unnecessary costs. Data collected from 260 IS professionals support all proposed hypotheses. The results suggest that common knowledge has a positive impact on requirement determination, which results in better project performance. The impact of common knowledge on requirement determination is contingent on the user–IS relationship. Moreover, user review moderates the relationship between requirement determination and project performance.

This study generates several implications for academics and practitioners. For academics, firstly, we have successfully showed that users, as knowledge co-producers, can generate impacts on different stages of development. Traditional wisdom indicates that users should be included in the development process to help clarify actual requirements. This study further shows that users can better contribute their knowledge when they are able to communicate with developers effectively. A past study has pointed out the importance of common knowledge on project performance (Tesch et al., 2009). By including knowledge integration as mediator between common knowledge and project performance, we have extended this research stream by illustrating the impacts of common knowledge on the development process. Furthermore, as shown in Fig. 3, when common knowledge is low, the user–IS relationship can play a substitute role. When the relationship between users and developers is harmonious, requirement determination can still be achieved with low common knowledge.

Secondly, the significant and positive result suggests that project performance is a function of the extent to which users and developers can integrate their own knowledge to develop new knowledge. User review plays a role in this process. When requirement determination is low and system design cannot fully meet users' needs, user review serves as a tool for detecting and repairing inappropriate designs. Projects can be better accomplished within time and on budget if inappropriate designs are identified as early as possible. However, when knowledge from both parties is integrated and a high quality design is therefore achievable, user review should be ignored since the extra cost might impair project performance. This supports the traditional theory that users should engage in the development process only when needed.

For practitioners, since common knowledge is associated with the effectiveness of system design, promoting common knowledge becomes critical. The member recruitment process should take domain knowledge into consideration. Whether the design can reflect the real needs of users is determined by how familiar users and developers are with their counterparts' knowledge domains. In cases where experiential developers or users cannot be obtained, training should be provided to enhance developers' business knowledge and users' IT knowledge. Furthermore, as indicated in Fig. 3, the user–IS relationship is critical for requirement determination when common knowledge is low. This highlights the importance for developers of maintaining an adequate relationship with users because a good relationship leads to effective communication, through which better design can be assured. At the same time, it is critical for the user group to select appropriate representatives to participate in the design stage. Choosing those who have strong relationships with developers allows knowledge to be integrated. This may serve to generate a high quality system design even though common knowledge is low.

In the development stage, user review may enhance or undermine ISD project performance. Since user review does not always guarantee a better outcome, timing is critical. High complexity of the contemporary information system increases the difficulty in generating a comprehensive design in the initial stage. The dynamic nature of the external environment further drives the changes to design in the later stage of development. Managers should invite user representatives to attend the review meetings when managers realize or suspect that the initial design may not fully meet users' needs. However, early system design can better reflect the actual needs when the target system is simple or when developers have adequate experience in developing this type of system. In this case, having users in the development process may, in contrast, reduce efficiency. In those circumstances, user involvement in the development process is not recommended.

6. Limitations and suggestions for future study

The findings of this study should be explained with caution due to inherent limitations. First of all, the sample of this study is limited to the Information Management Association (IMA) in Taiwan. As indicated, as most of them are project managers or developers, their perspectives, to a certain degree, adequately reflect the opinions of industry in Taiwan. However, how well their perspectives reflect the IS professionals in the western culture may be questionable. Thus, the generalization of this study might be limited. In light of this, it is suggested that future research could use different samples to verify the framework presented in this study. Secondly, as the data was collected solely from developers, possible biases should be considered when interpreting the results of this study. Since previous studies have shown that users and developers may view success differently, it is suggested that future research collects data from both users and developers of ISD projects to obtain more precise and objective results. Finally, this study examined the role of users in the design and development stages of ISD projects. This study has identified the value for project management of drawing upon the assistance of users in two stages. Future research is encouraged to explore other factors which may be associated with the user–IS relationship and user review.

Acknowledgement

This paper (work) is partially supported by "Aim for the Top University Plan" of the National Sun Yat-sen University and Ministry of Education, Taiwan, R.O.C.

References

Barki, H., Rivard, S., Talbot, J., 2001. An integrative contingency model of software project risk management. Journal of Management Information Systems 17 (4), 37–69.

