1988

A COMPARISON OF USER PERFORMANCE BETWEEN THE RELATIONAL AND THE EXTENDED ENTITY RELATIONSHIP MODELS IN THE DISCOVERY PHASE OF DATABASE DESIGN

Dinesh Batra
Indiana University, Bloomington

Jeffrey A. Hofrer
Indiana University, Bloomington

Robert P. Bostrom
University of Georgia

A COMPARISON OF USER PERFORMANCE BETWEEN THE RELATIONAL AND THE EXTENDED ENTITY RELATIONSHIP MODELS IN THE DISCOVERY PHASE OF DATABASE DESIGN[1]

Dinesh Batra
Jeffrey A. Hoffer
Department of Operations and Systems Management
Graduate School of Business
Indiana University, Bloomington

Robert P. Bostrom
Department of Management
University of Georgia

ABSTRACT

This paper reports on a laboratory study which compared conceptual data models developed by casual autonomous users using the relational and the extended entity relationship (EER) representation techniques. It was found that the EER model led to better user performance in modeling binary relationships, while the relational model was better in modeling unary relationships. Subjects found it difficult to model ternary relationships using either model, although the performance using the EER model was slightly better. In general, there was evidence that the EER model led to better user performance. Subjects using the EER model were more confident about their solutions and perceived the model as easier to use than their relational counterparts. The study's results raise questions concerning user performance using the relational model for a discovery (conceptual modeling) task.

1. INTRODUCTION

Currently, commercial database management systems (DBMSs) typically use one of three classical data models: *hierarchical* (IBM 1975; Tsichritzis and Lochovsky 1976), *network* (CODASYL 1971; Taylor and Frank 1976) and *relational* model (Codd 1970). A comparative data manipulation study by Lochovsky and Tsichritzis (1977) suggests that relational systems, as compared to hierarchical and network systems, lead to better user performance as measured by query correctness score. In fact, there has been a proliferation of relational systems in recent years. SQL/DS, INGRES, DBASE III, RBASE, and ORACLE are just a few of the relational systems which are now being extensively used. Most of these are or have a microcomputer version which is, in part, targeted toward novice and casual end-users. Further, most authorities consider a relational DBMS and a non-procedural query language to be a prerequisite of 4GLs (Davis and Olson 1985) which are often used by end-users to develop their own systems.

However, several researchers have noted the inadequacies of these three major data models in their abilities to capture complex relationships between entities. Kent (1979) mentions limitations of record-based information models, including the relational model. Schmid and Swenson (1975) note that the relational theory gives no indication about the way in which the world is to be represented by a collection of relations.

The limitations of the classical models have led to suggestions of semantic data models (Brodie 1984) that are capable of coping with more intricate semantics inherent in many situations. Chen (1976) proposed the *entity-relationship* model which adopts the view that the real world consists of entities and relationships. He also introduced an associated graphic representation technique as a tool for database design. Recently, the E-R model has been extended to include the notion of categories (Elmasri, Hevner and Weeldreyer 1985). This model is appropriately called the *extended entity-relationship* (EER) model. Teorey, Yang and Fry (1986) present the EER model as a logical design tool which can be used to conceptualize data requirements. The EER representation can then be converted to a relational representation (or any other data model) for database implementation. Thus implicitly, these authors make the assumption that the EER model, as compared to the relational model, is the better representation for conceptual design.

Many other semantic models have been proposed (e.g., Smith and Smith 1977a, 1977b; Hammer and McLeod 1981). However, there is little empirical evidence that the semantic models, in conceptual modeling or any other task, lead to better user performance than the classical models. In fact, few human factor studies comparing classical and semantic data representations have been reported in the database literature. To extend our understanding of this issue, we conducted a laboratory study to test if the use of a semantic model, as compared to the

relational model, resulted in a conceptual model which more correctly represented the characteristics of data intensive application domains. The study used the extended entity relationship model (EER) -- a popular semantic data model.

The purpose of this paper is to report the design and results of the study. In the next section, we present the characteristics of users under focus in this study and present a framework of the database design process specifically for this type of user. In section 3, the research framework for the study is presented. Section 4 provides a summary of prior literature on human factor issues in data modeling. The research problem and hypotheses are presented in section 5. The research strategy and design are explained in section 6, the results are presented and discussed in sections 7 and 8, respectively, and the concluding section discusses implications from the study and suggests directions for future research.

2. AUTONOMOUS USER AND DATABASE DESIGN

While new data models have proliferated, there has also been a widespread diffusion of database technology. This technology is now readily available for application to non-trivial problems by users with a range of skills. The recent phenomenon of *end-user computing* (Benjamin 1982; McLean 1979; Rockart and Flannery 1983) has been supported by relational data management technology. In the light of these changes, Davis (1986) defines a new category of users -- *autonomous* -- as users who develop, design, implement, and use application programs in either interactive or personal computing environments to support personal or a small group's information requirement for decision making. Such users possess a moderate amount of computing skills. Davis and Srinivasan (1988) define autonomous mode of usage as essentially characterized by system building/application development using tools that are easily available and learned. Autonomous users are typically *casual* users too. Casual users have been defined, by Card, Moran and Newell (1980), as users who have a moderate knowledge of systems. Everest (1986) defines casual users as users who interact with systems irregularly and occasionally. Cuff (1980) constructs a detailed profile of *casual* users and distinguishes them from regular and committed users.

According to Davis (1986) and others, the category of casual autonomous users is the fastest growing class of users because of increasing levels of computer literacy in society and the availability of inexpensive, easy-to-use computers and software packages. We therefore focused on this important class of users. Since such users will not be information technology experts, they will not be able to effectively use conventional software tools which have been typically designed for expert users and large systems. Since database systems are based on data models, there is a need to better understand which data models are best suited to casual autonomous users.

We now present a framework of the database design process specifically for autonomous users. The major difference between our framework and conventional approaches is that since only one user, or at most a few users, are involved, the concept of view integration is of little importance. Using the terminology from the network model, we may state that there will be little or no difference between a schema and its sub-schemas since all users will have almost the same view of the database. It is assumed that the designer and the user are the same person.

