Questionnaire-Based Risk Assessment Scheme for Japanese Offshore Software Outsourcing

Hiroshi Tsuji[1], Akito Sakurai[2], Ken'ichi Yoshida[3], Amrit Tiwana[4], and Ashley Bush[5]

[1] Osaka Prefecture University, Graduate School of Engineering,
1-1 Gakuencho, Nakaku, Sakai, 559-8531 Japan
tsuji@cs.osakafu-u.ac.jp
[2] Keio University, Graduate School of Science and Engineering
4-1-1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa 223-8521, Japan
sakurai@ae.keio.ac.jp
[3] Tsukuba University,
3- 29-1, Otsuka, Bunkyo, Tokyo 112-0012, Japan
yoshida@gssm.otsuka.tsukuba.ac.jp
[4] Iowa State University, Postfach 10 52 80,
2340 Gerdin Business Building Ames, Iowa 50011-1350, USA
tiwana@iastate.edu
[5] Florida State University,
Tallahassee, FL 32306-1110, USA
abush@garnet.acns.fsu.edu

Abstract. As the volumes of software development increase and the cost reduction is required, most Japanese IT companies are interested in offshore software outsourcing. Although a lot of engineers have experienced the success and failure on their projects, their know-how still remains as tacit knowledge. This paper proposes a risk assessment scheme for new projects by externalizing such tacit knowledge. Such a scheme requires collaboration between industry and academia because the tacit knowledge is scattered over many companies and cannot be formalized by a single company or academic institute. Defining fourteen attributes related to software development and designing questionnaire about project evaluation, this paper clarifies how to quantify the risk of offshore software outsourcing. Risk assessment tool based on the proposed scheme will promote a knowledge spiral for project management.

1 Introduction

As the volumes of software development increases, most Japanese companies are interested in offshore outsourcing [9]. Their expected benefits in offshore outsourcing include flexible human resource procurement, cost reduction and an improved ability to meet short deadline [2, 5, 6, 7, 10, 11, 13, 14]. There are many excellent Java programmers in India and their salaries are generally lower than those of ordinary Japanese ordinal programmers. Furthermore, the shorter development time afforded by the extra man power can save a company from losing a market opportunity.

B. Meyer and M. Joseph (Eds.): SEAFOOD 2007, LNCS 4716, pp. 114–127, 2007.

However, there are still risks in offshore outsourcing: miscommunication, cultural difference in business customs, quality issues, and so on [2, 6, 7]. Although the offshore outsourcing is not a new concept and there are many experienced managers, such risks management know-how still remains as tacit knowledge [10, 11]. Then it is not easy to transfer such tacit knowledge from a project manager to others [8]. While nominal risk items are known, their magnitudes have not been measured. Note that some risks have a trade-off relationship. If one tries to avoid one kind of risk, one may increase other kinds.

Therefore, it is important to measure risks and analyze their relationship to the success or failure of offshore software outsourcing. Such a task cannot be completed by only academic people because they do not have real project experiences. And it cannot be done by only a corporate people either because they are reluctant to disclose their confidential experiences. The only possible way to do it is through an academia-industry collaboration.

This paper presents an overview of how we are promoting a project of the Joint Forum of Strategic Software Research in Japan. First, showing the black box in software development, section 2 presents our motivation, research framework and research steps. This project involves five academic people and thirteen industrial people (from Toshiba, Hitachi, Fujitsu, IBM-J and Mitsubishi). Next, section 3 introduces company visits for pre-analysis. The pre-analysis included structured interviews with a protocol and voting for the likelihood of successful outsourcing on virtual projects with nine attributes. Reviewing the pre-analysis, section 4 describes our design for a new questionnaire for externalizing tacit knowledge from skilled managers. The responses to the questionnaire were analyzed by three methods in section 5. Detailed discussions on the analysis are given elsewhere individually [14, 15]. Section 6 discusses what we did in the context of knowledge spiral.

2 Framework for Risk Assessment

Taking an engineering approach to risk assessment, we regard software development as a function with input, output and control parameters [11] as illustrated in Fig. 1. Our motivation is to clarify the causal relationship among input, output and control parameters. Once the structure of the function is identified, the output of new software development can be estimated by assigning values for the input and control parameters.

