8-7-2011

Applying Grounded Theory to Conceptual Data Modeling: Effects of Constant Comparative Method

Taekyung Kim
Department of MIS, Graduate School of Business, Seoul National University, ktk051@snu.ac.kr

Jinsoo Park
Department of MIS, Graduate School of Business, Seoul National University, jinsoo@snu.ac.kr

Sangkyu Rho
Department of MIS, Graduate School of Business, Seoul National University, srho@snu.ac.kr

Applying Grounded Theory to Conceptual Data Modeling: Effects of Constant Comparative Method

Taekyung Kim
Department of MIS,
Graduate School of Business,
Seoul National University, Republic of Korea
ktk051@snu.ac.kr

Jinsoo Park
Department of MIS,
Graduate School of Business,
Seoul National University, Republic of Korea
jinsoo@snu.ac.kr

Sangkyu Rho
Department of MIS,
Graduate School of Business,
Seoul National University, Republic of Korea
srho@snu.ac.kr

ABSTRACT

This article presents an experimental design and the pilot experiment results of applying grounded theory to conceptual data modeling. The objective of this study is to develop a procedural method for concept discovery, which is essential in data modeling. The research focuses on addressing the lack of procedural methods for understanding domain knowledge by data modeler. The key idea of this article is that conceptual modeling can be strengthened by applying a constant comparative method of coding and analysis, which has been used to discover concepts in the social sciences. This article contributes new knowledge about the effects of applying interdisciplinary concept discovery in the context of conceptual data modeling. The results of the pilot experiment show that the proposed approach would have positive results.

Keywords

Data modeling, ER modeling, grounded theory, experiment

INTRODUCTION

Information system professionals have used conceptual modeling to address various business requirements. Innovation through such activities as business reengineering, implementation of new e-services, and customizing enterprise resource planning begins with understanding domain knowledge. Conceptual data modeling aims to fulfill the needs of both data modelers and business practitioners by providing a capability of communication.

Unsurprisingly, conceptual modeling has been considered as the most critical activity to ensure success, because common understanding of domain knowledge can evidently improve productivity of system development. Recent empirical studies have shown that data modeling professionals want to have practical and effective tools to address this issue (Fettke, 2009).

The surprising fact is that conceptual modeling itself is not supported by analytical processes. Research on conceptual modeling has suggested various alternative means for modeling domain knowledge by developing new modeling diagrams (i.e., modeling constructs). However, current problems of conceptual modeling seem to be linked with an insufficient capability to investigate domain knowledge, not insufficient modeling constructs. Bolloju and Leung (2006) argue that the problems in conceptual modeling are coupled not only with insufficient pedagogical sources for knowledge analysis but also with lack of experience in the analysis of an unfamiliar domain. Fettke (2009) supports that modeling professionals want to gain procedural knowledge on domain analysis or conceptualization. In addition, academic efforts to educate qualified modelers are likely hampered by limited research on developing procedural knowledge. Waiting for a novice modeler who lacks conceptualization skills to become an expert is not a practicable option because businesses are rapidly moving toward e-transformation.

In this article, the application of grounded theory to data modeling to establish elementary knowledge on how to effectively extract concepts is introduced. Grounded theory is a scientific research methodology for developing empirical theories. Qualitative researchers usually adopt this methodology to explore a social phenomenon that needs to be described by concepts such as hidden social categories, dimensions, and relationships. The objective of the present study is to discover the utility of constant comparison of coding and analysis, which is a core principle of grounded theory, in the context of data modeling. Grounded theory is known for its power to assist social scientists in understanding domain knowledge with which outside observers are unfamiliar.

An experimental methodology was selected from among various options for the present study based on the standpoint that internal validity is the most important requirement to ensure the positive effects of applying grounded theory in the early phase of research. Because a causal characteristic of a time-related event can be observed by a simple treatment in a controlled environment (Cook and Campbell, 1979), the experiment can clarify whether applying grounded theory results in better conceptual models. In this article, the research background is first discussed. Especially, we focus on perception issues in conceptual modeling. In addition, the constant comparison of grounded theory is introduced. Next, the experiment methodology is presented by following the experiment framework suggested by Javenpaa (1985), and Parsons and Cole (2005). We believe that describing the factors of our experiment in a certain framework is helpful for communication and further research development. Next, results of the pilot experiment are then discussed. Suggestions for further research conclude the article.

