www.calsec.or.kr
http://dx.doi.org/10.7838/jsebs.2013.18.1.165

지속비교 방법에 관한 연구 : 초심 모델러 교육으로부터의 교훈

Study on the Use of the Constant Comparison Method : Lessons from Training Novice Modelers

김태경(Taekyung Kim)*, 박진수(Jinsoo Park)**, 노상규(Sangkyu Rho)***

초 록

개념적 모델링은 성공적인 비즈니스 정보 시스템을 개발하기 위한 중요한 활동이다. 본 연구의 목적은 개념적 모델링 활동에 있어서 근거연구로부터 도입한 지속비교 방법을 적용하는 것에 관한 가능성을 평가하는 것이다. 이를 알아보기 위해, 우리는 초심 모델러를 훈련시키고 두 그룹으로 나누어 사후 평가를 실시했다. 실험 결과는 지속비교 방법을 훈련받고 이를 적용한 그룹이 그렇지 않은 그룹보다 경험이 많은 모델러에게 더 호의적인 평가를 받는다는 점을 보여주었다. 더욱이, 통제 그룹은 도메인 지식에 대해 덜 친숙할수록 문제해결에 어려움을 겪었지만, 실험 그룹은 어려움에 더 잘 대처했다. 또한 지속비교 방법의 적용은 개념적 모델링의 분석 시간을 단축시켰다.

ABSTRACT

Conceptual modeling is a critical activity for developing successful business information systems. The objective of this study is to evaluate the possibility of applying the constant comparison method from the grounded theory to conceptual modeling. To achieve the objective, we trained novice modelers and split them into two groups for evaluation. The experimental results show that applying the constant comparison method could increase acceptability from more experienced conceptual modelers. Moreover, while the control group was experienced difficulties when domain knowledge is unfamiliar, the experimental group could handle difficulties more effectively. In addition, applying the constant comparison method also decreased the time to complete analysis for conceptual modeling.

키워드 : 개념적 모델링, 지속비교 방법, 초심 모델러 교육
Conceptual Modeling, Constant Comparison Method, Training Novice Modelers

---

This study was supported by the Institute of Management Research at Seoul National University.
* College of Business Administration Seoul National University
** Corresponding Author, Graduate School of Business Seoul National University
(E-mail : jinsoo@snu.ac.kr)
*** Graduate School of Business Seoul National University
2012년 11월 12일 접수, 2012년 11월 21일 심사완료 후 2012년 11월 27일 게재확정.

1. Introduction

In this paper, we examine the impact of applying conducting constant comparison principles to conceptual modeling. Traditionally, conceptual modeling has been regarded as a crucial step for information system design. Conceptual modeling identifies what data is important and what data should be maintained. In addition, the activity helps establish a common ground on which users and developers can communicate to one another about desirable functions.

Information systems may include sets of concepts people use to organize knowledge about domains. Those concepts are manifested as entities or classes in information system development, and appear in information technologies such as databases and software applications. Therefore, searching for better ways to discover those elements can help improve performance of conceptual modeling [17]. Thus, properly identifying and stating a process of obtaining articulated concepts from the domain is critically important to the success of conceptual modeling for IS projects, and hence of equally vital concern to both clients and system designers.

Empirically, we have seen conceptual modeling performance is highly associated with modeler's experiences. The more time and resources they spend on a specific domain, the more cognitive clues they have for discovering valuable information. Experiences can make modelers accumulate useful patterns that are helpful in reducing information overload for analyzing domain knowledge. Nevertheless, information systems scholars have given little attention to accumulating procedural knowledge on how to improve the performance of modeler's cognitive process. What kinds of principles should be kept? How can IS institutes train modelers to ready for handling difficulties in understanding domain knowledge? More importantly, what does happen if we try to add a procedural method to a conceptual modeling task?

Answering to the questions may be complicated. It may be true if a conceptual modeler has lots of experiences in a specific domain, she may be comfortable to understand users' requirements, thus modeling them more successfully. But, for a modeler who is lack of experiences may be confused when she has to deal with unfamiliar domains. Sometimes, even an experienced modeler should observe a familiar domain carefully because knowledge is constantly changing.

Based on the research motivation, we try to answer to the question: "Does the constant comparison method can increase performance of conceptual modeling?" Originated from grounded theory research methodology, the constant comparison method aims to discover theories directly from empirical data usually reported by people situated in unfamiliar context by extracting concepts and relationships from the transcripts. By constantly, recursively comparing work-in-progress results, the researcher can narrow down the focus to more general and articulated concepts and the associations. The use of grounded theory methodology is a journey to understand facts and experiences that are difficult to be unveiled unless the researcher commits. Figuratively, grounded theory research is to develop abstract models based on data reported by people who have specific knowledge. In addition, its purpose is to enhance knowledge sharing between peer researchers and practitioners by presenting theories. The research methodology has great implications for IS researchers since conceptual modeling has similar features. In order to develop a conceptual model, we sometimes need to collect data reported by people, extract concepts, and draw diagrams to discuss with further implementation.

We believe that the constant comparison method from the grounded theory can provide useful insight for conceptual modelers in scrutinizing user requirement and articulating concepts. Inspecting things completely and comparing the current result to prior ones may be trivial; however, those activities are not mandatory for conceptual modeling. In order to test our propositions, we worked with novice data modelers who had not been exposed to any system development experiences including conceptual modeling. After training, we randomly split them into two separated groups, and exclusively introduced grounded theory methodology to one group. To capture differences, the laboratory experiment and interviews were conducted. As a result, we learned that applying constant comparison can increase modeling performance potentially. But the study also revealed pitfalls.

2. Background

2.1 Conceptual Modeling

Conceptual modeling refers to formally describing some aspects of the physical and social world around us for purpose of understanding and communication [16]. This activity targets to establish an unambiguous, consistent, and complete specification for developing information systems based on knowledge in the universe of discourse and strategic business requirements. Kung and Sølvberg [14] added that conceptual modeling can be viewed as the mutual activity of knowledge discovery between a modeler and a client since conceptual models are commonly used (1) to facilitate communications between people, (2) to support the analysts' understanding of the domain, (3) to serve as the basis for design and implementation of information systems, and (4) to record design rationales.

To understand domain knowledge, various kinds of modeling methods are used by information system designers [1]. Indeed, conceptual modeling can be helpful in reducing noises and errors since the designers become aware of necessary domain concepts, functions and processes. Even if standardized modeling techniques are not adopted, practitioners frequently use homemade modeling concepts or mix multiple modeling languages to achieve the goals of conceptual modeling [1]. We may say that conceptual modeling is quite common for system designers to obtain domain-specific knowledge.

