Accuracy in Modeling with Extended Entity Relationship and Object Oriented Data Models

Douglas B. Bock
Southern Illinois University at Edwardsville

Terence Ryan
Indiana University at South Bend

This research compares success in developing conceptual data models for the extended entity relationship (EER) model and Kroenke’s object oriented model. A laboratory study was used to evaluate model correctness for 38 subjects divided into two equally sized groups where each group was trained in one of the modeling methods. Modeling correctness is measured in terms of eight different facets of a conceptual data model: (1) entities/objects, (2) attribute/property identifiers, (3) categories, (4) unary one-one relationships, (5) binary one-many relationships, (6) binary many-many relationships, (7) ternary one-many-many relationships, and (8) ternary many-many-many relationships. The EER model provided significantly improved performance for the attribute/property identifier, unary one-one relationship, and binary many-many relationship facets.

A number of research areas are concerned with human factors and database technology. These include information elicitation, query formulation, and data modeling. This paper focuses on data modeling and extends prior data modeling research by investigating the degree of model correctness achievable by entry-level information system professionals for two methods commonly used to develop conceptual data models.

Conceptual data models primarily use graphical symbols to represent data and data relationships in an implementation independent manner. Model correctness is important for the creation of properly working information systems. A conceptual data model can be incorrect in several ways, for example, the model may fail to identify a class of data items to be represented, or the model may fail to capture fully the semantic meanings of individual data items, or the model may fail to specify correct relationships among data items. Each situation can result in an information system that is inadequate with respect to various system user information requirements. The implications of modeling error underscore the importance of understanding the impact of various conceptual data models on model correctness.

One of the most widespread conceptual data modeling methods is the entity-relationship (ER) model first proposed by Chen (1976). The ER model is classified as a semantic data model (Peckham and Maryanski, 1988) because it enables the capture of important meanings about data and relationships among data. Semantic data models alleviate some of the inadequacies of the three major classical data models: the relational, hierarchical, and network models (see Kent, 1979 for a description of record based data model limitations).

The ER model provides for the graphical representation of entities (i.e., real or abstract things about which data are collected), entity attributes (properties of entities), and relationships between entities. An extension of the ER model by Elmasri et al. (1985), termed the extended entity-relationship (EER) model, includes the ability to model entity categories (class-subclass).

Over the past decade, the object oriented (OO) model has grown in popularity as an alternative conceptual data modeling approach. OO modeling methods evolved from concepts put forth by Smith and Smith (1977) and Hammer and McLeod.
(1981) and are also classified as semantic modeling methods. These original works prescribed a definitional approach to data modeling using a set of extensive descriptive clauses to describe and define data. Recently Kroenke (1991), Coad and Yourdon (1991), and others have introduced various graphical extensions of the OO modeling approach.

Object oriented designers represent the world in terms of objects, i.e., real or abstract things about which data are collected. Like entities, objects have properties and there can exist relationships between objects. Example objects include customer orders, invoices, and management reports. Unlike entities, an object may be composed of successively smaller objects, or an object may be a component of more than one larger object.

Although both the EER and OO modeling approaches are classified as semantic modeling methods, there are significant differences between the two. The EER approach tasks the designer to identify and build a model out of basic components (entities, attributes and relationships) that, in aggregate, represent system user views about which one desires to store data. The EER method involves view decomposition and view integration, and is very much a building-block approach to modeling. For example, the CUSTOMER ORDER view is composed of the entities CUSTOMER, ORDER, and PRODUCT, along with various entity attributes and relationships among these three entities. Other views may share some of these entities and attributes, and require the definition of additional relationships.

The OO approach allows the modeling task to be accomplished at the object level; thus, the CUSTOMER ORDER would be modeled as an object. While OO modeling also requires a decomposition and integration approach to modeling, i.e. the CUSTOMER ORDER object would undoubtedly have both CUSTOMER and PRODUCT objects as properties, the OO model may be superior to other modeling methods in terms of model correctness since the representation developed by OO modelers is purported to bear a closer semblance to reality. Recent literature has focused on the promise held by the OO approach to improve modeling success and systems development productivity (Wirfs-Brock and Johnson, 1990). Additionally, OO approaches include the ability to model methods (sometimes termed “services”) as properties of an object. “Methods” are the rules or procedures that operate on the data properties of an object.

This research compares the EER and OO approaches in a controlled laboratory experiment. Success in modeling by entry-level information system professionals is measured by the degree of model correctness. We proceed as follows. Part two discusses related research. Part three provides brief explanations of the two conceptual modeling approaches. The research method is detailed in part four, and the experimental results are given in part five. The concluding section includes a discussion of results, limitations of the study, and recommendations for future research.

**Figure 1: Research Model**

**Related Research**

Recent human-factors information systems (IS) research can be mapped to a research model attributed to Jenkins (1982). Figure 1 gives an adaptation of the Jenkins model by Batra et al. (1990) that aids in framing conceptual data modeling research. Jenkins’ research framework identifies three classes of variables that may interact: system, decision maker, and task. These three variable classes have a causal effect on the dependent variable performance. Batra et al. replace the system variable with the data model variable. They also substitute the variable human for the decision maker variable in Jenkins’ model. In this research, performance is operationalized by model correctness. This measures how well the human designer applies the data model approach to the modeling task in developing an abstract representation of reality.

Early research by Durding et al. (1977) investigated how people organize data by having subjects organize word sets. Durding found that the semantic structure of an application interacts with the data structure and affects how people organize data. This suggests that model correctness is, in part, a function of the structure of data in an application, and that a data model which provides for ease of use in modeling an application may provide a more correct result.

Brose and Schneiderman (1978) used instance diagrams to investigate comprehension, problem solving situation, and memorization tasks as variables in experiments comparing the relational and hierarchical models. Subjects were classed as programmers or non-programmers. Programmers performed best overall, while the hierarchical model was easiest for the non-programmer group to use.

Hoffer (1982) conducted an experiment in which novice subjects developed a conceptual model of an application.
Related Content

Semantic Association Identification and Knowledge Discovery for National Security Applications
[www.igi-global.com/article/semantic-association-identification-knowledge-discovery/3325?camid=4v1a](www.igi-global.com/article/semantic-association-identification-knowledge-discovery/3325?camid=4v1a)

Implementing a Data Mining Solution for an Automobile Insurance Company: Reconciling Theoretical Benefits with Practical Considerations
Ai Cheo Yeo and Kate A. Smith (2006). *Cases on Database Technologies and Applications* (pp. 189-201).
[www.igi-global.com/chapter/implementing-data-mining-solution-automobile/6212?camid=4v1a](www.igi-global.com/chapter/implementing-data-mining-solution-automobile/6212?camid=4v1a)

A Case Study of the Use of the Viable System Model in the Organization of Software Development
Peter Kawalek and David Wastell (2002). *Advanced Topics in Database Research, Volume 1* (pp. 120-134).
[www.igi-global.com/chapter/case-study-use-viable-system/4325?camid=4v1a](www.igi-global.com/chapter/case-study-use-viable-system/4325?camid=4v1a)

Benchmarking Data Mining Algorithms
[www.igi-global.com/article/benchmarking-data-mining-algorithms/3274?camid=4v1a](www.igi-global.com/article/benchmarking-data-mining-algorithms/3274?camid=4v1a)