The database design process for an autonomous user involves the following:

> The process of mapping objects of reality in a conceptual model representation constitutes the *discovery phase* (Juhn and Naumann 1985). Since the "designer," in this scenario, may be quite familiar with the application, the elicitation of user requirements may be trivial. In the discovery process, a data model provides representation primitives to aid in the development of a conceptual model. Once the requirements are represented in a conceptual model, the user may be asked to *validate* the requirements. *Validation* of user requirements may be less pertinent in a user developed application. The conceptual model can be implemented using an available Database Management System (DBMS).
>
> First, a *schema* is prepared using the *data definition facilities* and *integrity constraints* of the DBMS. The data definition facilities and integrity constraints of a DBMS are based on the rules imposed by a data model (Tsichritzis and Lochovsky 1982). The user can then use a *database query language* (DBQL) to obtain the desired information or *user reports*. The data manipulation commands of a DBQL depend on operators defined in a data model. The process of query writing begins with an end user needing information from the database. To obtain the information, a user mentally prepares a query plan or *strategy* (Gould and Ascher 1975). In the process of preparing a strategy, the conceptual model and data definition facilities can assist the user in *query planning*. The strategy can then be expressed as a *query code*. This process of *query coding* requires a DBQL which provides the syntax and commands. The query code can finally manipulate the data stored according to the schema to result in the desired information.

This framework chunks the lengthy database design process into smaller phases. This is desirable, since it would be difficult to carry out a single study which could evaluate the effectiveness and ease-of-use of a model for the

complete design process. The study focuses on one of the phases, that is, the *discovery phase*.

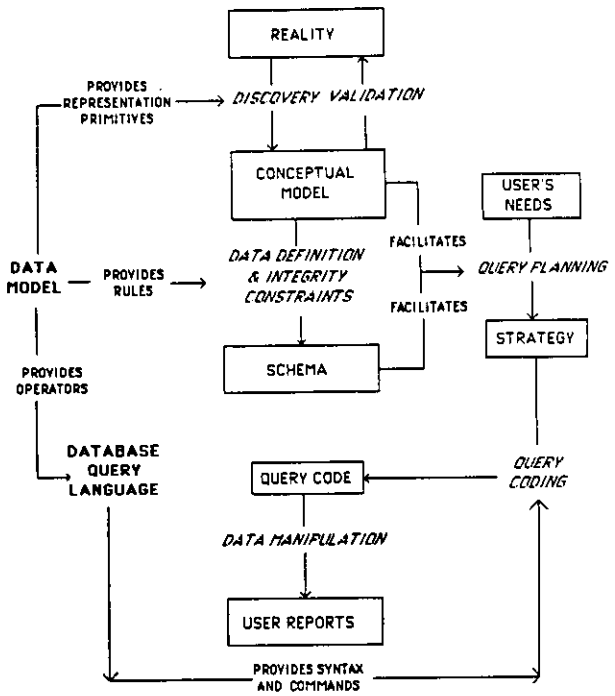

Figure 1. A Framework for Database Design Process

3. RESEARCH FRAMEWORK FOR THE STUDY

The research framework for the study is based on the frequently referenced Jenkins' general model (Jenkins 1982) for human interaction with information systems (Figure 2). The model shows potential relationships between four classes of variables: *system*, *decision maker*, *task*, and *performance*. We have used a specific variation of this research model relevant for data modeling studies. In the new framework, the category *system* is replaced by *data model* and *decision maker* is replaced by *human* (user).

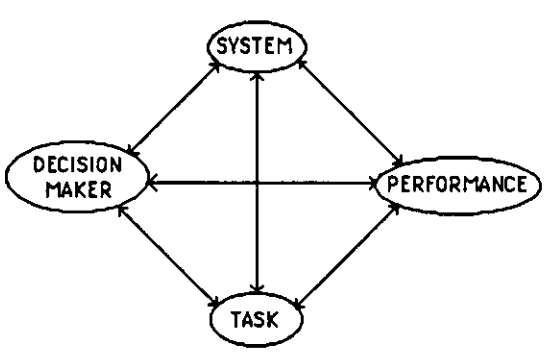

Figure 2. Jenkins' Model

Table 1: Human Factor Studies in Data Modeling

Study	Human	Data Model	Task	Performance
Lochovsky and Tsichritzis (1977)	Experience: Less experience and More Experience	Relational Network Hierarchical	Query Writing	Query Correctness
Brosey and Shneiderman (1978)	Experience: Beginner and Advanced	Relational Hierarchical	Comprehension Problem Solving Memorization	Correctness
Hoffer (1982)	Cognitive Style Situation Familiarity Situation Specificity	Relational Network Hierarchical	Discovery	Database Image Architecture Confidence Number of files
Juhn and Naumann (1985)	Novice/Casual GPA, Computer Experience, DBMS Experience, Work Experience treated as covariates	Logical Data Structure Entity Relationship Data access Diagram Relational	Validation (relationship finding, cardinality finding, identifier comprehension) Database Search Data Modeling (Discovery)	Correctness
Ridjanovic (1986)	Novice/Casual	Logical Data Structure Relational	Discovery	Number of Relationships Number of Attributes
Shoval and Even-Chaime (1987)	Casual	Normalization (Relational) Information Analysis(Binary Relationship)	Discovery	Correctness

Table 2: Framework for Human Factor Studies in Database

Human	Data Model	Task	Performance
Database experience Novice Casual Expert Cognitive Ability Intelligence Memory Reading/Semantic Reasoning Skills Visual Ability Cognitive Style Analytic/Heuristic Field Dependency Locus of Control Preferred Mode of Learning Perceived/Tested Task Knowledge Attitude/Anxiety To computers To Job Situation Familiarity Specificity Demographic Educational Background Work Experience Years of Education Grade Point Average Typing Speed Computer Ownership	Classical Relational Hierarchical Network Semantic Entity Relationship Extended Entity Relationship Binary Relationship Semantic Data Model Semantic Hierarchy Model Logical Data Structure	Discovery Validation Data Definition Data Manipulation Query Writing Query Reading Query Interpretation Query planning Comprehension Memorization Problem Solving	Modeling Correctness Time Learning Database Image Architecture Number of Files Number of Relationships Number of Attributes Behavioral Attitudes Perceived Ease-of-Use Confidence

A survey of the literature on the human factor studies of databases suggests that these four categories can be used to structure an overview of the database human factors research (Table 1). This summary was extended to a general framework for human factor studies of database design and use (Table 2). Besides database studies, we used the following literature to develop the list of possible variables in each category:

1. For the *human* variable, we used the list of individual difference variables in learning of end user software developed by Bostrom, Olfman and Sein (1988).