RESEARCH BACKGROUND

Perception

The main purpose of conceptual modeling is to elicit the conceptual modeling script of the corresponding information systems (Wand and Weber, 2002). In the context of information systems, one of its primary objectives may be to obtain feasible and valuable blueprints for database construction (Wand, Storey, and Weber, 1999). Ideally, the scripts written by a certain conceptual modeling grammar should be independent of specific database management systems (Chen, 1976). However, the popularity of relational database systems has primarily limited conceptual models as the initial backbone of logical database models. In addition, practitioners largely use the Entity-Relationship diagram or its equivalents as a major conceptual modeling grammar (Davies, Green, Rosemann, Indulska, and Gallo, 2006).

Interestingly, some empirical studies have demonstrated that much of the implementation failure of databases resides in requirement errors (Lauesen and Vinter, 2001; Martin, 1989). The cost of the initial design failure may increase exponentially over the whole database design process, and could be substantial (Boehm, 1981). Theorists have argued that conceptual modeling should involve activities to link modeler perceptions of real-world knowledge to concept representation (Wand, Storey and Weber, 1999). If the perceived structure of the universe of discourse does not meet a requirement level, the conceptual modeling script may not be useful for assessing business functionality in terms of value creation (Browne and Ramesh, 2002).

Wand et al. (1995) suggest that a conceptual model is an outcome of analysis of real-world perception. Conceptual modeling involves a process of harmonizing with the perception of business practitioners. This kind of alignment is difficult because of (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 (Davis, 1982), among others.

If a business practitioner can describe her or his perception in a clear and formal or semiformal manner, then the modeler would simply have to describe or translate matters with modeling diagrams (Simsion, 2007). Unfortunately, a modeler occasionally has to collect various information sources, including business goals, processes, transaction data, and user behavior (Browne and Ramesh, 2002). Batra and Davis (1992) have shown that the experienced modeler uses strategies to design conceptual models different from those of the novice modeler, with the former type decomposing a problem into knowledge chunks and using modeling patterns and the latter one demonstrating incapability to use certain strategies to construct conceptual models. As Wand and Weber (2002, p. 368) state, "Creating a faithful representation with a grammar entails two activities: identifying the phenomena to be modeled and mapping the phenomena into the grammar's constructs." Nevertheless, committing to these activities seems to be directly relevant to field experiences, rather than academic training or using well-guided procedures.

Grounded Theory

Grounded theory is a widely accepted qualitative research methodology that systematically derives theories of human behavior based on the perception of researchers. It helps social scientists develop theories rooted in empirical data. Generating substantive theories is a primary objective for adopting grounded theory (Glaser and Strauss, 1967). It is typically applied to content analysis of transcribed interviews, observation notes, and other data directly collected in a specific context (Glaser, 1992).

Grounded theory uses constant comparison of joint coding and analysis to identify concepts and their relationships (Glaser and Strauss, 1967). This method compares substantive findings as many times as possible over the analysis process.

Grounded theory necessitates the application of a constant comparative method, from reading each incident to writing theories (Glaser, 1992). The following is the illustration of grounded theory to theory development in the social sciences (Glaser and Strauss, 1967; Glaser, 1992).

- **Comparing incidents:** A researcher may consider and note possible concepts while s/he reads data. An incident refers to an occurrence in a specific context (Glaser and Strauss, 1967). It may be defined as an event that contains relevant information about an interesting topic. A researcher can produce incidents by grouping relevant concepts, such as categories, properties, and dimensions. Each incident needs to be compared repeatedly until the researcher feels that all incidents are rich and clear.
- **Comparing developed concepts:** Some incidents may have an association. By comparing these incidents, the researcher develops higher levels of concepts that cover juxtaposed meanings of the incidents.
- **Comparing meaning structures:** The researcher investigates meaning structures of high-level concepts or categories. Each category may have its own meaning structure. The researcher may have substantive theories while s/he compares meaning structures to explain relationships between categories.

Some studies suggest that grounded theory may be useful for data modeling. Lamp and Milton (2007) argue that a data modeler is able to investigate meaning structures of domain knowledge by using a tool for grounded theory. The result can be an input for an ontology of a data model. Pidgeon, Turner and Blockley (1991) have directly investigated the effects of grounded theory on requirement engineering. The reason is that grounded theory is known for its power to extract tacit knowledge from interview data. Pidgeon et al. (1991) hypothesized that grounded theory would improve knowledge acquisition for data modeling. Their case study has demonstrated that grounded theory has potential; however, directly using qualitative research methods could be harmful.