It seems to be obvious that conceptual modeling is crucial for developing systems. Some modeling standards, such as entity-relationship diagrams and class diagrams, are deceptively simple to learn; therefore, we can easily overlook how it difficult to map domain knowledge into grammatical representations. Interestingly, some empirical studies have demonstrated that conceptual modeling is actually a challenge for some engineers since user requirements are insufficiently decoded. It is worth to note that a hidden failure of initial stage in system development can be often fatal [4]. Conceptual modeling as one of incubating activities in the initial stage should be carefully managed.

Unfortunately, human perception can be different from person to person, and conceptual modeling involves a cognitive activity to perceive core concepts and necessary interactions for completing tasks. Indeed, it is quite common in conceptual modeling to produce different models even if almost the same sources of knowledge and modeling tools are presumed [20]. One side of conceptual modeling is to use a toolkit which contains ontologically verified grammars, and the other side is to use cognitive power to discover concepts, properties and interactions in order to fit those into the given modeling language [24]. The source of difficulties can be found in (1) human constraints on information processing, (2) the variety and complexity of information requirements, (3) communication issues between analysts and users, and (4) the unwillingness of users to provide requirements [7].

2.2 Constant Comparison Method

Social scientists have conducted research with investigating symbolic meanings from data generated by social interactions. The scholars collect interview data to discover hidden and general categories for explaining social events. By doing so repeatedly, they expect that emerging theories would be established rooted in data thus fitting into true understanding about human being and institutions. Therefore, testing applicability of the method from qualitative social research may be helpful in finding a better pedagogical solution to the current problem.

Among various alternatives, constant comparative features of grounded theory methodology attract our attention. Grounded theory methodology and its procedures are now cited as the most influential and widely adopted modes of conducting qualitative research when developing emerging theories in the various field of medicine, sociology and organization science to name a few [23]. According to Glaser and Strauss [10], the inventors of grounded theory methodology, comparing facts to know whether or not they are delivering similar or different meanings can help scholars generate useful properties of categories for generalizing theories.

Since the methodology aims to explain social interactions hidden in human activities, substantive theories as the result cannot be interchangeably interpreted as conceptual models for information systems. While a researcher develops theories using the methodology, she actually needs to be treated as a tool for interpreting things; in other words, previously learned ontological foundations on how to perceive the global structure of focal events strongly affect the way of imposing theoretical associations [6]. Therefore, the same data can produce different theoretical explanations due to the differences of perceptual readiness or theoretical sensitivity of researchers [22].

The constant comparison method is a research strategy for developing grounded theories. It is rather simple, but has a quite well organized procedure as follows. First, a research reads through a give text source that is a developed from transcribing interview data. Next, the researcher highlights a part of data, which is named an incident. At the same time, she tries to think about properties that describe the incident. Namely, the pair of coding and analyzing an incident is a basic unit of constant comparison. For example, let us assume we have statements, "I had an accident in 1989 - fell backwards in the stairs. Well, over twenty years, my pain never stops. When it was cold, that was, that was killing me and my wife." The first code name may be chronic pain. By highlighting the first and the second sentence in the example, we can learn he has suffered from an injury. The second code may be traumatic damage. From the last sentence, we infer that his wife also needs a care because she has to endure watching his pain. The reason we choose words, chronic and traumatic, to describe incidents is that those are indicating adequate properties for capturing meanings.

According to Glaser and Strauss [10], "The purpose of the constant comparative method of joint coding and analysis is to generate theory more systematically" [10, p. 102]. In detail, they suggest guidelines for conducting the constant comparative method. First, a researcher starts by developing codes with writing memos about them. By comparing them to existing categories of codes developed so far, she can discover emerging codes and note differences. Secondly, as the coding continues, the existing units of joint coding and analysis can be changed since the researcher learns more about the relating incidents. In this case, properties need to be compared more deeply so that the researcher can understand relations between categories consisting of codes. Next, the researcher can develop an emerging theory by delimiting overlapping results. The reduction of categories and relations aims to formulating more general explanation rather than just eliminating duplication. Finally, the researcher produces a substantive theory that fits into the given information source. Based on the result, she targets the next information source to analyze. If the previous substantive theories exist, the current theory needs to be compared with them. Open coding, axial coding and selective coding can used to ensure the constant comparative method in sociological domains [21]. Open coding is to inquire the meaning of each incident. In axial coding, the result of open coding is merged and mapped into a predefined perceptual model, for example the paradigm model [21]. A substantive theory obtained from the axial coding can be enhanced by conducting selective coding. In sum, the constant comparative method used by grounded theorists has a systematic approach from discovering concept names, properties and relations to developing theoretical models. Conceptually, borrowing the idea for developing conceptual models seems to be reasonable and harmless.

Notwithstanding, following the constant comparison method of grounded theory methodology can be overwhelming. Pidgeon et al. [18] tested applicability of grounded theory methodology to requirement engineering. They learned the idea could be appealing since transcribed interview data could be decoded effectively. However, using grounded theory methodology produced too many things to be handled. First of all, system designers had to learn the research methodology that was originally designed for social scientists. It may an unintended direction to train the designers to be researchers. Secondly, too many unnecessary codes were produced. If we have a fixed ontological representation to the world, we can conveniently map results of open coding to the axial coding. Moreover, if we have fixed words and grammar, the number of properties for describing codes can decrease. Pidgeon et al. [18]'s study alludes that an absence of modeling assumptions may yield too many things that do not contribute to develop conceptual models. In turn, that may reduce performance and cost a lot.

Although applying grounded theory methodology to conceptual modeling seems to be a good idea, it is unsure whether or not the result is actually positive. Moreover, we know little about how to train novice data modelers when a method of constant comparison is required. If grounded theory methodology is helpful in overcoming difficulties from the lack of experiences, we can develop further pedagogical materials based on the result to increase readiness for a system designer.