2. For *data models*, we also included Semantic Hierarchy Model (based on notions of aggregation and generalization in Smith and Smith 1977b) and Semantic Data Model (Hammer and McLeod 1981).

3. For *task*, we used the tasks illustrated in Figure 1, and the tasks suggested in Reisner (1981).

4. For *performance*, we used some of the relevant measures from the list in Jenkins' framework (1982). Further, *perceived ease-of-use* was added from Shneiderman (1980).

4. LITERATURE SURVEY

A survey of human factors studies on databases suggests that most of the literature has focused on programming tasks using database query languages (DBQLs). The interested reader may refer to a survey article by Reisner (1981). However, the focus of our study was on data representation rather than data manipulation. Therefore, we have not included the literature about database query languages and present only the literature relevant to the scope of our study. These studies were listed in Table 1 and are described below.

Lochovsky and Tsichritzis (1977) compared the three classical models: hierarchical, network and relational. Each model was implemented by using a different language: the IMS language DL/I (IBM, 1975), the DBTG COBOL DML (CODASYL, 1971) and ALPHA (Codd, 1971), respectively. Fifty-eight subjects were given query writing tasks. Results showed that for the less experienced users, the relational group scores were significantly better than the other two groups. Although the authors concluded that the relational model was superior, they pointed out that it is difficult to ascribe the results either to the data model or to the language since different models used different query languages.

To overcome the problem of query languages confounding the effects of data models, Brosey and Shneiderman (1978) compared relational and hierarchical models using instance diagrams. Comprehension, problem solving situation, and memorization tasks were performed by undergraduate subjects. Significant effects were found for the data model, presentation order, subject background, and tasks. The hierarchical model was easier to use, but only for the beginning programmer group. The conceptual model used in the experiment was hierarchical in structure, and may have favored the hierarchical model.

Durding, Becker, and Gould (1977) conducted three experiments to investigate how people organize data. This study did not use specific data models (and is, therefore, not included in Table 1). Subjects were given sets of 15 to 20 words and asked to organize them on paper. Each word set had a predefined organization (hierarchy, network, lists, table) based on semantic relations among the words. Results showed that the subjects organized most word sets based on semantic relations inherent in them. These results suggest that the ease-of-use of a model is dependent on the inherent structure of data in an application. However, real world applications are generally a mix of various structures. This study did not, therefore, provide answers to whether any organization approach "in general" was better.

None of the above studies considered individual differences as factors (although Brosey and Shneiderman [1978] did control for possibility of any confounding effects of individual differences by using a within-subjects design.) Hoffer (1982) first reported the result of an investigation of individual differences in using database models. He found that subjects had individualized images of a database and that a process flow structure was the most frequently used image. He also reported that subjects omitted identification of database keys from their images and were not able to clearly specify data relationships. The study considered three types of individual differences: human, situational, and experiential. Greater situation familiarity and more situation structure was found to lead to greater confidence in a database resource. Cognitive styles and programming or other professional experience were not found to be significant factors in influencing a naive user's choice of data model.

Even though these experiments did not provide any distinct conclusions about the relative ease-of-use of the relational *model*, it seems that the ease-of-use of relational *systems* is now widely accepted. The major factor responsible for this is probably the ability of the model to support a non-procedural query interface. Therefore, the relational model has been the focus of many studies. Recent studies have compared the relational model with semantic models.

Juhn and Naumann (1985) focused on the user validation process in database design. They found that the graphic models (entity relationship and logical data structure) were more understandable than the relational and data access diagram in relationship existence finding and cardinality finding tasks. Relational models did outperform graphical models with respect to identifier comprehension tasks. In the data modeling task, the authors found that subjects using the relational model did not follow a systematic modeling process of first identifying entities, then identifying attributes and identifiers of the entities, and finally establishing relationships between entities.

Ridjanovic (1986) conducted a lab experiment using MIS MBA students to investigate differences in the quality of data representations produced by nonexperts using the Logical Data Structure (LDS) and the Relational Data Model (RDM) formalisms. The subjects were asked to read a case, ask questions, and generate application data models which were then evaluated using an instrument developed by the researcher. Results indicated that, contrary to the author's hypotheses, the LDS subjects' questions were not relationship-driven, and the RDM subjects' questions were not attribute-driven. On comparing the

two representations, it was found that there were significant differences in the number of relationships in favor of LDS and in the number of attributes in favor of the RDM group.

Shoval and Even-Chaime (1987) compared two different methods for designing a database schema: *normalization* and *information analysis* (IA). The normalization method is based on the relational data model. The study involved 26 analysts who were trained to use the two methods in conjunction with the structured analysis method of system analysis. There was evidence that the quality of the database schemata designed using normalization was better than that designed using IA, that normalization required less time than IA to perform, and that the analysts preferred normalization. The authors however suggest that the IA model may be more suitable for complex tasks.

Several general observations can be made from this body of research. First, the relational model was used in each study reported in this section. Second, there does seem to be some inconsistency in the results obtained between these studies. This may be because of the different tasks and dependent variables used. Finally, the studies do not report whether equivalent training was imparted to subjects assigned to different data model groups.

Our study builds on and extends the existing literature. It complements Juhn and Naumann's (1985) user validation study by considering the discovery phase in database design. It extends Ridjanovic's (1986) study by considering a more appropriate and comprehensive dependent variable: *modeling correctness*. It also achieves maximum control over confounding variables by considering a fairly homogeneous pool of subjects and providing equivalent training to different treatment groups by using the same examples, identical data modeling concepts, and similar terminology. A pilot study was used to test instrumentation and to estimate task completion time.