Compared with Lamp and Milton (2007), Pidgeon et al. (1991) focus more on communication issues. Understanding domain knowledge may be crucial for data modeling, as corroborated by their research. Clear communication can enhance the performance of requirement analysis (Hirschheim, Klein and Lyytinen, 1995). Grounded theory is expected to achieve this goal by enhancing understanding of field practices. Learning concepts grounded in the field and understanding the meaning structures formed by those concepts are both important for data modeling (Coughlan and Macredie, 2002).

METHODOLOGY

Experimental Design

A researcher designs an experiment to test a causal characteristic in the form of a trial (Cook and Campbell, 1979). A successful experimental design leads to strong internal validity. According to Parsons and Cole (2005), an experiment for conceptual modeling needs to include four major factors: independent variables, dependent variables, participants, and an experimental procedure. In the present study, the research plan is described by each factor. Because the whole experiment plan is not completely executed when the paper is submitted, we expect that the details we present will help readers to understand and verify results of our pilot experiment (Jarvenpaa et al., 1985).

Experimental Procedure

Figure 1 shows the experimental procedure. The diagram was partially taken from UML notations. There are four phases: “ready,” “pilot experiment,” “experiment,” and “analysis.” The ready phase first developed the experimental environment. Two Java applications for an experimental group and a control group were developed. Five qualified MIS professionals checked the usability of the applications. Open lectures to control prior knowledge of participants were then conducted. The lectures consisted of data modeling foundations, ER notations, and case studies. Next, from November 2010 to January 2011, the first pilot experiment participants were trained as business data modelers. They all spent a minimum of 18 hours to learn data modeling and 6 hours to conduct case studies. They were tested using textbook examples from Sanders (1995). Advanced subjects, such as generalization and specialization, were not included.

The experimental process is ongoing as of this writing. The second phase and basic analysis and feedback modules in the fourth phase were completed before the third phase commenced in January 2011. All modules in the procedure are scheduled to be completed by May 2011.

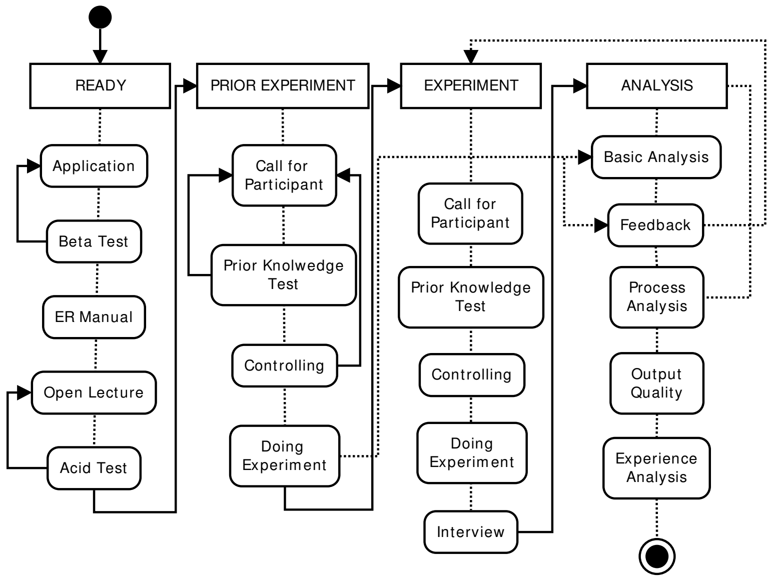


Figure 1. Experimental Procedure

Independent Variable

Social scientists who adopt grounded theory usually generate quantifiable codes. Grounded theorists present procedural coding styles, such as open coding, axial coding, and selective coding. All these coding styles must follow the constant comparative method of joint coding and analysis. First of all, a researcher must be viewed as a *tool* (i.e., the researcher uses her or his intellectual power to understand a phenomenon and represents the phenomenon with an adequate symbol). Therefore, given materials are compared with previous concepts stored in the knowledge base of the researcher.