3. Research Model

3.1 Hypothesis

Training conceptual modeling skills need to involve a method of how to understand domain knowledge. If we fail to educate candidates for system designers about how to handle domain knowledge during conceptual modeling, they cannot help but to learn by doing. Previous studies on differences between experienced modelers and novice ones clearly show that we should find a way to get ready for cognitive problems in conceptual modeling. Experienced modelers have better capabilities in terms of understanding domain-specific knowledge, structuring problems and dealing with cognitive difficulties [19]. Novice modelers are apt to avoid doing in-depth examination thus resulting in insufficient distinction among concepts [5]. Schenk et al. [19] reveal that novice modelers gain fewer domain-specific concepts than experienced ones and over-emphasize general issues instead of functional requirements and focused information issues. Moreover, the experienced modelers took a different approach in the empirical study. While the novices adopted a top-down approach to understand complex events, the experienced modelers focused on a more bottom-up approach to problem solving. Schenk et al. [19] add that lacking proper knowledge organization lead to limited and stereotypical understanding on information sources. More experienced modelers try to cope with unfamiliar domain knowledge before drawing diagrams [3]. The above-mentioned findings indicate that novice modelers require learning procedural methods to deal with difficulties relating to unfamiliar domain knowledge.

Understanding unfamiliar things may be difficult for anyone. Familiarity is one's understanding of an entity that is perceived as having association to other known entities. The concept can be applied to cases of what, who, how and when of what is happening [9]. It is noted that familiarity may reduce the uncertainty of personal expectation through increased understanding of what has occurred in the past. Usually, greater familiarity reduces uncertainly about how another person will behave in the future [11]. Although it is uncertain whether or not the effect of familiarity on conceptual modeling is positive, we intuitively know that more familiar subjects are easier to learn and summarize.

Since the constant comparison method can provides a systematic way of understanding written requirement statements, we can expect that even a novice modeler can perform better conceptual modeling tasks. Especially we posit that unfamiliar domain knowledge can be understood effectively; therefore, a conceptual model produced by the novice modeler can be acceptable by peer expert modelers. Based on the argument, the following hypothesis is established :

Hypothesis : If a novice modeler is trained to use the constant comparison method, s/he can produce better conceptual models that can be acceptable by peer experts in case on that even an unfamiliar task is given.

3.2 Preparation

We recruited twenty participants from an undergraduate business school who had no experience in conceptual modeling. In class, they were taught basic concepts of management information systems including database theories and the applications. The subjects were suitable for our research since they only learned basic concepts and never had conducted conceptual modeling.

Since the participants had no prior experience in conceptual modeling, one of authors taught them how to create entity-relationship models. We strictly followed the textbook written by Elmasri and Navathe [8]. Basic concepts about entity-relation modeling and relational constraints in chapters 3, 4, and 5 were taught. To attract their attention, a sample project to develop a database application was also presented using Microsoft Access 2007 and Excel 2007. Since the project was not a hand-on practice, we interactively asked the participants whether or not they understood the procedure of the project.

The training session continued for about two months, totally eighteen hours. The reason we did not push them was to minimize pressure on peer competition. The participants had overcome intensive competition to enter the college positioning the top of national university rankings. Additionally, we observed that a test was easily perceived as a compelling challenge to prove one's value; therefore, each participant was apt to spend extra time to study, which was obviously harmful to control subject's experiences.

The total sample size diminished to ten, which was half of the original participants. Five persons refused to take part in the experiment because of personal reasons. Two persons were overqualified to solve the sample problems used to check conceptual modeling skills because they spent extra time to study other references. To the contrary, three persons were not sufficiently trained. Finally, eight males and two females completed the experimental tasks with randomly dividing into the control group and the experimental group. The divided groups were tested to validate that there were no differences in modeling skills. The test was conducted by using several quizzes including items on basic notations about entity-relationship modeling and relational database. In addition, the participants were asked to solve a simple exercise appeared in [8]. We discovered no differences between the groups.

The sample size is not sufficient to draw a rigorous conclusion in terms of statistical inference. However, we could more deeply understand the effects of applying constant comparison since we had spent time enough to observe actual behavior and have chances to ask questions individually. Moreover, educating people with the qualitative methodology required active interactions. If we consider the important of ensuring everyone in the experimental group to understand constant comparison, too many samples can reversely diminish reliability. Thus, the sample size was not determined by time and resource constraints, but by the extent of control-ability and sincerity.

3.3 Training

For the experimental group, two case studies from Strauss and Corbin [23] were provided as background information about grounded theory research. In addition, short hands-on materials for practicing open coding and axial coding were introduced. Those were borrowed from the department of nursing; therefore, the context of them was nothing to do with system development or conceptual modeling relating to experimental tasks. The two hours training was highly interactive. We had group discussions and spontaneous question and answer (Q&A) sessions to understand constant comparison methods of grounded theory methodology. The small sample size was actually beneficial because an intimate environment where everybody felt free to speak up could not be created if the size had been larger than we could control.

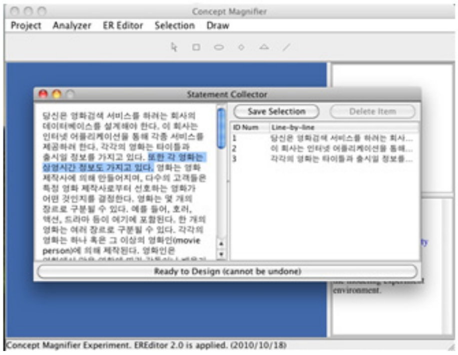
3.4 Task

Two different tasks were developed for our test. In order to overcome a language barrier, all the materials were written by the domestic language for the participants of our experiment. The outline of tasks was extracted from the textbook written by Elmasri and Navathe [8, pp. 99-100].

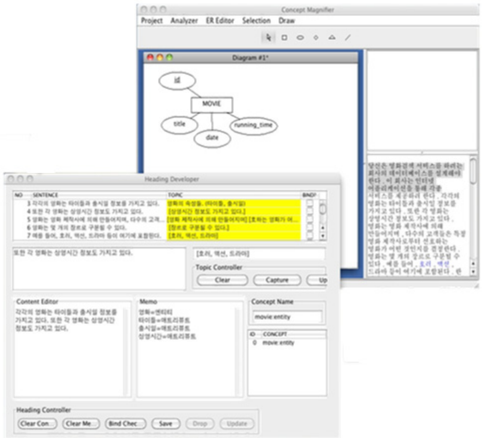
In detail, the first task was about a movie domain. It was described in a similar way of short examples that were taught in class. For example, attributes were followed by the "has" clause. Identifiers and relationships could be easily identified because of patterns of repeating clauses. Background information was described in a separated paragraph. The second task was about ordering automobile parts. It was rather unfamiliar to the most of subjects in our test. In order to achieving correspondence between the task and our assumption on unfamiliarity, we directly asked participants about the main topic of the given task and the extent of familiarity. We found that all the subjects felt that the first task was much familiar than the second one.

