1990

A COMPARISON OF THE DATA AGGREGATION APPROACH WITH THE LOGICAL RELATIONAL DESIGN METHODOLOGY

Dinesh Batra
Florida International University

Peeter J. Kirs
Florida International University

A COMPARISON OF THE DATA AGGREGATION APPROACH WITH THE LOGICAL RELATIONAL DESIGN METHODOLOGY

Dinesh Batra
Peeter J. Kirs
Department of Decision Sciences and Information Systems
Florida International University

ABSTRACT

A laboratory study comparing relational representations developed using the Data Aggregation approach with the Logical Relational Design Methodology (LRDM) was conducted to investigate whether non-expert users could better comprehend and apply either methodology. While no significant differences between user performance were noted, the study did find that subjects following the LRDM produced quality Entity-Relationship (ER) representations, but there was a marked deterioration of the translation to the relational form. The Data Aggregation solutions were generally poor in quality. The study concludes that while non-expert designers can produce acceptable data abstractions using a conceptual modeling methodology (e.g., ER diagrams), problems may arise during conversion to normalized relations (e.g., relational representations).

1. INTRODUCTION

Given the potentially complex nature of database design, considerable attention has recently been devoted to the development and refinement of methodologies intended to assist in capturing relationships between entities using direct or natural representations. A variety of conceptual data models have been proposed for the explicit purpose of typifying structures and linkages in a simple and semantically appealing manner, including the entity-relationship (ER) and extended entity-relationship (EER) models (Chen 1976; Elmasri, Weeldreyer and Hevner 1985; Yao 1985; Teorey, Yang and Fry 1986), NIAM (Verheijin and van Bekkum 1982; Nijssen and Halpin 1989), the Database Abstraction (DA) Model (Smith and Smith 1977a, 1977b), and the Semantic Data Model (Hammer and McLeod 1981).

Complicating the issue is the trend toward end-user developed (EUD) activities, including database development. Although EUD projects have been promoted as a means of reducing systems development backlogs and design time (Wetherbe and Leitheiser 1985), encouraging improved problem specification (Peckham et al. 1989), and providing responsive systems (Brancheau and Wetherbe 1987), concerns about the potential risks to organizations have also been expressed (Alavi and Weiss 1986). It has been suggested that improperly directed and managed EUD applications can increase costs and limit effectiveness and efficiency (Cheney, Mann and Amoroso 1986) because of users' lack of expertise. These risks seem especially acute for applications requiring technical knowledge and experience, such as database design.

Consequently, an important concern is whether non-expert database designers can better comprehend and apply any specific methodology. While a few studies have compared the usability of data models using non-expert subjects, the outcomes are in need of further investigation. The purpose of this exploratory study is to examine the quality of relational database structures developed by non-expert end users using Data Aggregation (DA) Concepts. By way of contrast, this paper focuses on the efficacy of the DA versus the Logical Relational Design Methodology (LRDM) as a means of transposing abstract relationships into relational database representations.

2. PROBLEM BACKGROUND

Traditionally, relational database design has relied on low-level, bottom-up approaches intended to construct normalized relations using inter-data element dependencies. However, several authors have observed that as the scale of the database or information structure expands and the number and complexity of relationships increases, the overall structure can become obscured to even experienced analysts. In response, top-down conceptual modeling approaches have been suggested and examined as a means increasing problem understanding, communication of requirements, and as a framework for transforming component elements into normalized relations.

The concept of *Data Aggregation* (DA) has become an important feature of semantic data models (Peckham and Maryanski 1988). Initially proposed by Smith and Smith (1977a), it facilitates abstraction by allowing a relationship between objects to be viewed as an object in itself. In the same paper, the authors showed how this concept could be used to develop a conceptual data model which can then be converted to a relational representation.