5. RESEARCH PROBLEM

5.1 Overview

The research framework used in the study is shown in Figure 3. The main purpose of this study was to compare classical and semantic models. Therefore, specific models had to be selected to represent each type. The *relational* model was chosen as a classical model and the *extended entity relationship* (EER) as a semantic model. The relational model was selected since it is now generally accepted that relational systems lead to significantly better user performance than other conventional systems (Lochovsky and Tsichritzis 1977). Further, the relational model has been the basis for several PC-based DBMSs and other end-user development tools. The EER model (Teorey, Yang and Fry 1986) was selected since it is an extension of ER model which has been widely quoted in the literature, and is the most popular semantic model actually used for logical design by database designers.

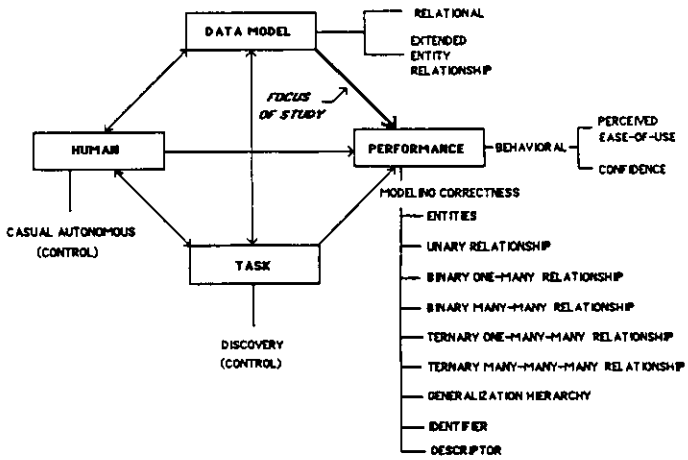

Figure 3. Research Framework for the Study

Since the purpose of the study was to compare user performance between the relational and the EER models in the discovery phase of database design, a *discovery* task was selected. The task required users to read a case and represent the characteristics of data in the form of a conceptual model. The user type was selected based on computer experience. This study focused on *casual* users. Later sections will describe the subjects, task, and training in more depth.

The main performance variable was *modeling correctness*. This was treated as multivariate (Figure 3). The various dimensions of this variable have been termed *facets*. For example, a binary one-many relationship is one facet which may occur in a conceptual data model. A short explanation of the notion of facet is presented below.

A data model may be considered as consisting of various constructs such as entities, relationships, attributes, etc. A construct such as entity requires a fairly consistent set of modeling rules and uniform representation. Essentially, one has to be able to classify an object as an entity or an attribute. However, there is no consistent way of modeling relationships since these may differ in terms of degree and connectivity. Representation of a relationship depends on its degree and connectivity. Hence, it is not appropriate to discuss a conceptual model at the level of relationships; one must qualify the relationships with their degree and connectivity. It is, therefore, pertinent to introduce a construct which is more detailed. This construct is termed a *facet*. *Different instances of a facet have the same representation. Different facets have different representation*. For example, since any instance of a many-many binary relationship is modeled the same way, a *many-many binary relationship* is a facet.

The correctness scores in each of the facets of the conceptual model (e.g., entities, unary relationships, etc.) was graded separately. Thus, the overall modeling correctness score was a vector of scores on various items. This approach was favored since there were serious construct validity concerns with forming a composite score by simply adding scores obtained in individual items.

5.2 Representational Differences

This study was limited to the following data modeling constructs: entity, relationship, category, identifier, and descriptor. It is expected that the reader is familiar with the relational model. The EER model and approach was based on the Teorey, Yang and Fry (1986) paper. It may be noted that the relational representation does not support the category concept. Therefore, a simplified representation of the Smith and Smith (1977b) *generic type* was used.

5.3 Hypotheses

The overall hypothesis of the experiment is that the user performance using the relational model and the extended entity relationship model would be different. Since the dependent variable -- modeling correctness -- is multivariate, we did not hypothesize "higher" or "lower" overall performance for either model.

HM) There will be an overall difference in user performance between the relational model and the entity relationship model.

Further, we list specific hypotheses for individual items:

Entities: Since both the relational and the EER models provide a fairly direct representation of an entity, no difference in user performance for modeling entities was expected.

H1) There will be no difference in user performance in modeling entities between the two models.

Relationships: For representing a relationship and its characteristics, EER provides a direct method, that is, a notation. However, the relational model accomplishes this by associating identifiers of the involved entities. In this study, there were the following kinds of relationships: unary, binary one-many, binary many-many, ternary one-many-many, and ternary many-many-many. For all types of relationships, we predicted better performance using the EER model. Thus the hypotheses were:

The EER model, as compared to the relational model, will lead to better user performance in modeling:

H2) unary relationships.

H3) binary one-many relationships.

H4) binary many-many relationships.

H5) ternary one-many-many relationships.

H6) ternary many-many-many relationships.

These relationship-based hypotheses can be explained by applying the concepts developed by Hutchins, Hollan and Norman (1985) in their model of the human-computer interface. According to this model, there is a *gulf* (or distance) between user's goals and knowledge, and the level of description provided by the systems with which the person must interact. The amount of cognitive effort it takes to manipulate and evaluate a system is directly proportional to this gulf.

In case of the discovery task, the user's goals and knowledge are captured in a representation to produce the conceptual model. As in the Hutchins model, we can identify two different kinds of distances that have to be spanned between the user and the conceptual model: *semantic* and *articulatory distance*. Semantic distance concerns the relationship of the meaning of the conceptual model to user's knowledge of real world data. Between the two models, EER and relational, it was hypothesized that the EER model would facilitate lower semantic distance because it captures the characteristics of the relationships between entities in a more "direct" fashion. The relational model, on the other hand, captures relationships in a more complicated manner and will lead to a larger semantic distance. For example, binary one-many relationships are captured very differently than binary many-many relationships. Articulatory distance is related to the meaning of the conceptual model and its physical form. We noted two reasons why the EER model is likely to lead to lower articulatory distance and better user performance. First, in an EER representation, a relationship is always shown explicitly between the objects. However, in the relational representation, the relationship is represented by associating the identifiers of the objects, and not the objects themselves. Second, since a relationship, by its very definition, is an association between objects, the connection of objects by *graphically connecting* them in an EER representation is a more direct way of showing the relationship.