3.5 Procedure

To become a conceptual modeler, it does not require learning a qualitative research methodology, including grounded theory. We need to note that even professional researchers usually have to spend years learning a qualitative research methodology. For a conceptual modeler, it may be difficult to follow steps suggested by the constant comparison method unless a supplementary tool is provided. Therefore, we developed a tool to support



Screen Capture of READING



Screen Capture of MEMOING and ERD

〈Figure 1〉 Screen Capture of the Tool for the Experiment

subjects who participated in the experiment as shown in <Figure 1>.

For convenience, we named the tool, Concept Magnifier (CM). The program has three modules as follows: READING, MEMOING and ERD. The program module, READING, has two purposes: first, the module is needed to make sure every sentence read at least once, and second, it helps the user properly index sentences. CM pops up the Sentence Collector window after importing a text file. In the experiment, all instructions were printed out via the terminal console. CM shows that the user should read all sentences to proceed, and the result of the collection would be used as a source for further analysis and design. It is natural for a good conceptual model to reflect requirements thoroughly. That is also true for the grounded theory researchers. In the view of grounded theory methodology, it is crucial to inspect sentences line by line using the open coding method for understanding meanings of concepts and relationships. Reading sentences should be accompanied by reminding previous experiences and general knowledge on the focal phenomenon. The step evokes memories of prior experiences of similar topics thereby increasing performance of subsequent activities of analysis and design.

The second model is MEMOING which contains three sub-modules: Heading Developer (HD), Content Editor (CE) and Concept Developer (CD). In turn, HD shows sentences and topics. CE is to comment sentences sharing the same topic. CD contains names of derived concepts. First, a user has to discover a main topic for each sentence. If there are more than two sentences sharing the same topic, the user can bind them and leave a comment about difference and similarities. If necessary, the user can use CD to record names of concepts that are perceived to be important. Sentences, topics, contents and concepts are all integrated and constantly compared. The user should refer the panel shown in the session MEMOING until she assures that the conceptual modeling is finished.

The final module, ERD is a drawing tool. A user can open a new window for drawing diagrams during the MEMOING session. In CM, the entity relationship diagrams are provided, such as entity, attribute, relationship, and generalization. ERD can save diagrams as pictures; therefore, the user can easily compare alternative representations.

CM controls the overall modeling process to keep the constant comparison method strictly. The following description shows detail information about how CM manages the process:

1. Let the use read a text material.
2. Let the user define topics.
3. Let the user discover similar topics by binding information.
4. Let the user collect emerging names of concepts during the analysis.
5. Let the user draw models.
6. Repeat the step 2 through 5 until a final conceptual model is obtained.

3.6 Manipulation

The control group only knew entity-relationship modeling technique, whereas the experimental group was additionally taught about the grounded theory methodology. Before the experiment, we asked the participants in the experimental group to try to follow instructions suggested by the application in the experiment. On the other hand, the control group could use any methods they preferred, including pen and pencil, and/or notebook and desktop word-processor. Therefore, in given duration, the experimental group was forced to use the Concept Magnifier, and the control group freely used any analysis technique.

3.7 Data Collection

If the suggested idea in this paper positively affects novice modelers, the quality of conceptual models from the experimental group should be better than ones from the control group. The issue is how to judge the comparative quality. In order to score each model, we recruited three evaluators from a website development company. About twenty US dollars were paid per hour for compensation. They had three to five years of experiences in programming and developing databases.

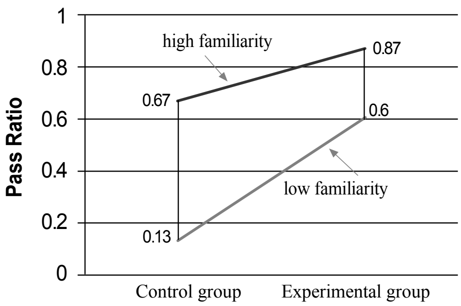
The evaluation process is as follows: (1) all evaluators are informed about the dimensions of conceptual modeling quality as a general guideline [15]; (2) the evaluators have time to discuss with each other about the characteristics of an acceptable conceptual models for each task; (3) two document envelopes are provided (titled “better” and “worse,” respectively), and (4) the evaluators individually classify conceptual models and put them into one of the envelopes. We put different values for each envelope—positive one for the better label and negative one for the worse label. If an evaluator failed to classify, the model get zero. We conducted interviews based on the evaluation. By inviting a subject individually, we asked unstructured questions in order to understand the modeling process and personal experience.

4. Result

4.1 Effect of the Constant Comparison Method

We calculated inter-rater reliability between any two evaluators. Traditionally, Cohen’s Kappa or Scott’s Pi has been used to evaluate reliability of human judgment. According to Gwet [13], those ratios can be a problem when the extent of agreement between evaluators is too high. To avoid obtaining biased reliability, we adopted the AC_1 reliability measure [12, 13]. Overall, the average of AC_1 shows positive signs. For the control groups, the value was 0.62, and the experimental group had the value of 0.67. We learned that one evaluator considered that the subjects were novices so that he had more generous attitude comparing to the others. Since the values were well over 0.6, we concluded that the evaluations were acceptable [2].

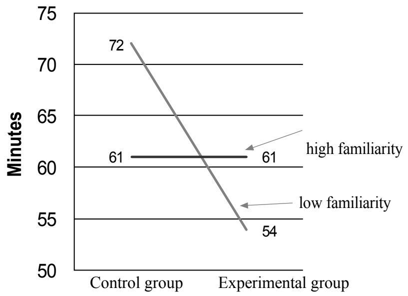
The result of evaluation can be summarized by a ratio between zero and one. We term the range as Pass Ratio. If a value of Pass Ratio is zero, it means a subject in the group is surely assigned to the envelope labeled “worse.” On the other hand, if the ratio reaches one, the subject in the group is surely assigned to the envelope labeled “better.” In <Figure 2>, the values of Pass Ratio are shown. While the y-axis represents a range of Pass Ratio in numeric values, the x-axis contains two nominal values for separating the control group and the experimental group respectively.



〈Figure 2〉 Pass Ratio

First, we can see there are two solid lines differentiated by the extent of familiarity. The upper line indicates the case of high familiarity, and the lower line is the case of low familiarity. Both cases have positive slopes; namely, overall the experimental group received better evaluation scores. However, the subjects in the control group did not take positive evaluation when they produce conceptual models for the case of low familiarity. Meanwhile, the experimental group took more positive results.