While the usability of the DA model has not been empirically tested in any known prior study, the notion is intuitively appealing since it is an abstraction common in everyday usage. For example, a reservation is an abstract concept of a person, a hotel, a room and a date. The two levels of abstraction (person, room, hotel, and date on one level; reservation at a higher level relating the three objects) allow individuals to refer to the relationship between components as an abstract concept. If data abstraction is indeed a natural concept, then a model guided by this notion should be easy to understand and use and should reduce the number of data modeling errors.

Although there are arguments in favor of DA, deficiencies are also apparent. Merely expressing an object as an aggregation of certain objects may not suffice as a database design approach. There may be semantic constraints between the component objects which need to be captured and which may affect the specification of the identifier of the aggregate object. For example, unless the object RESERVATION has its own identifier (e.g., REF#), issues of semantic constraint between the identifiers of the participating objects (PERSON, ROOM, HOTEL and DATE) must still be resolved. If RESERVATION is then translated to a relational representation, these constraints would then be introduced as functional dependencies. For some conceptual data models (e.g., ER), such problems do not arise since semantic constraints can be captured by representing the connectivity of the relationship.

Thus, while the DA approach may initially provide a mechanism to structure data via an aggregate object, there is still reliance on the relational model to capture the semantic constraints between the objects constituting the aggregate object. Past research (e.g., Batra, Hoffer and Bostrom 1990) suggests that the relational model is not an effective conceptual modeling tool. Whether the implied advantages outweigh the implicit disadvantages of the DA approach remains to be resolved through empirical investigation.

To test the usability of the DA approach, we compared it with the Logical Relational Design Methodology (LRDM) proposed and extended by Teorey, Yang and Fry (1986). The approach involves the conceptualization of data requirements as an EER model which is subsequently converted to a relational database representation. One group of subjects followed this procedure using ER diagrams, while the other group developed corresponding DA models. Both groups then translated their representations to a relational database structure. The relational representations derived via the approaches were then graded according to a prescribed scheme and compared against a "correct" solution.

3. DIFFERENCES BETWEEN THE ER AND DA MODELS

There are two major differences between the ER and DA models. First, the ER model treats *entity* and *relationship* as separate concepts, while the DA model treats both as *objects* (either primitive or aggregate). While Chen (1976) mentioned that under certain circumstances a relationship may be treated as an entity and that this decision depends on the enterprise administrator, we adopt the view that this should be done only if the relationship has its own identifier, as suggested by Teorey, Yang and Fry (1986). For example, if VENDOR and PRODUCT are two entities, then the relationship between them, SUPPLY, should be treated as an entity *only* if SUPPLY has its own identifier (e.g., ORDER#). The DA model, on the other hand, does not make a distinction between entity and relationship: both are treated as objects and have the same representation. The relationship is viewed as merely an aggregate object at higher level than the objects participating in the primary relationship. In this example, SUPPLY is a higher-level aggregate object which associates with the objects VENDOR and PRODUCT (both at the same level below SUPPLY). The DA representation of the situation is presented in Figure 1a; the corresponding ER representation is shown in Figure 1b.