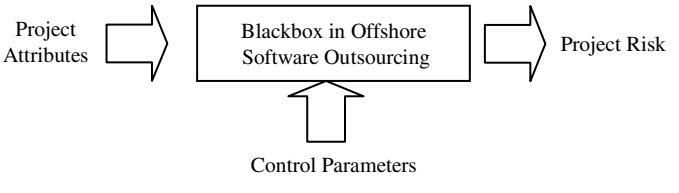

Fig. 1. Causal Relationship between Risk and Project

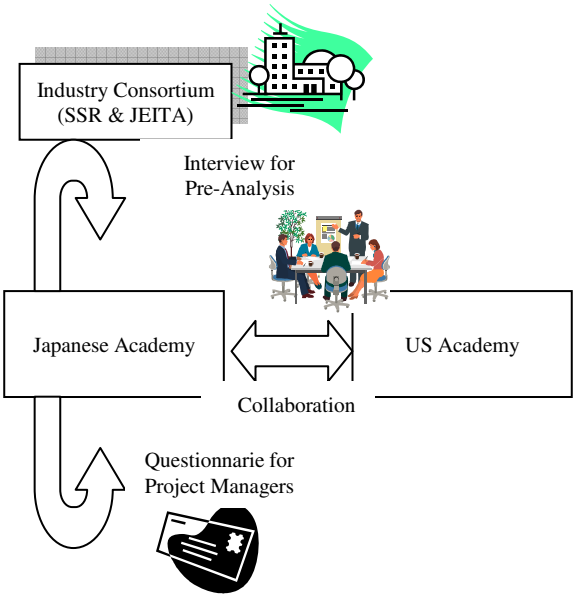

Fig. 2. Framework for Offshore Software Outsourcing Research

Table 1. Interview Protocol in Company Visit

Parts	Contents	notes
Project Background	- Vendor Type - Expected and actual man-month, budget and duration - Stage vendor become involved - Benefit - Satisfaction on outcome	Capital Relationship Both for client and vendor Design, Coding, Test, etc. Cost, vendor expertise, etc. Effectiveness and quality
Project Control	- Project decision - Outcome setting - Change Management -Contract Detail	Platform, Development methodology, etc Mile stones, budget, etc. Co-location, Weekly meeting US style vs. JP style
Collaboration	- Business custom - Domain Knowledge projection - Communication quality maintenance	Trigger, flexibility Intellectual propriety Bridge SE
Conclusion	Three top considerations	Success and failure

To identify the structure of the function, we established the framework for a research project as shown in Fig. 2. This framework is an instance of a joint forum called SSR (Strategic Software Research) in IISF (International Information Science

Foundation). SSR sends requests for proposal (RFPs) to Japanese academic institutes every year. The requirements in an RFP include: 1) the participants should involve industry people, and 2) the theme should be related to an international matter.

To promote the research, Japanese researchers collaborate with US researchers. The basic idea in this questionnaire scheme is borrowed from the original work of a US researcher [11]. To design the questionnaire and ensure a high response rate to it, we asked the industry members of JEITA as well as SSR to collaborate with us. JEITA (Japan Electronics and Information Technology Industries Association) has a committee on future human resource for software development and has interest in our work.

The research steps can be summarized as follows: 1) company visit for pre-analysis during one week, 2) repetitive face-to-face meetings and electronic meetings for questionnaire design during three months, 3) questionnaire delivery and collection during two months and 4) statistical analysis during two months. These steps are overviewed in the following sections.

3 Pre-analysis by Visiting Skilled Project Managers

Under the premise that the questionnaire is clue for risk assessment, one of our missions is to design a questionnaire. Since there already was a questionnaire for US project managers, we wanted to check whether it is applicable for Japanese project managers.

3.1 Interview Analysis

To overview the risk factors and find out the difference between Japanese offshore and America offshore outsourcing if any, we visited five Japanese client companies and two vendor companies (One Chinese and the other Indian) in October, 2005. The analysis was done as follows:

1) Each client company was visited by three or more of the authors who met at least two engineers there.
2) Each interview ran for thirty minutes. All questions and answers including translation were recorded using a voice recorder.
3) Each interview followed the structured protocol shown in Table 1.
4) To avoid hearsay and to obtain specific comments, the interviewee was asked to remind the last project and to describe it.