From the interview with evaluators, we found that conceptual models of novices had many errors and misleading names. Regardless of group types, the outputs needed to be modified and clarified. One of the evaluators complained that the subjects did not think about implementation. Nevertheless, evaluators put positive values in large part because they assumed that conceptual modeling might be repetitive in nature. During the discussion, we additionally heard that the conceptual models provided by the control group were fragmented and even worse useless. The harsh review focused on the case of low familiarity. One evaluator felt that a subject too strictly followed natural language forms consisting norms and verbs. He indicated that a requirement statement could be unorganized and sometimes contained unnecessary information. The models from the control groups



〈Figure 3〉 Time to Complete

failed to filter those background data. In comparison, the experimental group made fewer mistakes.

Besides, the experimental group reduced time of completion. In the case of low familiarity, the average of completion time was about 61 and 72 minutes respectively for the control group and the experimental group as shown in <Figure 3>. However, about 54 minutes were consumed for the experimental group to submit completed models. Still, the control group spent over 60 minutes. Interestingly, we were told during the interview with the experimental group about that they understood which entities and attributes were required, but they could not be sure of relationships. Admittedly, we did not provide intense drills of handling with various types of relationships. The interviewees asked us how to represent relationships among entities very specifically. When we asked the reason why they spent much time in the case of high familiarity, all the interviewees commonly reported that writing memos took longer time. However, the memos were useful in understand the case of low familiarity. Two persons stated that memos were used to discover some patterns between the tasks. As we designed, those tasks shared common structures. Some of the experimental group found the fact; however, none of the control group did state the commonality. Analyzing a written text following by constant comparison methods consumed substantial time;

but, the working-in-progress results boosted the overall productivity due to discovering of reusable patterns. Intuitively, difficult tasks may take longer time to complete. In our case, the experiment group consumed less. As we mentioned above, two tasks used in the experiment shared common features. If a subject found those, s/he could reuse them. That might result in less time to complete although the given task was unfamiliar.

Grounded theories who are originally using constant comparison methods have discussed patterns between categories will emerge as more informants provide empirical data. We learned that the same effect could be found in conceptual modeling with constant comparison. Overall, we concluded that the hypothesis was supported from the result.

4.2 Experiences

<Table 1> shows the summary of modeling activities. In average, the subjects in the control group produced more entities, attributes and relationships than the experimental group, and, in turn, they modified names or deleted objects more frequently.

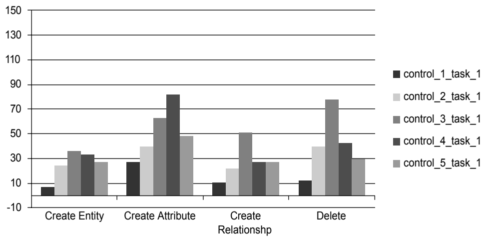
〈Table 1〉 Summary of Modeling Activities

Group	Activity Classifier	Familiarity	
		High	Low
Control	Editing	594	529
	Entity Creation	25	26
	Attribute Creation	52	72
	Relationship Creation	25	36
	Delete	41	52
Experimental	Editing	275	391
	Entity Creation	20	19
	Attribute Creation	45	67
	Relationship Creation	20	27
	Delete	25	31

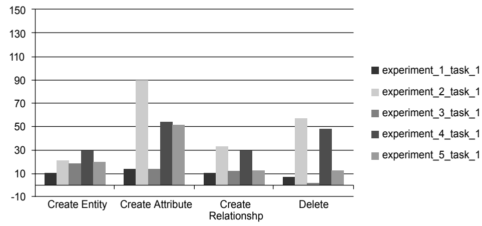
In detail, from <Figure 4> to <Figure 7>, individual records are shown. As we can see, the experimental group more conservatively drew diagrams. Especially, the code name experiment_1 and the experiment_3 were abstained from creating and deleting objects. In the case of high familiarity, the control_1 and the control_5 were similar to those in the experimental group in terms of activity patterns. However, they did differently when the low familiarity task was given.

Those variations we captured from the modeling processes are analyzed by interviewing subjects. We found that the subjects in the control group partially used ER diagrams for the purpose of understanding the requirement statements whereas the persons who did constant comparison partially produced diagrams as examples. The differences may come from the treatment setting in the experiment. The analytic procedure we provided might separate context analysis and modeling sequentially; thus, the subjects in the experimental group had much information on major entities, attributes and relationships. Meanwhile, the subjects in the control group confessed that they tried to convert any norms and verbs they found in the requirement statement first, and, in turn, they wanted to review interim results quickly. If the work-in-progress did not convince the subject, the many parts of the model were removed or modified. The problem became worse in the unfamiliar task since the subjects in the control group were easily anchored by the previous result of conceptual modeling. From the interview with the control_1 and the control_5, we learned that they tried to recall how they had drawn conceptual models by trying to reproduce models. However, the results made them disappointed because they in fact failed to connect two tasks due to the literal complexity of the unfamiliar task. On the other hand, four persons from the experimental group actively used the previous and current analysis results with comparison. Although they also put many diagrams, the purpose was to evaluate their understanding rather than discovering hidden patterns. Moreover, the interviewees recalled how they unveiled knowledge from the requirements, and actively discussed on weaknesses of their results. From the interview, we found that the experimental group acquired much information and experience by conducting constant comparison.

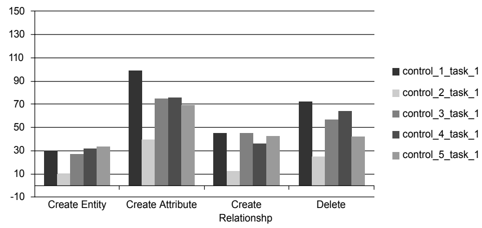
From the interviews based on the result of the experiment, we learned that applying constant comparison to analyzing requirement statements and drawing ER diagrams would be helpful for novice designers in improving the quality of the modeling process. However, it should be noted that interviewees



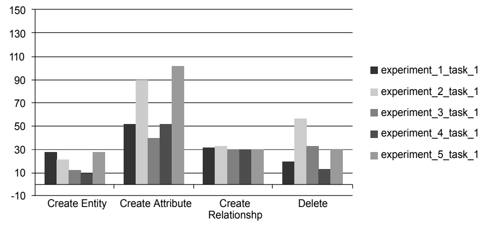
〈Figure 4〉 Activities of the Control Group with the Familiar Task