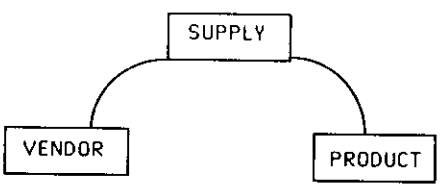


Figure 1a. DA Representation

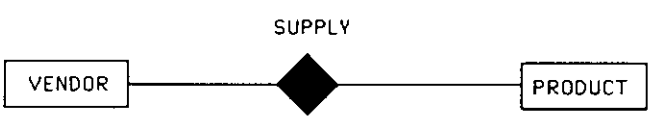


Figure 1b. ER Representation

The next major difference between models lies with the inability of the DA model to unambiguously represent the *connectivity* of a relationship. The connectivity of a relationship indicates how instances of one entity are mapped to another. For example, SUPPLY is a binary relationship (i.e., of degree 2) linking the two entities VENDOR and PRODUCT. The value of the connectivity can be "one" or "many." If a VENDOR can supply many PRODUCTs, but a PRODUCT can be supplied by no more than one VENDOR, the connectivity of SUPPLY is "one" to "many." If a VENDOR supplies many PRODUCTs, and a PRODUCT is supplied by many VENDORs, the connectivity of SUPPLY is "many" to "many." Connectivity implies certain semantic constraints. For example, a one-to-many connectivity indicates that, corresponding to an instance of an object on the many side, there is no more than one instance of the object on the one side. The DA model, however, may not fully convey such semantics in certain situations since the value of connectivity for the relationship cannot clearly represent the aggregate object. The example below illustrates this point.

Consider the aggregate object SUPPLY. If vendors supply products to customers, these semantics can be captured by representing the relationship between SUPPLY and CUSTOMER as a higher level object, such as SALE (see Figure 2a). The connectivity of CUSTOMER may be assumed to be "many" since many customers may purchase the same product shipped by the same vendor. The connectivity of SUPPLY in SALE may similarly be assumed to be "many" if a customer purchases many "supplies."

If, however, there exists a constraint that a customer buy a given product from only one vendor, the above representation of SALE is still valid but is inadequate because SALE is viewed as a binary many-many relationship between SUPPLY and CUSTOMER and is not directly linked to VENDOR and PRODUCT. To represent this constraint, SALE should be viewed as a ternary (degree 3) relationship between VENDOR, PRODUCT and CUSTOMER where the connectivity of VENDOR is *one* and that of PRODUCT and CUSTOMER is *many* (see Figure 2b). This constraint cannot be inferred from Figure 2a. In fact, based on Figure 2a, one is likely to infer that the connectivity of the constraint SALE is many-many-many.

Operationally, the designer has the option to treat SALE as an aggregate of VENDOR, PRODUCT and CUSTOMER, or as an aggregate of SUPPLY and CUSTOMER. However, in the latter case, it may not be possible to capture certain semantic constraints related to the connectivity of the aggregate object SALE. On the other hand, the ER representation (see Figure 2c) unambiguously conveys these semantics since SALE can only be defined as a relationship between the three entities.