Bassellier, G., Benbasat, I., 2004. Business competence of information technology professionals: conceptual development and influence on IT-business partnerships. MIS Quarterly 28 (4), 673–694.

Bassellier, G., Reich, B.H., Benbasat, I., 2001. Information technology competence of business managers: a definition and research model. Journal of Management Information Systems 17 (4), 159–182.

Bassellier, G., Benbasat, I., Reich, B.H., 2003. The influence of business managers' IT competence on championing IT. Information Systems Research 14 (4), 317–336.

Bettencourt, L.A., Ostrom, A.L., Brown, S.W., Roundtree, R.I., 2002. Client co-production in knowledge-intensive business services. California Management Review 44 (4), 100–128.

Boehm, B., Turner, R., 2003. Using risk to balance agile and plan-driven methods. Computer 36 (6), 57–66.

Carlile, P.R., 2004. Transferring, translating, and transforming: an integrative framework for managing knowledge across boundaries. Organization Science 15 (5), 555–568.

Carte, T.A., Russell, C.J., 2003. In pursuit of moderation: nine common errors and their solutions. MIS Quarterly 27 (3), 479–501.

Coates, N., 2009. Co-creation: New Pathways to Value. LSE Enterprise www.promisecorp.com/newpathways.

Doherty, N.F., King, M., 2001. An investigation of the factors affecting the successful treatment of organisational issues in systems development projects. European Journal of Information Systems 10 (3), 147–160.

Galbraith, J.R., 1973. Designing Complex Organizations. Addison Wesley, Massachusetts.

Gemino, A., Reich, B.H., Sauer, C., 2007. A temporal model of information technology project performance. Journal of Management Information Systems 24 (3), 9–44.

Granovetter, M.S., 1973. The strength of weak ties. The American Journal of Sociology 78 (6), 1360–1380.

Grant, R.M., 1996. Toward a knowledge-based theory of the firm. Strategic Management Journal 17, 109–122.

Guinan, P.J., Cooprider, J.G., Faraj, S., 1998. Enabling software development team performance during requirements definition: a behavioral versus technical approach. Information Systems Research 9 (2), 101–125.

He, J., King, W.R., 2008. The role of user participation in information systems development: implications from a meta-analysis. Journal of Management Information Systems 25 (1), 301–331.

Henderson, J.C., Lee, S., 1992. Managing I/S design teams: a control theories perspective. Management Science 38 (6), 757–777.

Hsu, J.S.-C., Chan, C.-L., Liu, J.Y.-C., Chen, H.-G., 2008. The impacts of user review on software responsiveness: moderating requirements uncertainty. Information Management 45 (4), 203–210.

Hunton, J.E., Beeler, J.D., 1997. Effects of user participation in systems development: a longitudinal field experiment. MIS Quarterly 21 (4), 359–388.

Iacovou, C.L., Thompson, R.L., Smith, H.J., 2009. Selective status reporting in information systems projects: a dyadic-level investigation. MIS Quarterly 33 (4), 785–810.

Ives, B., Margrethe, H.O., 1984. User involvement and MIS success: a review of research. Management Science 30 (5), 586–603.

Jiang, J.J., Klein, G., Shepherd, M., 2001. The materiality of information system planning maturity to project performance. Journal of the Association for Information Systems 2 (1), 1–23.

Jones, M.C., Harrison, A.W., 1996. IS project team performance: an empirical assessment. Information Management 31 (2), 57–65.

Kale, P., Singh, H., 2000. Learning and protection of proprietary assets in strategic alliances: building relational capital. Strategic Management Journal 21 (3), 217–237.

Liang, H., Saraf, N., Hu, Q., Xue, Y., 2007. Assimilation of enterprise systems: the effect of institutional pressures and the mediating role of top management. MIS Quarterly 31 (1), 59–87.

Markus, M.L., Mao, J.-Y., 2004. Participation in development and implementation — updating an old, tired concept for today's IS contexts. Journal of the Association for Information Systems 5 (11/12), 514–544.

Martinsuo, M., Kantolahti, T., 2009. Knowledge integration between the change program and the parent organisation. International Journal of Knowledge Management Studies 3 (3/4), 241–258.

McKeen, J.D., Guimaraes, T., Wetherbe, J.C., 1994. The relationship between user participation and user satisfaction: an investigation of four contingency factors. MIS Quarterly 18 (4), 427–451.