〈Figure 5〉 Activities of the Experimental Group with the Familiar Task



〈Figure 6〉 Activities of the Control Group with the Unfamiliar Task



〈Figure 7〉 Activities of the Experimental Group with the Unfamiliar Task also reported pitfalls. They were confused about linking the constant comparative methods and the process of conceptual modeling. The subject in the experiment group who took the lowest points from the evaluators complained that examining a requirement statement with discovering concepts and writing memos required too much effort to follow up. Moreover, he argued that the procedure of iteratively comparing work-in-progresses might be less valuable than just drawing diagrams since the final model would not need the whole information produced during the analytic work. In addition, two subjects told us that they were very confused about how to translate the result of constant comparison into diagrams. They felt that examining text line-by-line was helpful in understand content, but they were skeptical about how ER diagrams would be obtained from the verbatim descriptions. When we asked the subjects of the experimental group whether or not they would use the suggested methods, four out of five conditionally accepted the method. They commonly said that efforts to produce written descriptions should decrease if the method would be more valuable.

4.3 Discussion

According to Wand and Weber [25], conceptual modeling method should be studied further for the purpose of improving modeling quality. In line with the argument, we agree that a conceptual modeling method needs to supply a useful procedure on mapping domain knowledge into an output conceptual model. In this study, we focus on how to support novice designers who are lack of experiences in analyzing domain knowledge. In detail, we tried to understand the effects of applying the constant comparative method from grounded theory methodology. The exploratory experiment and interviews revealed that constant comparison can be useful, but we also learned there may be pitfalls.

Modeling is to abstract complex things into simple and easy-to-understand concepts and their interactions. Hence, understanding thoroughly the focal events in workplaces may be crucial for the modeler to discover core concepts for describing important facts. Although conceptual modeling may reduce available details, it is actually a process of constructing more general concepts applicable to various occasions. Definitely, comparing what have been analyzed to new things will increase probability to discover general concepts with fewer redundancies. Figuratively, that is a valuable work for organizing one's thought and connecting memories.

For system designers, a conceptual model is an output from a system requirement as an input for physical implementation. The transitional position requires both capability of analyzing domain knowledge and developing formal specification. Traditionally, research on conceptual modeling has focused on how to describe things properly in terms of modeling constructs [25]. Besides, even studies on knowledge elicitation have greatly stressed on preparing articulated inputs for analysis rather than a process of cognitive understanding those sources. We admit that the above mentioned efforts are quite valuable and useful for the system designers; however, the understanding effort itself should be not be overlooked. Especially, conducting conceptual modeling can be challengeable for novice system designers who are not familiar with a certain kind of domain knowledge. Although they should be trained by doing, it can be helpful if there is a set of procedural guideline to follow.

Our study contributes to the effort of searching possible elements for enhancing designer's understanding on given domain knowledge. We tested whether constant comparison from grounded theory methodology would be useful for novices to overcome difficulties of acquiring domain knowledge and improve acceptability by peer experienced designers. The in-depth interviews with the subjects who were divided by the experimental setting showed that our attempt yielded both positive and negative responses. The ability of utilizing previous experiences to the case of an unfamiliar domain increased, thus the novices could explain what they had done more fluently and clearly. The result was amazingly impressive since they also tried to discover how to express perceived concepts using modeling constructs. We felt that their work could be eventually improved as more efforts on learning database and programming would be done. On the other hand, a potential problem of applying constant comparison was also found. For a novice, linking analysis and drawing diagrams was perceived as separated tasks rather than a unified process. Originally, we thought that an explicit step for acquiring necessary information in advance might be natural and convenient. To the contrary, the novices we interviewed confessed that drawing diagrams was itself a kind of process to search alternatives visually. Since the primary purpose of conceptual modeling in the experiment was to produce a fitting model to the given task, the analytic side done with constant comparison methods might be perceived as an annoying thing. For a qualitative researcher, the process of constant comparison can be valuable itself since she may continuously develop theoretical models. Writing memos and rewriting codes can be useful to test theoretical assumptions at any moment. Linguistic memos as we used, however, may be unproductive for conceptual modeling. Additionally, comparing parts of models may require another measurement that is quite different from grounded theory research.

5. Conclusion

Our work contributes to the research

community in several ways. First, it focuses on how to support novice system designers when they need to accumulate domain knowledge. In fact, conceptual modeling has been regarded as an artful work rather than a cognitive process of understanding ethnographic differences. Requirements from business fields may be accumulated and situated results of social interactions and even visions. Those can be stated in verbatim expressions that are not clearly understood by others who are not in the same shoes. To overcome difficulties in understanding unfamiliar, unstructured knowledge materials, system designers should be capable of discovering hidden concepts and managing findings continuously. Admittedly, the expert designers can be a natural for conducting knowledge discovery processes before or with developing conceptual models. We can wait until the novice becomes the expert learning by doing in workplaces; but, if there is a plausible countermeasure to overcome the difficulty in handling unfamiliar domains, it is worth of sharing findings with peer researchers from the application.

Although repeatedly comparing things has been perceived as one of natural skills to learn unfamiliar events, there has been little knowledge on whether or not the method is actually helpful for the novice in improving performance. Moreover, the exotic guideline in terms of social science methodology may or may not be acceptable in the context of conceptual modeling. In our study, the constant comparative method can be used by the novice designer, and the result can be acceptable by other experience designers. However, the user of the method may be uncomfortable since there is a gap between the method and the purpose of conceptual modeling. Hence, we learned that the novice designer can do better when an additional procedural guideline of constant comparison is given. In addition, we recognized that academic efforts are required to integrate effective ingredients from various fields into conceptual modeling theories.

According to Pidgeon et al. [18], qualitative research methodologies from social science can be used for discovering knowledge from interview data. Although they heavily focused on articulating unstructured data to determine major requirements, the findings from the field study reveal that a system designer can work better if procedural guidelines borrowed from other mature fields are adequately used. Our study extends the finding by actually testing the focal principle of grounded theory in the context of entity-relationship modeling. In addition, Pidgeon et al. [18]'s work was related to general field experiences, whereas we tried to understand applicability in a purposeful micro view.