Then the followings are found while the interpretation of the reasons for the difference between Japan and American cases will be discussed elsewhere [13]:

1) There are three categories of attributes for offshore software development: software, vendor and project properties,
2) Each attribute seems to have a preference value for outsourcing and is not negligible in the decision to choose outsourcing,
3) Each company has a different strategy, especially for project control, for dealing with vendors.

3.2 Feasibility on Votes for Projects Evaluation

The questionnaire designed in the previous research makes use of three theories as shown in Table 2: transaction cost theory, agency theory and knowledge-based theory. We performed conjoint analysis on nine attributes based on these theories where each attribute has the value HIGH or LOW [11].

For the feasibility test, we asked skilled engineers to vote on the likelihood of success in the project using nine attributes where the range of vote was from 1 to 9. The results show the differences between Japanese and American outsourcing. This will be discussed elsewhere. It worked well for sophisticated professionals but still had some problems for various types of engineers.

1) It is difficult to imagine a project with nine attributes. Some participants in the experiments claimed that they could consider at most five attributes at once,
2) While HIGH means a positive value for outsourcing and LOW means the opposite from the view of a designer, these expressions confused interviewees somewhat,
3) A nine-point range was too wide for vote when there were twelve projects to evaluate.

Table 2. Project Attributes for Pre-Analysis

Category	Attributes
Transaction Cost Theory	Relative cost advantage
	Threat of opportunism
	Project complexity
	Project strategic importance
Agency Theory	Project outcomes measurability
	Vendor behavior observability
Knowledge Based Theory	Client technical knowledge
	Requirements knowledge specifiability
	Requirements volatility

4 Questionnaire Design

Based on the pre-analysis, we choose three property types for describing software development as shown in Table 3 instead of the previous nine attributes: software property with four attributes, vendor property with five attributes and project property with five attributes. The questionnaire has four parts and was designed so that a responder could answer all items in thirty minutes.

4.1 Part 1: Control Parameters

This part requires the personal information from the responder. These parameters are designed so as to adjust the bias of answers:

1) Numbers of years of IT experience, number of years of experience in the current company, and number of offshore projects experienced,
2) Position/ role: planner, project manager, project member,
3) Standard for evaluating vendors: ISO or CMM [4] ratings,
4) Type of projects: customer application, middleware or embedded software,
5) Vendor countries: China, India, Vietnam or others.

Table 3. Attributes of Offshore Software Development in Questionnaire

	Category	Attributes	Two levels for Attribute	
S1	Software Property	Software complexity and scale	Simple and small	Complex/ large
S2		Software quality measurablity	Easy to measure	Difficult to measure
S3		Requirement specifiablity	Easy to specify	Difficult to specify
S4		Requirement volatility	No change	Shall change
V1	Vendor Property	Communication skill	Good	Bad
V2		Project management capability	Much reliable	Unreliable
V3		Vendor flexibility on specification changes	flexible	Not flexible
V4		Attrition rate	Smal rate	Large rate
V5		Long term strategy	Yes	No
P1	Project Property	Deadline urgency	Urgent	Not urgent
P2		Relative cost advantage	High advantage	Low advantage
P3		Client side technical expertise	Lack	Suffcient expertise
P4		Strategic importance for future project	High	Low
P5		Ability to monitor vendor behaviour	Easy to monitor	Difficult to monitor

4.2 Part 2: Separate Evaluation on Attribute Importance

This part is designed to verify whether a responder knows the weight of each attribute in the outsourcing decision. The question is "According to your experience and knowledge, how important is each attribute in Table 3? Assume that you are the person in charge even if you should actually follow the decision made by a top manager". There are five options for the answer, ranging from 1 (negligible) to 5 (Very important).