Generalization Hierarchy: No significant differences were expected in user performance in modeling generalization hierarchies using either of the models. We feel that once the generalization hierarchies are identified, representing them using either of the models requires very little effort.

H7) There will be no difference in user performance in modeling generalization hierarchies between the two models.

Identifier: In either of the models, an identifier serves to uniquely distinguish instances of an entity. However, in the relational model, identifiers are also used to define the relationships between entities. Therefore, we expected better discipline in specifying identifiers using the relational model.

H8) The relational model, as compared to the EER model, will lead to better user performance in specifying identifiers of the respective entities.

Descriptor: Since the representation for a descriptor is straightforward in either models we did not expect any significant differences in the user performance between either model.

H9) There will be no difference in user performance in modeling descriptors between the two models.

The above mentioned hypotheses were the main focus of the experiment. However, two behavioral variables were also considered: *confidence* and *perceived ease-of-use*. It must be admitted, however, that the behavioral variables were not the primary focus in the study and were measured by simple one-item questionnaires. Since the EER model has a more direct approach of modeling relationships, the hypotheses were framed in favor of the EER model:

H10) Users would be more confident about their developed solution using the EER model.

H11) Users would perceive EER model as higher in ease-of-use.

6. RESEARCH STRATEGY AND DESIGN

6.1 Overview

It was decided to conduct a laboratory experiment since it is usually characterized by maximum control and high internal validity (Stone, 1978). A pilot study was conducted in November 1987 with 20 subjects. This study was helpful in exploring the ability of the subjects to prepare conceptual models for nontrivial applications. It also provided useful information on the estimated time for completion of task and estimated time for training and testing of task instructions. A grading scheme was prepared based on the typical errors found for each facet (Appendix A). The focus of the grading scheme was on semantic errors,. However, we did not explicitly separate syntactic and semantic errors in the grading scheme. The study reported here was conducted in February 1988 with a different group of students and an enhanced set of experimental procedures.

6.2 Subjects

Twenty five graduate students were recruited mainly from introductory MIS courses. Most students had taken a programming language course. Others had worked with software such as spreadsheets, wordprocessors, modeling languages, and statistical packages. Some of the students had worked with small databases using DBASE III. No subject had previously designed a nontrivial database. Participation in the experiment was voluntary. No monetary remuneration was given for participation. Subjects could withdraw from the experiment at any stage of the experiment. Two students dropped out of the experiment.

6.3 Procedure

Consenting subjects were provided, a few days before the laboratory session, a short note, "Conceptual Modeling," which introduced them to the basic terminology generally used in database design.

The experiment, which was conducted at the Behavioral Laboratory at the School of Business at Indiana University, had the following sequence:

1. The subjects were asked to complete a questionnaire relating to personal demographics and computer experience.

2. They were then provided with a set of notes and trained by one of the authors for approximately 45 to 50 minutes. These training notes had been prepared for each modeling technique by one of the researchers and reviewed by other researchers and database faculty for completeness and comparability. The pilot study had suggested that the amount of time was adequate, given the complexity of the task. Two examples were used for training subjects, one dealing with sales of products, and the second dealing with instructor development. The training for either treatment group had the same format and length.

3. The subjects were asked to develop, in approximately 30 minutes, a conceptual model for an employee database. They were provided with a textual description of the problem (refer to Appendix B) and were allowed to use the training notes to complete the task. The pilot study had suggested that 30 minutes should be sufficient to complete the task. However, subjects were allowed to take more than the recommended time if necessary.

4. After each subject had finished the task, a debriefing questionnaire was provided to the subject so that s/he could provide feedback and report any ambiguities in the exercise. The questionnaire included one-item questions on the subject's confidence about

the solution and how s/he perceived the ease-of-use of the modeling technique used. The questionnaire also asked the subject to rate the complexity of the task, adequacy of training, and whether s/he enjoyed the experiment.

7. RESULTS

The representation prepared by each subject was graded for correctness by comparing it with the solution developed by the experimenters. The experimenters' solution had the following items: six entities, one unary relationship, one binary relationship with connectivity one-many, one binary relationship with connectivity many-many, one ternary relationship with connectivity one-many-many, one ternary relationship with connectivity many-many-many, two categories based on a single attribute of one of the entities, six identifiers corresponding to the six entities, and twelve descriptors distributed among various entities and relationships. Alternative solutions, if equivalent, were considered correct.

7.1 Grading Scheme

The grading scheme (Appendix A) was designed to provide maximum consistency of scoring between the two models and with the data modeling training. The scheme was developed by one of the researchers and graded by another. The grading was then discussed by the two researchers, any changes proposed by the first researcher were discussed, and a consensus was reached.

Each item was graded separately. A score of 1 was awarded for each correct item and 0 for an incorrect or missing item. Partial credit was given. To facilitate this, errors were classified as minor, medium or major and 0.25, 0.50, and 0.75 were deducted respectively.

7.2 Comparison of Representations

The dependent variable *model correctness* is treated as a multivariate with the various items (entity, etc.) as characteristics. The appropriate statistic for testing differences between the two models, therefore, is the Hotelling T^2 statistic. It was found to be 40.4 with a corresponding F-value of 2.78. The significance level α for testing differences in mean was selected as 0.05. The F-value was found significant at this level and, therefore, the hypothesis of overall differences between the modeling correctness using the two models was supported.

However, the mere significance of the Hotelling T^2 statistic does not show which of the characteristics have contributed to the support of the hypothesis. It is erroneous to carry out univariate t-tests for that purpose because of possible correlations between the various items (Morrison, 1976). Therefore, simultaneous intervals were used to test differences between the individual items. The results are shown in Table 3. The scores were standardized to percentages. For example, if a subject obtained a score of 5 out of 6 in modeling entities, then the score was recorded as 83.3 percent. The mean score in modeling entities is 93.4 percent using the relational model, and 93.5 percent using the extended entity relationship model (see Table 3). This is not significant at 0.05 level, therefore we conclude that there are no significant differences in the score. Thus the hypothesis purported earlier for modeling entities using the two models is supported. The two binary relationships and the ternary relationship with one-many-many connectivity were found to have significant differences, all in favor of the EER model. Hypotheses H1, H3, H4, H7, and H9 were supported, while H2, H5, H6, and H8 were not supported. The hypotheses H10 and H11, on behavioral dependent variables, were also supported.