The study also has several practical implications. Conceptual modeling has been perceived as one of the most crucial steps for successful system development; however, the usefulness of the activity is limited to those who are able to understand both a system and the application domain. It is noted that a system designer should be ready for communication with business practitioners who may be lack of knowledge on actual implementation. It is difficult for a user to tell what she exactly wants before the system is physically shown since it is very difficult to imagine specifically what kind of functions can be supported by the information system without adequate technical background. It is usually posited that a conceptual model guides subsequent development phases. If so, the model needs to include voices from the business user at least in terms of a complete scenario about current jobs and a set of vocabularies that are needed to understand value activities.

The results of our study indicate that a procedure of analyzing user's voices by doing constant comparison can be helpful to support a system designer to communicate with the user. The method we applied in the study is originally intended to understand meanings of social experiences in which people are stuck in problems, such as chronic disease, demanding workload in intensive care units and conflicts with patients to name a few. Practically, a system development is to provide a solution for business problems to generate more values. In order to cure the problems and suggest better ways, the designer needs to be immersed in true voices from the workplace. Our study shows that the system designer can emulate the role of ground theorist who tries to investigate exhaustively empirical data to extract substantive theories. In the study, we found that there is a considerable chance to exploit the value of constant comparison, and, as a result, the idea presented in this study will be helpful in supporting practitioners.

We note that our findings must be interpreted in light of the study's limitations. First, we tried to obtain controlled samples that should not be exposed to conceptual modeling previously. And we trained them in order to learn about the effect of our suggested procedure. In the lab experiment setting, this approach produced a small number of samples that were not quite enough to use classical statistics, such as t-test, chi-square test and McNemar test. The sample size in the study was not sufficient to conclude that the constant comparative method will be determinately useful in analyzing domain knowledge in the context of conceptual modeling. Although the primary purpose of our study is to closely examine behavior of novice modelers to understand the applicability of the suggested idea, larger sample size will be eventually important for peer researchers who want to extend our exploratory findings. The other limitation is that the experiment we conducted only provided limited information since we did not track of the growth of the novice designers. If we collected data to learn the effect on designer's work performance in multiple times, the findings would be richer and meaningful.

In the experiment, we did not test alternative methods of investigating domain knowledge or multiple levels of familiarity and difficulties of tasks. If we want to obtain more useful information to develop an application that supports practitioners based on our findings, the constant comparative method imported from grounded theory should be tested in various levels. We posit that further effort should be done to test external reliability of the suggested method. The prescriptive theory developed in our work needs to be perceived as a start point rather than a complete conclusion. By conducting retests in different settings, we can collect much information about how to improve strengths and reduce potential risks. Those kinds of knowledge will never be obtained without cumulative efforts from information system research communities; therefore, our study encourages subsequent participation and cooperation of peer scholars.

References

[1] Anaby-Tavor, A., Amid, D., Fisher, A., Ossher, H., Bellamy, R., Callery, M., Desmond, M., Krasikov, S., Roth, T., Simmonds, I., and de Vries J., "An Empirical Study of Enterprise Conceptual Modeling," ER 2009, Lecture Notes in Computer Science 2009, Vol. 5829, Springer, Berlin, pp. 55-69, 2009.

[2] Banerjee, M., Capozzoli, M., McSweeney, L., and Sinha, D., "Beyond Kappa : a Review of Interrater Agreement Measures," Canadian Journal of Statistics, Vol. 27, No. 1, pp. 3-23, 1999.

[3] Bera, P., "A Cognitive Perspective on How Experts Develop Conceptual Models," Proceedings of Americas Conference on Information Systems (AMCIS 2011), August 12-15, Peru, Lima, Paper 422, pp. 1-10, 2010.

[4] Boehm, B. W., Software Engineering Economics, Prentice Hall, NJ, 1981.

[5] Bolloju, N. and Leung, F. S. K., "Assisting Novice Analysts in Developing Quality Conceptual Models with UML," Communications of the ACM, Vol. 49, No. 7, pp. 108-112, 2006.

[6] Charmaz, K., "Constructing Grounded Theory: a Practical Guide through Qualitative Analysis," Sage, London, 2006.

[7] Davis, G. B., "Strategies for Information Requirements Determination," IBM Systems Journal, Vol. 21, No. 1, pp. 4-30, 1982.

[8] Elmasri, R. and Navathe, S. B., Fundamentals of Database Systems, Addison-Wesley, Boston, 2007.

[9] Gefen, D., Karahanna, E., and Straub, D., "Trust and TAM in Online Shopping : an Integrated Model," MIS Quarterly, Vol.

27, No. 1, pp. 51-90, 2003.

[10] Glaser, B. G. and Strauss, A. L., The Discovery of Grounded Theory : Strategies for Qualitative Research, Aldine de Gruyter, NY, 1967.

[11] Goodman, P. and Leyden, D., "Familiarity and Group Productivity," Journal of Applied Psychology, Vol. 76, pp. 578-586, 1991.

[12] Gwet, K., "Inter-rater Reliability : Dependency on Trait Prevalence and Marginal Homogeneity," Statistical Methods for Inter-Rater Reliability Assessment, Vol. 2, pp. 1-9, 2002.

[13] Gwet, K., "Computing Inter-rater Reliability and Its Variance in the Presence of High Agreement," British Journal of Mathematical and Statistical Psychology, Vol. 61, No. 1, pp. 29-48, 2008.

[14] Kung, C. and Sølvberg, A., "Activity Modeling and Behavior Modeling of Information Systems," In Olle T. W., Sol H. G., and Verrijn-Stuart A. A. (eds), Information Systems Design Methodologies: Improving the Practice, North-Holland, Amsterdam, pp. 145-171, 1986.

[15] Moody, D. L., "Metrics for Evaluating the Quality of Entity Relationship Models," In Ling, T. W., Ram, S., and Lee, M. L. (eds), Proceedings of the Seventeenth International Conference on Conceptual Modeling, November 16-19, Singapore, Springer-Verlag, pp. 211-225, 1998.

[16] Mylopoulos, J., "Conceptual Modeling and Telos," In Loucopoulos P. and Zicari R. (eds), Conceptual Modeling, Databases and CASE : an Integrated View of Information Systems, Wiley & Sons, NY, pp. 49-68, 1992.

[17] Parsons, J. and Wand, Y., "Using Cognitive Principles to Guide Classification in Information Systems Modeling," MIS Quarterly. Vo. 32, No. 4, pp. 839-868, 2010.

[18] Pidgeon, N. F., Turner, B. A., and Blockley, D. I., "The Use of Grounded Theory for Conceptual Analysis in Knowledge Elicitation," International Journal of Man-Machine Studies, Vol. 35, No. 2, pp. 151-173, 1991.