4.3 Part 3: Evaluation of an Experienced Project

This part is designed to reveal the relative weight among software, vendor and project properties by evaluating fourteen attributes at once. While it was difficult for a

responder to imagine a virtual project described by nine attributes in the pre-analysis, we suppose that it would not be difficult to imagine all attributes of a experienced project that they had actually experienced.

The question is "Think of one recently outsourced software development project. Keeping this project in mind, please evaluate its result in terms of fourteen attributes." Each attribute has two possible values as shown in Table 3. The development result is assigned by a value ranging from 1(fatal failure) to 5 (success beyond expectation)].

4.4 Part 4: Evaluation on Virtual Projects

This part is designed to identify the importance of attributes in the separate properties by conjoint analysis [1] [14]. Because the pre-analysis showed that it was difficult to image nine attributes at once, we classify fourteen attributes into four software attributes, five vendor attributes and five project attributes. Based on orthogonal planning of conjoint analysis, we prepare three sets of virtual projects.

An example question for the vendor property is "You will be presented with a series of 9 virtual vendor profiles in Table. Based on this information and your own experience and knowledge, please circle the appropriate numbers in the following table. How attractive would it be for your company to OUTSOURCE to this vendor?" The similar questions are provided on software property and project property, too. The evaluation for profile is assigned by value ranging from 1 (low possibility for success) to 5 (high possibility for success) .

5 Overview of Risk Extraction by Statistical Method

There are two approaches to sampling: random sampling and intentional sampling. In general random sampling does not include bias, but the return rate may be terrible because the contents requested by the questionnaire are too confidential for responders to disclose. Therefore, we use two channels for questionnaire delivery as mentioned before: SSR and JEITA. Each company in SSR collected twenty responses and JEITA collected thirty responses. There were other volunteers who answered the questionnaire. In total, we collected one hundred and seventy five responses. They are all Japanese client-side people.

5.1 Frequency Analysis

The first Analysis is a simple frequency analysis, Figure 3 shows the distribution of software category and Figure 4 shows that of vendor countries. Note that about sixty percent of outsourced software to foreign countries is customer applications and fifty percent are outsourced to china.

Table 4 shows attribute importance by separate evaluation for Part 2 questions described in 4.2. It shows that there is little difference in importance among attributes for any property. In a sense, this confirms that separate evaluation has no meaning in determining risk magnitude.

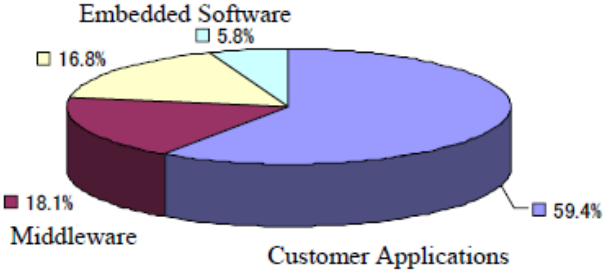

Fig. 3. Rate of Software Categories

Table 4. Attributes Importance by Separate Evaluation

Score in Software	Software complexity and scale	Quality measurability	Requirement Specifiability	Requirement volatility
5	59	38	53	54
4	49	67	67	50
3	34	32	24	34
2	10	14	8	13
1	4	5	4	5
Average	3.96	3.76	4.01	3.87
Rage	0.25	0.24	0.26	0.25

Score in Vendor	Communication skill	Project management capability	Vendor flexibility	Attribution rate	Long term strategy
5	90	82	47	23	48
4	56	53	60	68	51
3	9	20	40	49	47
2	1	1	8	13	6
1	0	0	1	3	3
Average	4.51	4.38	3.92	3.61	3.87
Rage	0.22	0.22	0.19	0.18	0.19

Score in Project	Deadline urgency	Relative cost advantage	Client side technical expeitise	Strategic impoitance for future project	Ability to monitor vender behavior
5	66	64	51	18	35
4	36	56	68	54	64
3	47	29	29	60	50
2	4	6	6	17	7
1	3	1	2	7	0
Average	4.01	4.13	4.03	3.38	3.81
Rage	0.21	0.21	0.21	0.17	0.2

5.2 SEM Analysis

This analysis is done for the experienced projects collected by Part 3 of questionnaire. The main concern is to determine the degree of importance among three property types: software, vendor and project properties. Introducing four latent variables