Mitchell, R., Nicholas, S., 2006. Knowledge creation in groups: the value of cognitive diversity, transactive memory and open-mindedness norms. Electronic Journal of Knowledge Management 4 (1), 67–74.

Mohammed, S., Dumville, B.C., 2001. Team mental models in a team knowledge framework: expanding theory and measurement across disciplinary boundaries. Journal of Organizational Behavior 22 (2), 89–106.

Newell, S., Tansley, C., Huang, J., 2004. Social capital and knowledge integration in an ERP project team: the importance of bridging and bonding. British Journal of Management 15 (S1), 43–57.

Nidumolu, S., 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.

Nissen, M.E., Jennex, M.E., 2005. Knowledge as a multidimensional concept: a call for action. International Journal of Knowledge Management 1 (3), i–v.

Patnayakuni, R., Rai, A., Tiwana, A., 2007. Systems development process improvement: a knowledge integration perspective. IEEE Transactions on Engineering Management 54 (2), 286–300.

Pavlou, P.A., Gefen, D., 2005. Psychological contract violation in online marketplaces: antecedents, consequences, and moderating role. Information Systems Research 16 (4), 372–399.

Payne, A.F., Storbacka, K., Frow, P., 2008. Managing the co-creation of value. Journal of the Academy of Marketing Science 36 (1), 83–96.

Podsakoff, P.M., MacKenzie, S.B., Jeong-Yeon, L., Podsakoff, N.P., 2003. Common method biases in behavioral research: a critical review of the literature and recommended remedies. The Journal of Applied Psychology 88 (5), 879–903.

Preston, D.S., Karahanna, E., 2009. Antecedents of IS strategic alignment: a nomological network. Information Systems Research 20 (2), 159–179.

Procaccino, J.D., Verner, J.M., 2009. Software developers' views of end-users and project success. Communications of the ACM 52 (5), 113–116.

Ramasubbu, N., Mithas, S., Krishnan, M.S., Kemerer, C.F., 2008. Work dispersion, process-based learning, and offshore software development performance. MIS Quarterly 32 (2), 437–458.

Ravichandran, T., Rai, A., 2000. Quality management in systems development: an organizational system perspective. MIS Quarterly 24 (3), 381–415.

Ringle, C.M., Wende, S., Will, A., 2005. SmartPLS 2.0 (M3), Hamburg, Germany.

Rizzo, J.R., House, R.J., Lirtzman, S.I., 1970. Role conflict and ambiguity in complex organizations. Administrative Science Quarterly 15 (2), 150–163.

Robert, J.L.P., Dennis, A.R., Ahuja, M.K., 2008. Social capital and knowledge integration in digitally enabled teams. Information Systems Research 19 (3), 314–334.

Robillard, P.N., 1999. The role of knowledge in software development. Communications of the ACM 42 (1), 87–92.

Ross, J.W., Beath, C.M., Goodhue, D.L., 1998. Develop long-term competitiveness through IT assets. IEEE Engineering Management Review 26 (2), 37–47.

Schwalbe, K., 2002. Information Technology Project Management. Course Technology, Boston.

Stuart, F.I., Tax, S., 2004. Toward an integrative approach to designing service experiences: lessons learned from the theatre. Journal of Operations Management 22 (6), 609–627.

Swanson, E.B., 1974. Management information systems: appreciation and involvement. Management Science 21 (2), 178–188.

Szulanski, G., 1996. Exploring internal stickiness: impediments to the transfer of best practice within the firm. Strategic Management Journal 17, 27–43.

Tait, P., Vessey, I., 1988. The effect of user involvement on system success: a contingency approach. MIS Quarterly 12 (1), 91–108.

Tesch, D., Sobol, M.G., Klein, G., Jiang, J.J., 2009. User and developer common knowledge: effect on the success of information system development projects. International Journal of Project Management 27 (7), 657–664.

Tiwana, A., McLean, E.R., 2005. Expertise integration and creativity in information systems development. Journal of Management Information Systems 22 (1), 13–43.

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

Williams, L.J., Edwards, J.R., Vendengerg, R.J., 2003. Recent advances in causal modeling methods for organizational and management research. Journal of Management 29 (6), 903–936.

Yetton, P., Martin, A., Sharma, R., Johnston, K., 2000. A model of information systems development project performance. Information Systems Journal 10 (4), 263–289.