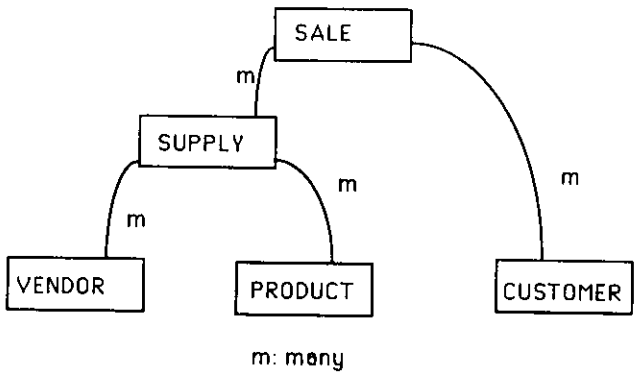


Figure 2a. DA Representation

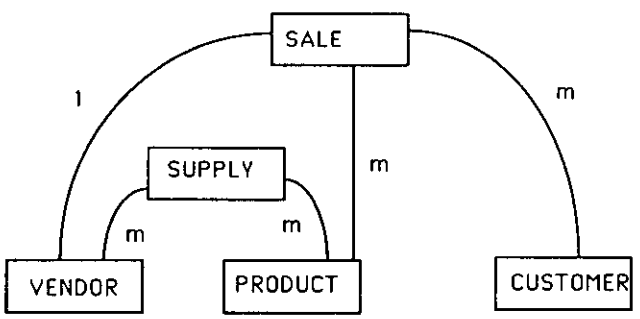


Figure 2b. DA Representation

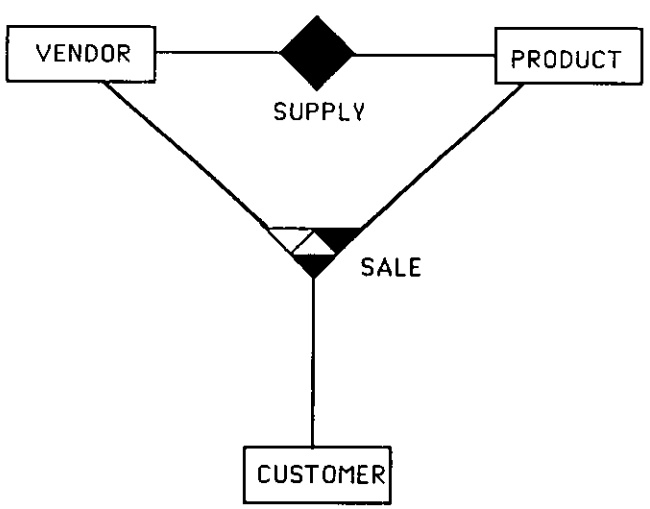


Figure 2c. ER Representation

To examine the issue of connectivity and its impact on model translation, an experimental study was conducted. Subjects were asked to prepare a relational representa-

tion via the ER or the DA representation. The ER subjects were instructed to show the connectivity of the relationships directly in the ER diagrams. For reasons mentioned earlier, the DA subjects were instructed not to show the connectivity of the aggregate objects.

4. RESEARCH METHODOLOGY

There are two reasons why the ER and DA model were used as intermediate and not final representations. First, the connectivity of a relationship is an important semantic constraint. Unlike the ER model, which can capture connectivity directly (see Teorey, Yang and Fry 1986, for a review of the ER approach and its transformation to the relational representation), the DA model cannot. Therefore, for comparison purposes both representations were later converted to the relational model which supports connectivity. The relational model captures these semantics by way of functional dependencies between the identifiers of the associated object. For example, a one to many relationship between VENDOR and PRODUCT may be represented as PROD# → VENDOR#, and a many to many relationship as PROD#,VENDOR# → 0. The determinants in these dependencies can be used as candidate keys in relations.

The second reason refers to the practical utility of the study. The majority of database management systems (DBMS) available to end users are based on the relational model, and a commercial DBMS based on the semantic model does not seem likely in the near future. Thus, conversion of semantic representations to the relational form not only provides a common medium for comparison and evaluation, it also simulates the design process of a user employing an intermediate data model to develop a relational representation.

4.1 Research Model

This study explores the effect of the independent variable, data modeling technique, on the dependent variable user performance. Other variables possibly affecting the dependent variable were either controlled (task, trainer differences, instructional examples) or otherwise randomized (database design experience, other individual differences).

The choice of the data models (DA versus ER), was governed by past findings in this line of research which suggest that user performance in conceptual modeling tasks using the relational data model, as compared to ER model, is generally inferior (Juhn and Naumann 1985; Batra, Hoffer and Bostrom 1990). As an extension, it seemed logical to compare an ER-based data modeling approach with other approaches. The two semantic data models which have received the most attention are the Database Aggregation (Smith and Smith 1977a) and Generalization Model (Smith and Smith 1977b) and the Semantic Data Model (Hammer and McLeod 1981). We did not choose the latter because it was deemed too difficult for non-expert users.

4.2 Research Questions

Developing a conceptual data model essentially involves identifying and representing the entities (or objects), relationships between entities, and entity attributes (i.e., identifiers and descriptors). Since the representations of an entity and its attributes by the two data models considered in this study are similar and fairly straightforward, there was no motivation to investigate differences in the user performance using the two approaches to model these constructs. However, as discussed earlier, the two models vary in the manner in which they represent relationships. Therefore, this study focused on the differences in user performance between the ER and the DA approaches in modeling different kinds of relationships.