Table 3: Results of the Study

Hypothesis	Facet	Mean Relational	Mean EER	Confidence Level	Hypothesis Support
H1	Entities	93.4	93.5	1.00	Yes
H2	Unary Rel	62.5	38.6	0.20	No
H3	Binary One-Many Rel	60.4	88.6	0.0034	Yes
H4	Binary Many-Many Rel	54.2	90.9	0.0049	Yes
H5	Ternary One-Many-Many Rel	10.4	29.6	0.079	No
H6	Ternary Many-Many-Many Rel	43.7	27.3	0.45	No
H7	Categories	68.7	90.9	0.079	Yes
H8	Identifiers	79.2	83.3	0.996	No
H9	Descriptors	89.9	91.2	0.999	Yes

Note:

1. The confidence level is based on F test with numerator 9 and denominator 13 degrees of freedom
2. Mean scores are in percentages

Hypothesis	Facet	Mean Relational	Mean EER	Confidence Level	Hypothesis Support
H10	Confidence	5.08	3.45	0.01	Yes
H11	Perceived Ease-of-Use	3.91	2.72	0.04	Yes

Note:

1. The confidence level is based on T test
2. Mean scores are based on ratings on 7-point Likert scale. Lower values indicate higher confidence and perceived ease-of-use.

8. DISCUSSION OF RESULTS

This section discusses differences between scores on individual items obtained for each model (hypotheses H1 thru H11).

Entities (Hypothesis H1): There was no significant difference between the means of the correctness score of entities, and so the hypothesis was supported. In fact, both groups scored high in modeling entities and had almost equal scores.

Unary (Hypothesis H2): This hypothesis was not supported; that is, the EER model did not lead to a significantly higher score than the relational model. In fact, the relational group scored higher by 23.9 percent, although this did not result in significance. We feel the higher score in case of the relational model may be because of a more concrete method of modeling unary relationships. Further, the strict distinction between entities, relationships, and attributes in the EER model implies that there is a greater chance of an error. This was evident from the observation that some EER subjects showed the unary relationship by using an attribute.

In the case given to the subjects, a unary relationship was required to capture the following semantics: "If an employee is married to another employee of Projects Inc., then it is required to store the date of marriage and who is married to whom. However, no record need be maintained if the spouse of an employee is not an employee of the firm." In a relational model, this can be captured by the following representation:

MARRIAGE(EMP#, SPOUSE#, DATE_OF_MAR)

Even though the relationship involves only one entity, an instance of a relationship involves distinct instances of the EMPLOYEE entity. This is more concretely captured by EMP# and SPOUSE# in the relational model. However, in case of the EER model, it is captured by a relationship symbol connected to the same entity. This hides the fact that the relationship is between two distinct instances of the same entity. This was also evident by the fact that some subjects showed SPOUSE as a separate entity and then showed the marriage relationship as binary.

One-Many and Many-Many (Binary) Relationships (Hypotheses H3 and H4): Both hypotheses were supported. The mean score of the EER group was 88.6 for the one-many relationship and 90.9 for the many-many relationship. The corresponding scores for the relational group were 60.41 and 54.17. The results clearly point out the inadequacy of the relational model for capturing binary relationships. The binary relationships are the most frequently occurring relationships in real world applications. This outcome, therefore, is especially significant.

There were many problems with the way the subjects using the relational model captured relationships. First, there was confusion about the connectivity of the relationships. This was possibly due to the fact that, when using the relational model, the connectivity of the relationship dictates if it will be captured explicitly (e.g., many-many), or implicitly (e.g., one-many). Second, it was found that subjects frequently attempted to capture a relationship by using the entity names and not their identifiers.This could have been due to the fact that the relationships are represented by associating the identifiers of the involved entities and not the entities themselves.

One-Many-Many and Many-Many-Many Ternary Relationships (Hypotheses H5 and H6): The scores on ternary relationship with connectivity one-many-many and many-many-many relationship were not found significantly different. However, the most important observation was the sharp fall in the scores in general when the connectivity was changed from binary to ternary. The mean score for the EER group was 29.6 for the one-many-many relationship and 27.3 for the many-many-many relationship. The respective mean scores for the relational group were 10.4 and 43.7.

These results suggest two important points. First, for casual autonomous users, ternary relationships are difficult to model. In fact, we should not expect such users to model relationships of degree higher than 3. Second, relationships where the connectivity is partly one and partly many seem to be more difficult to model. This may be because there are more possible combinations of such cases. For example, there are three possible configurations of a one-many-many relationship, but only one possible configuration of a many-many-many relationship.

The question whether ternary relationships are easier to model using the EER model as compared to the relational was not totally clear from this study, although there was some evidence to support it. The relationship with connectivity one-many-many was close to significance ($p = 0.08$) in favor of the EER model. More empirical work is needed to investigate this issue.

Generalization Categories (Hypothesis H7): There was no significant difference between the mean scores of the two treatment groups, although the EER group did have a higher mean score. It may be mentioned that the category concept has no explicit support in the relational model. The representation to support the category concept was, therefore, devised by the authors.

Identifiers (Hypothesis H8): There was no significant difference between the mean scores of the two treatment groups. This was counter to the hypothesis which predicted higher score for the relational group. In fact, both groups performed very well. This is somewhat contrary to Hoffer's (1982) finding which reported that subjects frequently omitted specifications of identifiers. We feel that the training imparted to the subjects, which stressed specifications of identifiers, was probably responsible for our results.

Descriptors (Hypothesis H9): As hypothesized, there was no significant difference in the mean score between the two treatment groups. In general, neither group had any problem identifying and representing descriptors although there were instances where a descriptor for a relationship was associated with an entity participating in the relationship.