Second, all the incidents are compared to review what had been literally symbolized using the theoretical sensitivity of the researcher. The second layer of constant comparison takes the initial analysis result in order to investigate meanings more thoroughly. Theoretical sensitivity refers to the power of a researcher to focus on important concepts and link associable information.

Finally, the emergence of analysis needs to be considered. Grounded theorists stress that note-taking is important because it helps a researcher manage emerging concepts without overly intervening with other activities. During the analysis, the researcher uses her or his emerging concepts to categorize substantive findings. Because the emerging concepts may be locally defined, additional general concepts may need to replace them. In this sense, a tool that supports the constant comparative method should compare emerging concepts for enhancement.

In the present study, the participants were randomly divided into two groups. The experimental group used a tool that supports the constant comparative method with drawing capability, whereas the control group used a drawing tool without the constant comparative method. The mechanism of combining the constant comparative method with data modeling capability was named the "concept magnifier." The Java applications provided to both groups had ER drawing capability; however, only the application for the experimental group used the concept magnifier. In summary, the present study adopted one independent variable with a Boolean value: if a subject is in the experimental group, the data instance has 1; otherwise, the point is 0.

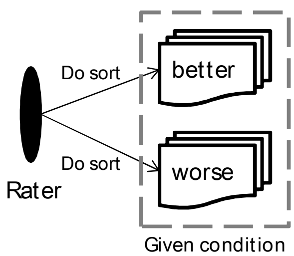
Dependent Variables

Empirical studies have shown that information system development can be prone to error unless conceptual modeling is successful (Moody, 2005). If conceptual models can be evaluated in advance, the overall cost of development could be significantly cut. The present study aims to produce better conceptual models by applying grounded theory. Table 1 summarizes the dimensions of conceptual modeling quality that have been repeatedly suggested by other researchers.

Dimensions of Conceptual Modeling Quality	Reference
Syntactic, semantic, and pragmatic qualities	Lindland, Sindre, and Sølvberg (1994)
Completeness, integrity, flexibility, understandability, correctness, simplicity, integration, and implementability	Moody (1998)
Understandability, legibility, simplicity, analyzability, modifiability, stability, and testability	Genero, Jimenez, and Piattini (2000)
Accuracy, completeness, conflict free, and no redundancy	Shanks, Tansley, and Weber (2003)

Table 1. Evaluation of Conceptual Data Models

Although the term "better" is subjective, comparing instances of a given set to select the better one is acceptable primarily because there are no generally accepted guidelines for evaluation in the academic field; practitioners use their own criteria to judge the quality of data modeling. Academic studies have suggested some characteristics of good conceptual models (shown in Table 1); however, the present study assumes that practitioners may use their experience and insight to determine the best alternative. Figure 2 illustrates how conceptual modeling quality was measured in the present study.



Let $s_i \in S$ be a conceptual model to be rated, where S is a set of all the conceptual models that are obtained from an experiment.

$f(s_i) = 1$, if an item is selected. The function $f(\cdot)$ denotes a judgment action.

Otherwise, $f(s_i) = 0$

Let raters be numbered from $j = 1$ to N. We can calculate the size of S. Thus,

$$\text{Expected Quality Score of a Script} = \frac{\sum_{j=1}\sum_{i=1} f_j(s_i)}{|S| \cdot N}$$

Figure 2. Sorting Task

Function $f(\cdot)$ in Figure 2 represents the judgment of each rater. If a given instance of a conceptual model is in the "better" group, the instance receives a 1. Each rater sorts instances, and an application that supports the sorting task automatically assigns points. The AC_1 inter-rater reliability proposed by Gwet (2002) ensures that the sorting task is adequate. AC_1 was chosen among such alternatives as Cohen's κ and Scott's π because it fares better than others when the extent of agreement between raters is high (Gwet, 2008). Inter-rater reliability measures the proximity of scores assigned by a group of raters to the same object. If the scores are sufficiently high, the data collection method may be considered more reliable (Banerjee, Capozzoli, McSweeney and Sinha, 1999).

Three practitioners voluntarily participated as raters in the present study. All raters were experienced in conceptual data modeling in the workplace. All spent at least three years in major system development companies. Tasks for the experiment were discussed with the raters, who were all qualified to evaluate results.