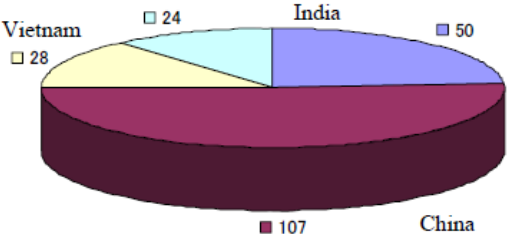

Fig. 4. Rate of Vendor Countries

(software, vendor, project and satisfaction) that are not observed in answers from responders, we refine path diagrams step by step. The modifications and findings are discussed in detail elsewhere [15].

The final model is shown in Fig. 5. The findings are summarized as follows:

1) Vendor property such as communication ability and project management ability mainly affected the result of development,
2) Software property such as requirements specificity and requirements volatility did not affect the result directly but did affect it indirectly through project property such as relative cost advantage and project strategic importance,
3) Control parameters such as vendor companies and software type did not improve the precision of the models.

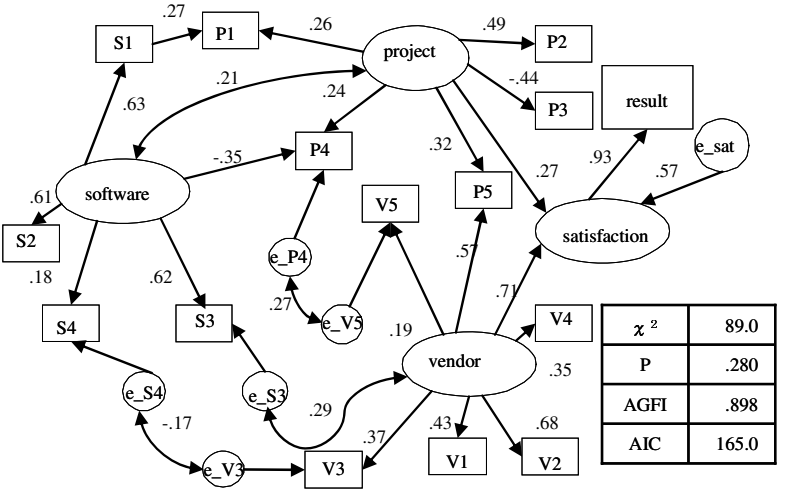

Fig. 5. Result of Structural Equation Modeling

5.3 Conjoint Analysis

As introduced in the description of Part 4 of the questionnaire, this analysis is done for the assessment of virtual projects generated by orthogonal planning [1]. There are three sets for virtual projects. One set is described in software attributes defined in

Table 3, second set is described in the vendor attributes and the final set is described in the project attributes.

The main concern is to detect the relative importance of attributes in their properties. Conjoint analysis [1] calculates the relative importance rate and partial utility of attributes. The sum of partial utility decides the range of total utility. The total utility is an estimated value for project evaluation.

$$TotalUtility = \sum PartialUtility + Const$$

Table 5 shows an example result of a conjoint analysis result where the samples are classified by software category: customer application, middleware and embedded software. From this table, we obtained followings:

1) There are different risk magnitudes for attributes in each category. For example, in outsourcing embedded software, the attrition rate cannot be used to evaluate a vendor. Instead, communication skill and project management capability are the key attributes for selecting vendors for this category,
2) Because the sum of partial utilities and a constant for middleware is smaller than those for the other two categories, the success beyond expectation is unlikely to occur in the middleware category.
3) According to Pearson's R and Kendall's taw, the fitness of the model is excellent for any category.

Let us show another example result of conjoint analysis results. Fig. 6 shows the individual partial utilities in software property: software complexity and scale, and requirement volatility.