Confidence (Hypothesis H10) and Perceived Ease-of-Use (Hypothesis H11): Both hypotheses were supported. This indicates that, for the category of users considered in the study, the relational model is more difficult to use. It further suggests that the behavioral performance variables, which have been generally neglected, should be included in human factor studies on database design.

Another significant result obtained from the debriefing questionnaire suggests that subjects using the relational model felt that the training was inadequate for the task ($p = 0.02$). This was the case even though the same examples had been used in the training session for both groups. Although not significant, it was also found that subjects using the EER model enjoyed the experiment more than their relational counterparts ($p = .12$). Both groups found the task to be fairly complex.

9. IMPLICATIONS AND FURTHER RESEARCH

The general evidence from the study was that user performance in a discovery task using the EER model, as compared to the relational model, was better. Further, the study suggested that the two modeling approaches -- relational and EER -- lead to significantly different user performance in modeling relationships, but not in other areas. The EER model led to better user performance in modeling binary relationships, while the relational model led to better performance in modeling unary relationships. There was no difference in user performance in modeling ternary relationships. The study also found the user performance in modeling relationships, as compared to other items, was lower. As the degree of the relationship increased from binary to ternary, there was a sharp decline in user performance. Therefore, we can conclude that it is the relationships which mainly contribute to the complexity of a data model. The sharp deterioration of user performance in modeling ternary relationships probably sets limits to the degree of relationships that can be successfully modeled.

Our study provides some clues about whether casual autonomous users can design databases. The performance of the subjects in most items was satisfactory enough to suggest that this is a real possibility. In fact, even the subjects assigned to the relational group did fairly well considering that they had been trained for approximately 45 minutes. Many of the errors, especially minor and medium, found in the study would usually be caught as the database is defined via a DBMS. A follow-up field study is needed to better understand what types of errors or inadequacies are actually implemented by autonomous users.

We presented a general framework for empirical research in data modeling (Table 2). This framework can be used for future research in this area. Similar studies can be designed to test user performance using other semantic models, e.g., Semantic Hierarchy model. Prototype implementations of such models have been developed in laboratories. The effectiveness of these implementations should be empirically verified.

The results from the study also have practical significance. Currently, end users are only trained to use various DBMS software which are generally based on the relational model. However, for effective use of such software, there is a need to train and support users in the discovery and validation tasks (see Figure 1). Our research, along with other findings (Juhn and Naumann 1985) suggests that semantic models, e.g., the EER model, provide better mechanisms to support these tasks. In fact, the developers of DBMS software should consider implementations based on semantic models.

10. ACKNOWLEDGEMENTS

The authors wish to thank Joseph Davis and Ananth Srinivasan for their helpful discussions. We also thank ICIS anonymous referees for their constructive comments.

11. REFERENCES

Benjamin, R. I. "Information Technology in the 1990's: A Long Range Planning Scenario." *MIS Quarterly*, June 1982, pp. 11-32.

Bostrom, R. P.; Olfman, L.; and Sein, M. K. "The Importance of Individual Differences in End-User Training: The Case for Learning Style." *ACM SIGCPR*, Maryland, April 1988, pp. 133-141.

Brodie, M. L. "On the Development of Data Models." In M. L. Brodie, J. Mylopoulos, and J. W. Schmidt, *On Conceptual Modelling*, New York: Springer-Verlag, 1984.

Brosey, M., and Shneiderman, B. "Two Experimental Comparisons of Relational and Hierarchical Database Models." *International Journal of Man-Machine Studies*, 1978, Vol. 10, pp. 625-637.

Card, S.; Moran T. P.; and Newell, A. "The Keystroke-Level Model for User Performance Time with interactive systems." *Communications of the ACM*, July 1980, Vol. 23, No. 7, pp. 396-410.

Chen, P. P. "The Entity-Relationship Model -- Toward a Unified View of Data." *ACM Transactions on Database Systems*, Vol. 1, No. 1, March 1976, pp. 9-36.

CODASYL Data Base Task Group, April 1971 Report, ACM, New York.

Codd, E. F. "A Data Base Sublanguage Founded on the Relational Calculus." In *Proceedings of the ACM*

SIGFIDET Workshop on Data Description, Access and Control, 1971, New York, pp. 35-68.

Codd, E. F. "A Relational Model of Data for Large Shared Data Banks." *Communications of the ACM*, Vol. 13, 1970, pp. 377-387.

Cuff, R. N. "On Casual Users." *International Journal of Man-Machine Studies*, Vol. 12, 1980, pp. 163-187.

Davis, G. B., and Olson, M. H. *Management Information Systems: Conceptual Foundations, Structure and Development*, Second Edition, McGraw-Hill, New York, 1985.

Davis, J. G. "A Typology of Management Information Systems Users and its Implications for Effectiveness Research." Unpublished Doctoral Dissertation, University of Pittsburgh, 1986.

Davis, J. G., and Srinivasan, A. "Incorporating User Diversity into Information Systems Assessment." In N. Bjørn-Andersen and G. B. Davis (eds.), *Information Systems Assessment*, Amsterdam: Elsevier Science Publishers, B.V. (North-Holland), 1988.

Durding, B. M.; Becker, C. A.; and Gould, J. D. "Data Organization." *Human Factors*, Vol. 19, No. 1, 1977, pp. 1-14.

Elmasri, R.; Hevner, A.; and Weeldreyer, J. "The Category Concept: An Extension to the Entity-Relationship Model." *Data Knowledge Engineering*, Vol. 1, No. 1, 1985, pp. 75-116.

Everest, G. C. *Database Management: Objectives, System Functions and Administration*. New York: McGraw-Hill, 1986.

Gould, J. P., and Ascher, R. N. "Use of an IQF-like Query Language by Non-Programmers." *IBM Research Report RC 5279*, Yorktown Heights, New York, 1975 (30 pages).

Hammer, M., and McLeod, D. "Database Description with SDM: A Semantic Database Model." *ACM Transactions on Database Systems*, Vol. 6, No. 3, September 1981, pp. 351-386.

Hoffer, J. A. "An Empirical Investigation into Individual Differences in Database Models." In *Proceedings of the Third International Conference on Information Systems*, Ann Arbor, December 1982, pp. 153-168.