Participants

Two laboratory experiments were conducted to test the effects of applying the constant comparative method of grounded theory. The first pilot experiment participants to test the feasibility of the present study were recruited in November 2010. Although inclusion and exclusion criteria were not set, all the participants were undergraduate students who majored in business. The participants were strongly motivated to learn about data modeling, the primary reason being that most of them had spent at least 2 months for internship at work, and they had been informed of the importance of database systems.

Recruiting people who do not have experience in data modeling has two major benefits for the present study. First, experiences can be easily controlled because the same learning materials and modeling tasks for training are given. Second, reliability can be strengthened. Conducting an experiment on data modeling with practitioners may be difficult in that results may be less helpful to careers or daily work. The participants can be motivated to exert minimal effort, or they can be interested only in the monetary benefits of their participation. However, the present study offered business students the opportunity to learn something new about data modeling and guaranteed them feedback from task evaluation, participation features that likely positively influenced the experiment.

The pilot experiment had a sample size of 10. This number was actually half of the original volunteers. After finishing lecture sessions, each participant was tested on basic ER modeling. Individuals who failed the test were dropped because the experimental tasks required modeling knowledge and skills. The participants were randomly assigned to the experimental group or the control group.

Tasks

Two tasks were developed based on class examples from the textbook of Elmasri and Navathe (2007). Because the participants were Korean speakers, the original examples were carefully translated. The tasks of the experiment had different domains: the first one involved a movie database, and the other was related to ordering automobile parts. Requirement specifications were made considering actual services for each domain. Two academic researchers initially checked the tasks, after which minor errors were corrected and unclear sentences were modified. The final versions were sent to two field professionals for revalidation.

No answers were assumed as correct in the modeling tasks of the experiment. However, overall outlines of modeling results could be assessed based on the opinions of the raters who had validated the tasks. Table 2 shows a summary of the minimum features of each modeling task. If complexity can be defined by the number of diagram elements, the two tasks are similar in terms of complexity. However, Task 2 is twice as long as Task 1 because its description is more detailed, which means analysis of the business requirement in Task 2 required more time.

	Domain	Word count	Strong entity	Weak entity	Relationship	Attribute
Task 1	Movie	233	8	1	8	17
Task 2	Car parts transaction	589	5	1	4	19

Table 2. Minimum Features of the Modeling Tasks

Pilot Experiment

The pilot experiment can be considered as a simulation for our target experiment. The procedure was divided into four parts: call for participation, prior knowledge testing, controlling, and experimentation. Recruitment was carried out by distributing movie files with user-generated content about how to use Microsoft Access 2007 using Web-streaming services. This strategy was chosen because many white-collar workers today want to manage information using databases but have had limited opportunities to satisfy the requirements. E-mails, Internet boards, and word-of-mouth channels were used to call for participation. Of 27 persons who showed interest in the present study, 20 actually joined the first open lecture about conceptual data modeling.

The prior knowledge test was two-fold. First, whether an individual participant had sufficiently learned about Access 2007 was tested. Each participant was personally invited to report any difficulties in learning Access 2007. One of the authors took charge of handling questions and answers to assist the participants. The participants were asked whether they had any intention taking additional lectures about conceptual data modeling after the invitation. After 18 hours of learning sessions (i.e., 9 hours of classes and another 9 hours for assignments), one week was allotted to helping the participants work with a prototype of an ER editor. Ten persons regularly visited the laboratory, and two cases from Sanders (1995) were discussed face-to-face. At the end of the session, the authors discussed whether the participants were qualified for testing.

The controlling phase divided the participants into two groups: experimental and control. Excel was used to randomly assign them. Members of the experimental group had three hours to learn about grounded theory. Basic notations, such as open coding, axial coding, and selective coding, were included. In addition, the experimental group received a 30-minute tutorial about how to use a Java application implementing the concept magnifier. The control group had the same length of time to examine one additional modeling case. They also received a tutorial for application usage.

Finally, the three-hour pilot experiment was conducted. The participants were expected to finish Task 1 in 60 minutes and Task 2 in 90 minutes. We did not put any constraint for time usage; that is, a participant could freely decide how much time s/he would spend for analysis or modeling. This policy was created to collect information for subsequent experiments. As a result, the participants, on average, spent almost 50% of their time analyzing descriptions. The experiment was recorded by typing activities with a portable device. Questions other than those on application usage were not entertained.