Table 5. An Example Result of Conjoint Analysis

():Sample size

	Customer Applications (92)		Middleware (28)		Embedded Software (26)	
Vendor Property	Importance Rate	Partial Utility	Importance Rate	Partial Utility	Importance Rate	Partial Utility
Communication skill	25.02	.3638	24.05	.2404	28.26	.4459
Project management capability	19.95	.2992	21.44	.3173	24.21	.3986
Vendor flexibility on specification changes	18.61	.2795	18.2	.2788	18.72	.2703
Attrition rate	16.74	.2093	19.83	.2500	9.89	.0946
Long term strategy	19.68	.1896	16.48	.1731	18.92	.1959
Sum of Partial Utilities	1.3414		1.2596		1.4053	
Constant	2.4480		2.3654		2.500	
Pearson's R	.983		.994		.997	
Kendall's tau	1.000		.982		1.000	

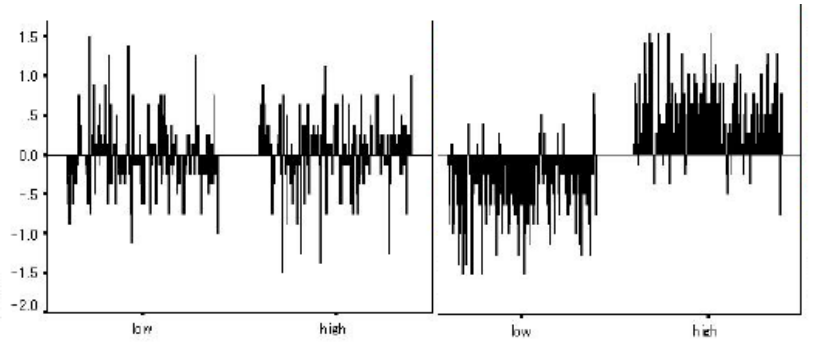

Fig. 6. Examples of Partial Utility in Software Property

Each bar expresses a responder's utility value. An upward bar expresses negative feeling for offshore outsourcing because there must be risk while a downward bar expresses positive feeling. Most people agree that the potential requirement volatility is risk. On the other hand, there are two different views of whether or not software complexity and scale is a risk.

To confirm the fitness of the model for vendor property, let us depict the frequency diagram as shown in Fig 7. The x-axis is the estimated total utility based on the previous formula and the y axis is the occurrence count. There are five lines and each expresses the same result of a project where the range is from1 to 5.

6 Discussion

Let us discuss what we did. Again, our basic assumption is that the experienced managers know risk factors and their magnitudes as their tacit knowledge. Asking them to answer the designed questionnaire forced them to externalize their knowledge. Then the written knowledge can be shared with other people. However, it is difficult to use such written knowledge separately because it is too subjective.

There is a hint in the SECI model proposed by Nonaka [8] where SECI means socialization, externalization, connection and internalization. To connect individual items of externalized knowledge, we have used statistical analysis methods like SEM and conjoint analysis. The connected knowledge based on statistical analysis can be shared as discussed in the previous section.

Furthermore, there should be internalization for knowledge transfer. Internalization allows persons to learn connected knowledge and increase their tacit knowledge. In our case, risk assessment for new project corresponds to the internalization. The model of knowledge spiral is shown in Fig. 8. Thus, we have chance to design and to install risk assessment tool.

Assigning values to the attributes of a new project, IT manager has chance to get diagnosis result. The risk assessment tool refers to the relative importance among