Hutchins, E. L.; Hollan, J. D.; and Norman, D. A. "Direct Manipulation Interfaces." *Human Computer Interaction*, Vol. 1, 1985, pp. 311-338.

IBM Corporation. "IBM Information Management System/Virtual Storage (IMS/VS), General Information Manual." GH20-1260-3, White Plains, N.Y., 1975.

Jenkins, A. M. *MIS Decision Variables and Decision Making Performance*. Ann Arbor: UMI Research Press, 1982.

Juhn, S., and Naumann, J. D. "The Effectiveness of Data Representation Characteristics on User Validation." In *Proceedings of the Sixth International Conference on Information Systems*, Indianapolis, 1985, pp. 212-226.

Kent, W. "Limitations of the Record Based Information Models." *ACM Transactions of Database Systems*, Vol. 4, No. 1, March 1979, pp. 107-131.

Lochovsky F. H., and Tsichritzis, D. C. "User Performance Considerations in DBMS Selection." *Proceedings ACM SIGMOD*, 1977, pp. 128-134.

Maier, D. *The Theory of Relational Databases*. Rockville: Computer Science Press, 1983.

McLean, E. R. "End Users as Applications Developers." *MIS Quarterly*, December 1979, pp. 37-46.

Morrison, D. F. *Multivariate Statistical Methods*. New York: McGraw-Hill, 1976.

Reisner, P. "Human Factor Studies of Database Query Languages." *Computing Surveys*, Vol. 13, No. 1, March 1981, pp. 13-31.

Ridjanovic, D. "Comparing Quality of Data Representations Produced by Nonexperts using Logical Data Structure and Relational Data Models." Unpublished Ph.D. Dissertation, University of Minnesota, 1986.

Rockart, J. F., and Flannery, L. S. "The Management of End User Computing." *Communications of the ACM*, Vol. 26, No. 10, October 1983, pp. 776-784.

Schmid, H. A., and Swenson, J. R. "On the Semantics of the Relational Data Model." *Proceedings 1975 SIGMOD Conference*, San Jose, California, May 1975, pp. 211-223.

Shneiderman, B. *Software Psychology: Human Factors in Computer and Information Systems*. Cambridge: Winthrop Publishers, 1980.

Shoval, P., and Even-Chaime, M. "Database Schema Design: An Experimental Comparison Between Normalization and Information Analysis." *Database*, Vol. 18, No. 3, Spring 1987, pp. 30-39.

Smith, J. M., and Smith, D. C. P. "Database Abstractions: Aggregation." *Communications of the ACM*, Vol. 20, No. 6, June 1977a, pp. 405-413.

Smith, J. M., and Smith, D. C. P. "Database Abstractions: Aggregation and Generalization." *ACM Transactions on Database Systems*, Vol. 2, No. 2, June 1977b, pp. 105-133.

Stone, E. *Research Methods in Organization Behavior.* Glenview, IL: Scott, Foresman & Co., 1978.

Taylor, R. W., and Frank, R. L. "CODASYL Data-Base Management Systems." *ACM Computing Surveys*, Vol. 8, No. 1, 1976, pp. 67-103.

Teorey, T.; Yang, D.; and Fry, J. P. "A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model." *ACM Computing Surveys*, Vol. 18, No. 2, June 1986, pp. 197-222.

Tsichritzis, D. C., and Lochovsky, F. H. *Data Models.* Englewood Cliffs: Prentice-Hall, 1982.

Tsichritzis, D. C., and Lochovsky, F. H. "Hierarchical Data-Base Management: A Survey." *ACM Computing Surveys*, Vol. 8, No. 1, 1976, pp. 105-124.

12. ENDNOTES

1. This paper is under consideration by *Communications of the ACM.*

APPENDIX A

Table 4: Grading Scheme for the Study				
Item	**Incorrect**	**Major Error**	**Medium Error**	**Minor Error**
Entity	Missing Represented as an attribute			Extra Entity
Relationships	Missing Incorrect degree except when alternative representation is plausible	(In EER only) Unary relationship shown by using attribute, and without relationship symbol	Incorrect connectivity but correct degree Unary relationship captured by categories (In relational only) Employing entity names instead of identifiers	No name (In EER only) identifiers mentioned but incorrect
Generalization Categories	Missing		Categories shown but incorrect representation Incorrect identifiers	Missing Identifiers
Identifiers	Missing Identifier different from the one specified in the task description		Attribute not underscored	
Descriptors	Missing Associated with another entity or relationship			

APPENDIX B

EXERCISE

Projects Inc. is an engineering firm with approximately 500 employees. A database is required to keep track of all employees, their skills and projects assigned and departments worked in. Every employee has a unique number assigned by the firm. It is required to store his/her name and date-of-birth. If an employee is currently married to another employee of Projects Inc., then it is required to store the date of marriage and who is married to whom. However, no record of marriage need be maintained if the spouse of an employee is not an employee of the firm. Each employee is given a job title (e.g., engineer, secretary, foreman, etc). We are interested in collecting more data which is specific to the following types: engineer and secretary. The relevant data to be recorded for engineers is the type of degree (e.g., electrical, mechanical, civil, etc.) and for secretaries is their typing speeds. An employee does only one type of job at any given time and we need to retain information material for only the current job for an employee.

There are eleven different departments, each with a unique name. An employee can report to only one department. Each department has a phone number.

To procure various kinds of equipment, each department deals with many vendors. A vendor typically supplies equipment to many departments. It is required to store the name and address of each vendor, and the date of last meeting between a department and a vendor.

Many employees can work on a project. An employee can work in many projects (e.g., Southwest Refinery, California Petrochemicals, etc.), but can only be assigned to at most one project in a given city. For each city, we are interested in its state and population. An employee can have many skills (e.g., preparing material requisitions, checking drawings, etc.), but s/he may use only a given set of skills on a particular project. (For example, an employee MURPHY may prepare requisitions for Southwest Refinery project, and prepare requisitions as well as check drawings for California Petrochemicals.) An employee uses each skill that s/he possesses in at least one project. Each skill is assigned a number. A short description is required to be stored for each skill. Projects are distinguished by project numbers. It is required to store the estimated cost of each project.