RESULTS

The results of the pilot experiment are summarized in Table 3. First, the control group spent more time on Task 2. Compared with the experimental group, the control group produced more relationships and deleted more ER diagrams in Task 2. Although the overall quality was higher than 0.6 for both groups in Task 1, the control group showed a dramatic decline from 0.62 to 0.13 in Task 2. The experimental group demonstrated the same tendency; however, the effect was smaller. For the experiment group, an average of 0.87 in Task 1 declined to 0.60 in Task 2. The difference is only 0.27. The control group has a mean difference of 0.54. The evaluations are reliable according to the AC_1 of Gwet.

	Control Group		Experimental Group	
	Task 1	Task 2	Task 1	Task 2
Analysis time	30 minutes	28 minutes	36 minutes	30 minutes
Modeling time	31 minutes	44 minutes	25 minutes	24 minutes
Entities	30 units	28 units	32 units	22 units
Attributes	59 units	78 units	50 units	78 units
Relationships	29 units	40 units	26 units	27 units
Deletion	47 times	56 times	32 times	31 times
Quality evaluation	0.67	0.13	0.87	0.60
Inter-rater reliability (Gwet's AC_1)	0.62		0.67	

Table 3. Results of the Pilot Experiment

CONCLUSION

This research studied the effects of applying the constant comparative method of grounded theory to data modeling, and still in progress. It specifically examined the background idea, the research methodology, and the results of a pilot study. To the best of our knowledge, the present study is the first experiment to apply grounded theory in establishing a procedural principle for conceptual data modeling. We expect our results to help current theoretical and practical achievements become better understood and further expanded.

The results of the pilot experiment show that applying grounded theory to conceptual data modeling has positive effects. We empirically examined whether the participants in the experimental group became easily accustomed to using the constant comparative method of grounded theory. The participants reported that grounded theory helped them choose modeling constructs and patterns by providing an analytical method. We believe that the findings of the present study will strengthen our further research. The first direction is to analyze modeling processes. The power of a constant comparative method may give a data modeler the information cues that may serve as a guideline for assigning entities, attributes or relationships. Therefore, we expect that modeling processes will be depicted differently due to the effect of the constant comparative method. The second direction is to develop an application that implements grounded theory for data modeling by applying semantic technologies. Because grounded theory usually produces lots of substantive concepts and associations, a data modeler may feel difficulties to manage analysis results. We expect that the resource description framework or RDF may be used to support the data modeler.

REFERENCES

1. Banerjee, M., Capozzoli, M., McSweeney, L. and Sinha, D. (1999) Beyond kappa: A review of interrater agreement measures, *Canadian Journal of Statistics*, 27, 1, 3–23.
2. Batra, D. and Davis, J. G. (1992) Conceptual data modelling in database design: Similarities and differences between expert and novice designers, *International Journal of Man–Machine Studies*, 37, 1, 83–101.
3. Boehm, B. W. (1981) Software engineering economics, Prentice Hall, Englewood Cliffs.
4. Bolloju, N. and Leung, F. S. K. (2006) Assisting novice analysts in developing quality conceptual models with UML, *Communications of the ACM*, 49, 7, 108–112.
5. Browne, G. J. and Ramesh, V. (2002) Improving information requirements determination: A cognitive perspective, *Information and Management*, 39, 8, 625–645.