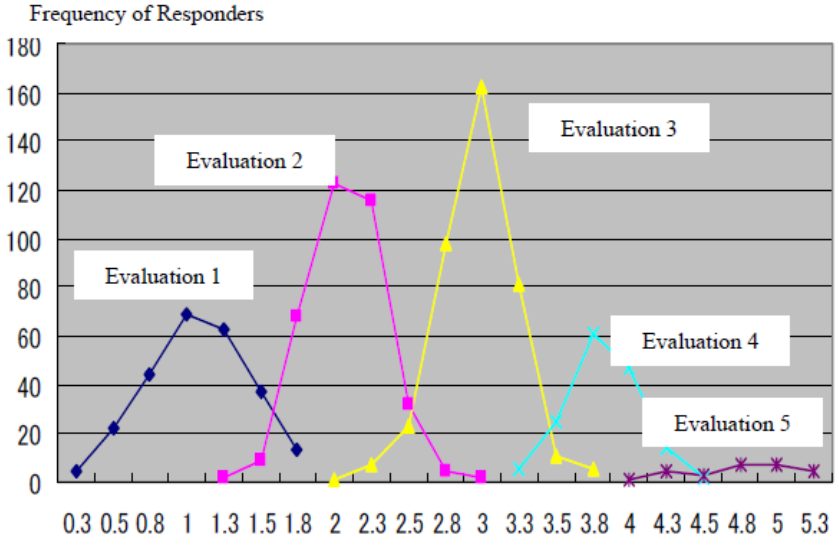

Fig. 7. Freuuency of Estimated Preference by Vendor Property

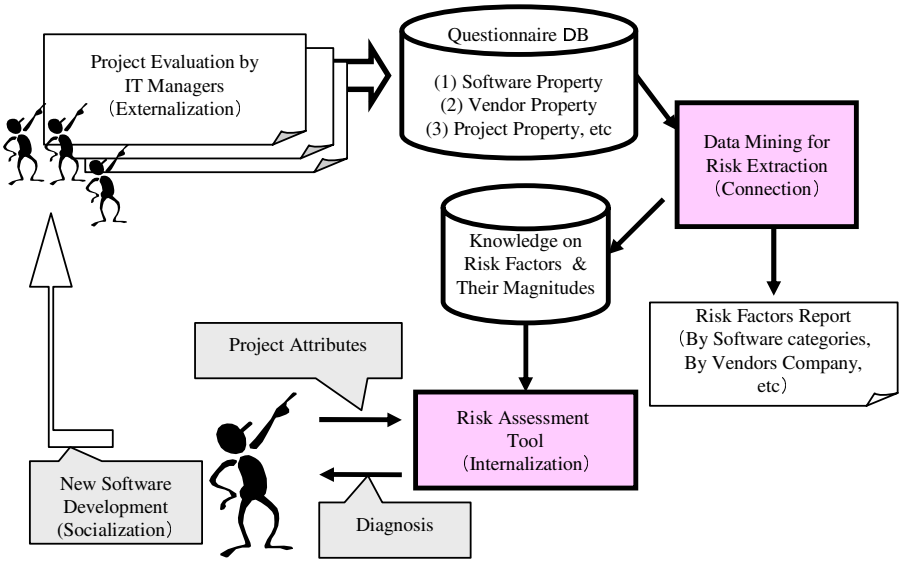

Fig. 8. Knowledge Spiral in Offshore Software Development

three properties discussed in 5.2 and the partial utilities values for attributes discussed in 5.3. The basic idea for tool itself is borrowed from [12] and the example screen is shown in Fig. 9.

Persons who internalize the experience in the past projects will collaborate with other people and outsource new project to a vendor. In SECI model, this collaboration

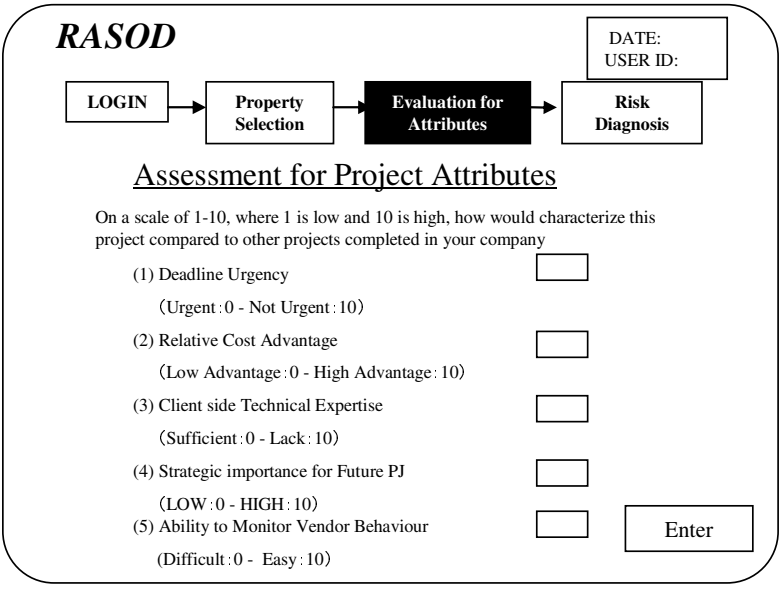

Fig. 9. An Example Screen for Offshore Software Outsourcing Assessment Tool

is socialization. Thus, their tacit knowledge will propagate among their brain. Then their externalization in the future should be new. This will lead to a knowledge spiral in offshore software outsourcing.

