A Comparative Study of Conceptual Data Modeling Techniques

Heeseok Lee  
Korea Advanced Institute of Science and Technology  
Byoung Gu Choi  
Korea Advanced Institute of Science and Technology

This paper compares conceptual data models including extended entity relationship (EER), semantic object model (SOM), object role modeling (ORM), and object modeling technique (OMT) in terms of model correctness, modeling time, and perceived ease-of-use. For an empirical study, 28 graduates and 72 undergraduates were selected and then divided into four equally sized groups. Each group was trained with one data modeling technique. Two cases were used; one was prepared in natural language and the other in enterprise form. The study results show some differences among the four conceptual data modeling techniques. These positive findings may help modelers better understand modeling techniques.

By using a conceptual data model one can describe a reality. A conceptual data modeling technique makes it easy to understand and interpret reality; therefore, selecting conceptual data modeling technique is very important for effective database design. The quality of the resulting schema depends not only on the skill of the database designers and users, but also on the qualities of the technique selected (Batini et al., 1992). Advocates of each conceptual data modeling technique assert that his or hers is better in model correctness and shorter time required in solving problem, but there is little evidence to support their assertions (Kim and March, 1995).

This paper compares four conceptual data modeling techniques: EER (Extended-Entity-Relationship), SOM (Semantic Object Model), ORM (Object Role Modeling), and OMT (Object Modeling Technique).

EER is an enhanced version of an Entity-Relationship (ER) model, the most popular data modeling technique that was proposed by Chen (1976). EER consists of entities, relationships, and attributes (Teorey, et al., 1986). Entities represent classes of real-world objects. Relationships represent real-world associations among one or more entities. Attributes represent elementary properties of entities or relationships. ER based models enjoy wide acceptance, which means that there will be a great likelihood that most other modelers will understand the model notation and will not require education to use them (Keuffel, 1996).

SOM was originally developed by Kroenke (1995); it is very similar to EER, but does not include the concept of relationship; all relationships are represented in terms of attributes. A semantic object is defined as a named collection of attributes that sufficiently describe a distinct identity. An attribute describes a characteristic or trait of the object being modeled. In SOM, relationships are always two-way, and are represented by a semantic object link (SOL) attribute.

OMT (Blaha et al., 1988) is an improvement of ER, and Logical Relational Design Methodology (LRDM) model (Teorey et al., 1986). OMT is a modeling technique that consists of three views of modeling systems such as object model, dynamic model, and functional model, but we concentrate ourselves on the object model because it contains a data model. The object model consists of class, attribute and associate. A class is defined as a group of objects with similar properties (attributes). An object is a data value held by objects in class. An association is a group of links with common structure and common semantics. A link is a physical
or conceptual connection between object instances. OMT as well as SOM offers the modeler the unique ability to hide details in objects of increasing generality (Keuffel, 1996).

Halpin (1995a) proposed ORM, based on NIAM (Nijssen Information Analysis Methodology) that was developed in Europe in the 1970s (Nijssen, 1994). It is widely used in Australia and Europe. ORM views the application world in terms of objects that play roles. Objects are similar to both entities and attributes in EER; that is, objects are defined as events or real-world things which are needed for business. The object is represented by two types; one is entity type which is shown as a solid ellipse, and the other is value type which is shown as a broken ellipse. Roles are defined as what is played by an object. Objects have one or more roles. Because ORM expresses design details using English sentences, many developers have found it an effective way to present a database design to non-technical end users who can easily understand it, and correct it where necessary. This ability to verbalize a model is a unique and compelling advantage of ORM (Halpin, 1995b).

Lastly, Table 1 depicts graphical notations for the four conceptual data modeling techniques. For further details, readers are referred to each reference.

In this paper, we compare conceptual data modeling techniques in terms of model correctness, modeling time, and perceived ease-of-use. Model correctness can be defined as the degree to which a conceptual model approaches the correct solution(s). Modeling time is the time duration needed for the modeler to complete the modeling task. Perceived ease-of-use is defined as the degree to which a person believes that using a particular technique would be free effort. It is measured by the use of questionnaires developed by Davis (1989).

In addition, we perform two other studies on data modeling techniques. First, we study the interaction effect between data modeling techniques and past experiences on EER; second, we study the correlation between natural language case and enterprise form case.

### Literature Review

Prior studies can be classified into three categories: (i) comparison of classical data models, (ii) comparison of conceptual data model with classical model, and (iii) comparison of conceptual data models. The first category compares one classical data model (for example, a hierarchical or network data model) with another classical data model. A data model is a collection of concepts that can be used to describe a set of data, and operations to manipulate the data. When a data model describes a set of concepts from a given reality, it is called a conceptual data model (Batini et al., 1990). In this paper, a classical model (e.g., hierarchical, network, relational) implies a data model. The second category compares a classical data model with a conceptual data model (for example, EER or SOM). The third compares one conceptual data model with another conceptual data model. Most recent researches are classified within the third category. Table 3 is a summary of past studies. It also compares our current research with those of others.

Brosey and Shneiderman (1978) compared relational and hierarchical data models, using instance diagrams. They investigated comprehension, problem solving situation, and memorization. The experiment was performed by the use of undergraduate subjects who were classified into programmers and non-programmers. They found that programmers performed the tasks better, while non-programmers used the hierarchical model more easily. Hoffer (1982) performed an experiment comparing data models and reported the result of an observed individual difference in using a database model. Subjects were free to select the model approach they preferred. Though there are many modeling techniques, most subjects selected a process flow structure. It was found that subjects had difficulties in identifying database keys and relationships from their conceptual models.

Shoval and Even-Chaime (1987) compared normaliza-