6. Chen, P. P. (1976) The entity-relationship model – Toward a unified view of data, *ACM Transactions on Database Systems*, 1:1, 9-36.
7. Cook, T. D. and Campbell, D. T. (1979) Quasi-experimentation: Design & analysis issues for field settings, Houghton Mifflin, Boston.
8. Coughlan, J. and Macredie, R. D. (2002) Effective communication in requirements elicitation: A comparison of methodologies, *Requirements Engineering*, 7, 2, 47–60.
9. Davies, I., Green, P., Rosemann, M., Indulska, M. and Gallo, S. (2006) How do practitioners use conceptual modeling in practice? *Data and Knowledge Engineering*, 58, 3, 358–380.
10. Davis, G. B. (1982) Strategies for information requirements determination, *IBM Systems Journal*, 21, 1, 4–30.
11. Elmasri, R. and Navathe, S. B. (2007) Fundamentals of database systems, Addison-Wesley, Boston.
12. Fettke, P. (2009) How conceptual modeling is used, *Communications of the AIS*, 25, 43, 571–592.
13. Genero, M., Jimenez, L. and Piattini, M. (2000) Measuring the quality of entity relationship diagram, in Alberto Laender, Stephen Liddle and Veda Storey (Eds.) *Proceedings of the Nineteenth International Conference on Conceptual Modeling*, Salt Lake City, Utah, USA, October 9-12, Springer-Verlag, 513–526.
14. Glaser, B. G. (1992) Emergence vs. forcing: Basics of grounded theory analysis, Sociology Press, Mill Valley.
15. Glaser, B. G. and Strauss, A. L. (1967) The discovery of grounded theory: Strategies for qualitative research, Aldine de Gruyter, New York.
16. Gwet, K. (2002) Inter-rater reliability: Dependency on trait prevalence and marginal homogeneity, *Statistical Methods for Inter-Rater Reliability Assessment*, 2, 1–9.
17. Gwet, K. (2008) Computing inter-rater reliability and its variance in the presence of high agreement, *British Journal of Mathematical and Statistical Psychology*, 61, 1, 29–48.
18. Hirschheim, R., Klein, H. K. and Lyytinen, K. (1995) Information systems development and data modeling: Conceptual and philosophical foundations, Cambridge University Press, New York.
19. Jarvenpaa, S., Dickson, G. and DeSanctis, G. (1985) Methodological issues in experimental IS research: Experiences and recommendations, *MIS Quarterly*, 9, 2, 141–156.
20. Lamp, J. and Milton, S. K. (2007) Grounded theory as foundations for methods in applied ontology, in *QualIT 2007*: *Qualitative Research: From the Margins to the Mainstream Abstracts and Papers*, Wellington, New Zealand, Victoria University of Wellington, November 18-20, 1–13.
21. Lauesen, S. and Vinter, O. (2001) Preventing requirement defects: An experiment in process improvement, *Requirements Engineering*, 6, 1, 37–50.
22. Lindland, O.I., Sindre, G. and Sølvberg, A. (1994) Understanding quality in conceptual modeling, *IEEE Software*, 11, 2, 42–49.
23. Maiden, N. and Gizikis, A. (2001) Where do requirements come from? *IEEE Software*, 18, 5, 10–12.
24. Martin, J. (1989) Information engineering: Book III. Design and construction, Prentice Hall, Englewood Cliffs.
25. Moody, D.L. (1998) Metrics for evaluating the quality of entity relationship models, in Tok Wang Ling, Sudha Ram and Mong Li Lee (Eds.) *Proceedings of the Seventeenth International Conference on Conceptual Modeling,* November 16–19, Singapore, Springer-Verlag, 211–225.
26. Moody, D.L. (2005) Theoretical and practical issues in evaluating the quality of conceptual models: Current state and future directions, *Data & Knowledge Engineering*, 55, 3, 243–276.
27. Parsons, J. and Cole, L. (2005) What do the pictures mean? Guidelines for experimental evaluation of representation fidelity in diagrammatical conceptual modeling techniques, *Data & Knowledge Engineering*, 55, 3, 327–342.
28. Pidgeon, N.F., Turner, B.A. and Blockley, D.I. (1991) The use of grounded theory for conceptual analysis in knowledge elicitation, *International Journal of Man-Machine Studies*, 35, 2, 151–173.
29. Sanders, G.W. (1995) Data modeling, Boyd & Fraser Publishing, Danvers.
30. Shanks, G., Tansley, E. and Weber, R. (2003) Using ontology to validate conceptual models, *Communications of the ACM*, 46, 10, 85–89.
31. Simsion, G.C. (2007) Data modeling: Theory and practice, Technics Publications, Denville.

32. Teorey, T.J., Lightstone, S. and Nadeau, T. (2006) Database modeling and design (4th edition), Morgan Kaufmann, San Francisco.
33. Thalheim, B. (2000) Entity–relationship modelling: Foundations of database technology, Springer-Verlag, Berlin.
34. Wand, Y., Storey, V.C. and Weber, R. (1999) An ontological analysis of the relationship construct in conceptual modeling, *ACM Transactions on Database Systems*, 24, 4, 494–528.
35. Wand, Y. and Weber, R. (2002) Research commentary: Information systems and conceptual modeling — A research agenda, *Information Systems Research*, 13, 4, 363–376.