7 Conclusion

This paper has presented a questionnaire-based risk assessment scheme. Our contributions are as follows:

- We established an academia-industry collaborative framework for research on offshore software outsourcing.
- To approach the issues from an engineering viewpoint, we presented how to collect sample data from experienced managers and how to identify risks and their magnitudes.
- By visiting companies for pre-analysis and by testing its feasibility on collecting reasonable responses, we have designed a four-part questionnaire.
- Delivering the designed questionnaire through two intentional channels, we collected one hundred and seventy five responses.
- The collected samples were analyzed by three statistical methods: frequency analysis, structural equation modeling and conjoint analysis. This paper has shown an example of the statistical analysis results and what they reveal.
- This paper also showed that our research plays the role of a knowledge spiral in the context of the SECI model. The development of the risk assessment tool is on going.

Acknowledgements

The authors would like to sincere thanks to members of Strategic Software Research forum (SSR) and Japan Electronics and Information Technology Industries Association (JEITA) who contributed to collect responses of questionnaire. Special thanks are also due to Mr. Yoshihisa WADA and Mr. Daiki Nakahigashi who helped our statistical analysis and developing risk assessment tool.

References

1. Andrews, R.L., Ansari, A., Currim, I.S.: Hierarchical Bayes Versus Finite Mixture Conjoint Analysis Models: A Comparison of Fit, Prediction, and Partworth Recovery. Journal of Marketing Research XXXIX, 87–98 (2002)
2. Aspray, W., Mayadas, F., Vardi, M.Y. (eds.): Globalization and Offshoring of Software, Report of the ACM Job Migration Task Force, Association for Computing Machinery (2006)
3. Bharadwaj, A., Tiwana, A.: Managerial Assessments of E-Business Investment Opportunities: A Field Study. IEEE Trans. on Engineering Management 52(4), 449–460 (2005)
4. Carnegie Mellon University: The Capability Maturity Model: Guidelines for Improving the Software Process, Addison Wesley Longman, Inc. (1994)
5. Gold, T.: Outsourcing Software Development Offshore - Making It Work. Auerbach Publications (2005)
6. Krishna, S., Sahay, S., Walsham, G.: Managing Cross-Cultural Issues in Global Software Outsourcing. CACM 47(4), 62–66 (2004)
7. Mayer, B.: The Unspoken Revolution in Software Engineering, Computer, pp. 121–124 (January 2005)
8. Nonaka, I., Takeuchi, H.: The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation. Oxford Univ. Pr., Oxford (1995)
9. Software Business Comittee: Report on Software Development Resource, Japan Electronics and Information Technology Industries Association, No. 06-P-9 (2006)
10. Thondavadi, N., Albert, G.: Offshore Outsourcing - Path to New Efficiencies in IT and Business Processes, 1st Books Library (2004)
11. Tiwana, A.: Beyond the Black Box: Knowledge Overlaps in Software Outsourcing. IEEE SOFTWARE 21(5), 51–58 (2004)
12. Tiwana, A., Keil, M.: The One-Minute Risk Assessment Tool. CACM 47(11), 73–77 (2004)
13. Tiwana, A.B., Tsuji, H., Sakurai, A., Yoshida, K.: Myths and Paradoxes in Japanese IT Outsourcing.Communications of the ACM (forthcoming)
14. Tsuji, H., Sakurai, A., Yoshida, K., Tiwana, A., Bush, A.: Risk Factors on Offshore Software Development by Conjoint Analysis (In Japanese). Transaction on Information Processing Society for Japanese 48(2), 823–831 (2007)
15. Wada, Y.D.N., Tsuji, H.: An Evaluation Method for Offshore Software Development by Structural Equation Modeling. In: Proc. of the First Software Engineering Approaches For Offshore and Outsourced Development, vol. 4716, pp. 114–127 (2007)