[19] Schenk, K., Vitalari, N., and Davis, K., "Differences between Novice and Expert Systems Analysts : What Do We Know and What Do We Do?," Journal of Management Information Systems, Vol. 15, No. 1, pp. 9-50, 1998.

[20] Soffer, P. and Hadar, I., "Applying Ontology-based Rules to Conceptual Modeling : A Reflection on Modeling Decision Making," European Journal of Information Systems Vol. 16, pp. 599-611, 2007.

[21] Strauss, A. L., Qualitative Analysis for Social Scientists, Cambridge University Press, NY, 1987.

[22] Strauss, A. L. and Corbin, J., Basics of Qualitative Research, Sage, CA, 1990.

[23] Strauss, A., Corbin, J. M., Grounded Theory in Practice, Sage, CA, 1997.

[24] Wand, Y. and Weber, R., "Research Commentary : Information Systems and

Conceptual Modeling－a Research Agenda," Information Systems Research, Vol. 13, No. 4, pp. 363-376, 2002.

[25] Wand, Y., Storey, V. C., and Weber, R., "An Ontological Analysis of the Relationship Construct in Conceptual Modeling," ACM Transactions on Database Systems, Vol. 24, No. 4, pp. 494-528, 1999.

〈Appendix〉

The following description is a translated version of the requirement statement for Task 1.

Design an entity-relationship (ER) schema for searching movie information. You are required to design a conceptual model for a movie database of Jack's Movie Search. Your client's business objective is to provide information services based on this database. Each movie has a title and a release date. In addition, a movie has information on running time. A movie maker makes a movie, and customers will use the information from movie makers and the attributes of the movies themselves to decide on their preferences. Each movie is identified by a unique movie code. Movies can be classified into various categories, such as horror, action, and drama. A movie can have multiple classification tags.

A movie person refers to an individual who participates in movie production. Jack's Movie Search reports that a customer requires information on actors, actresses, and directors. A movie person can take any of these roles. For the reason that people may have multiple roles, movie persons should similarly be able to serve multiple roles. For example, Sungjoon Ryu was both an actor and the director for a 2010 movie. Each movie person has a name and a birth date. Moreover, a personal ID uniquely identifies each movie person. Jack's Movie Search typically provides customers with casting information. Customers can identify famous lines from the table with the heading "Monthly Casting," which can be found in the magazine published by Jack's Movie Search. The Chief Executive Officer wants to make services more flexible via information systems. Therefore, individual casting information should be outlined so as to provide details on a famous line from a certain movie.

A famous line has both content and an identification number. A number of movies share the same story plot. For example, The Shepherd Boy has two versions, 1945 and 1990. Different companies have released these versions, but the story is based on the same detective novel published in 1920. Jack's Movie Search has a number of PDF files on plots. Therefore, a customer can download a file if desired. Finally, a moviemaker has a name and an address. Each moviemaker can make more than one movie. Moreover, several moviemakers can collaboratively work on one movie.

The following description is a translated version of the requirement statement for Task 2.

Design an entity-relationship (ER) schema for providing transaction information. You are required to design a conceptual model for a company that provides automobile tuning parts. Zenka is a leading company that operates several tuning shops. The company presently plans to start a new business that provides automobile tuning parts for other service shops. Mr. Kim is the Operations Manager of Zenka. He says that Zenka has a good directory database with ZIP codes, addresses, and map images. Zenka's shops have used the directory information system for years. Mr. Kim added that Zenka plans to collect information on potential customers so that the database of the directory information system can contribute to the new business. He wants to be able to calculate the distance between a customer shop and the nearest Zenka shop, after which the result can be visually represented. Using location information, salespersons can plan the most efficient route to accomplish their tasks. The replacement of original parts using third-party parts may result in different car conditions. Thus, the salesperson should be very knowledgeable about the effects of applying tuning parts.

Zenka plans to hire a car mechanic as a part-time employee. Each mechanic has a name, an address, a cellular phone number, and work experiences. Additionally, mechanics have their own network, information on which is required to determine the proper combination of tuning parts. Mr. Kim states that over 80% of tuning parts consumed have been supplied through interpersonal networks. Therefore, monitoring such networks is important in formulating marketing strategies. The new information system is expected to show a list of shops that a salesperson can get in contact with, as well as the strength of the network. Customer shops are identified by a ZIP code. Zenka wants to show information on CEOs, such as a name, an email, and phone numbers. Each customer shop has an address, foundation date, and annual sales estimation. Mr. Kim insists that the new information system should be capable of calculating the total and average numbers of transactions conducted with each shop. He adds that Zenka needs to know the number of parts delivered to shops so as to improve logistics performance in terms of time spans and inventory. A transaction data instance shows who buys what and who sells when. Obviously, tuning parts should be delivered on time. Hence, Zenka plans to buy a warehouse near Seoul.

Each part has an identification number, a name, and manufacturer information. The warehouse keeps information on the inbound and outbound dates and assigns a shelf number for each part. Manufacturer information comprises manufacturer brand and component type. The new information system is expected to show an image of a part with its factory price, as needed by salespersons. A customer usually obtains information on turning parts from a professional magazine. For example, Brembo M4 Cast Monoblock is introduced with its description, an image, and vehicles that adopt the break system.

저 자 소 개

김태경	(E-mail : masan.korea@gmail.com)
2005년	서울대학교 경영대학 (학사)
2008년	서울대학교 경영대학 (석사)
2013년	서울대학교 경영대학 (박사)
관심분야	Organic media, 네트워크 비즈니스 경영, 개념적 모델링, 경영 혁신

박진수	(E-mail : jinsoo@snu.ac.kr)
1991년	계명대학교 문학사 (미국학 전공)
1994년	The University of Pittsburgh (석사)
1999년	The University of Arizona (박사)
2003년	University of Minnesota, Carson School of Management (조교수)
2005년	고려대학교 경영대학 (조교수)
현재	서울대학교 경영대학 (부교수)
관심분야	온톨로지, Semantic Web, 개념적 모델링

노상규	(E-mail : srho@snu.ac.kr)
1987년	서울대학교 경영학과 졸업 (학사)
1990년	University of Minnesota, Carlson School of Management 졸업 (석사)
1995년	University of Minnesota, Carlson School of Management, Information and Decision Sciences Dept. (박사)
현재	서울대학교 경영대학 교수
관심분야	Organic media, 네트워크 비즈니스 경영, 데이터베이스