Four types of relationships were considered:

- *Binary* (One-Many)
- *Binary* (Many-Many)
- *Ternary* (One-Many-Many)
- *Ternary* (Many-Many-Many)

The direction of relationship between data model and user performance was not hypothesized. However, we did expect subjects using the DA approach to experience difficulties in modeling ternary relationships since such relationships actually capture constraints and not relationships *per se*, and cannot always be readily named. It is unlikely that a relationship will be modeled if it cannot be named. For example, the assignment of SKILLs of EMPLOYEEs to PROJECTs is a ternary fact *only* if employees do not use all their skills in each of the projects to which they are assigned; otherwise, in the spirit of the fourth normal form, it should be captured as two binary facts. The ternary relationship does not explicitly have a name associated with it, so one has to be devised (e.g., EMP-SKILLS-IN-PROJ). Additionally, since the connectivity is not shown in the DA representation, the determination of an identifier for the ternary relationship is also postponed until it is transformed to a relational representation. Past studies have already shown that subjects have problems in modeling ternary relationships using the relational model.

4.3 Subjects

Thirty-eight undergraduate MIS students participated in the study. The subjects were in the fourth week of a required course in database applications. Each had completed the necessary prerequisites consisting of an introductory course in MIS and a course in Systems Analysis and Design. One week prior to the experiment the students received a two and one-half hour lecture on relational data structures.

4.4 Treatments

Subjects were randomly assigned to one of two treatment groups: a Data Aggregation (DA) Group or an Entity-Relationship (ER) Group. Each group consisted of nineteen subjects. Each group was unaware of the treatment given to the other group. In order to promote motivation, students received credit on their final course grades for participation.

To determine whether prior learning and experience might be intervening variables, a survey questionnaire was administered. Simple t-tests were used to compare the two groups. The findings indicate that there were no differences between the groups with respect to prior training, experience, and/or familiarity with information systems concepts and techniques in general, and database design issues in specific. The typical subject had not completed any database course and, although the average rating of experience with database design was 3.5 (on a seven-point scale), later questioning revealed that the rating was based primarily on limited usage of a DBMS, not on actual involvement in the design of a database. Consequently, it seems reasonable to classify the subjects as "non-expert users."

4.5 Training Sessions

The groups received similar, but separate, training in one of the data modeling techniques. Each student was given a training script to be followed during the session. The same instructor conducted both training sessions and adhered to the script as closely as possible. Each script contained the same examples (similar to that given later as the experimental task), presented in increasing order of complexity, which were explained and diagrammed during the session. The scripts varied in the terminology used (e.g., Entities and Relationships in the ER script; Objects and Aggregate Objects in the DA Script), in the figures given as solutions to the sample problems (corresponding to the data modeling technique used), and in the general approach. Both scripts contained examples of binary and ternary relationships and illustrated the mapping between the diagram and relational model. The ER session lasted 75 minutes while the DA session lasted 70 minutes, the difference attributable to the questions asked by the participants.

4.6 Experimental Task

Both groups received the same experimental task (see Appendix A). The subjects were instructed to read the case, diagram the relationships using the assigned data modeling technique, and convert the models to relational databases. The case problem was stated in the same terminology and format as the training script examples, but was deemed to be more difficult.

Subjects diagramed their solutions on separate pieces of paper which were numbered sequentially. Erasures were discouraged unless absolutely necessary; the students were requested to make modifications as separate drawings in order to represent the modeling process. When they were satisfied with their diagram, they labeled it as "Final" and proceeded to the translation of the diagram into the relational database model.

All student material was collected at the end of the session. Each submission contained starting and ending times (verified by the instructor upon receipt) as well as a student identifier and required labels. The entire experiment took approximately two hours and fifteen minutes to complete. Immediately following the experiment, each student completed a multi-item questionnaire designed to identify any differences between the presentation of the material, the scripts used, and the data modeling technique employed. Differences in responses between the two treatment groups were again analyzed using simple t-tests. The results indicate that there were no differences with respect to trainer presentation, script clarity and instructional value, and perceived usefulness of the modeling technique.

5. RESULTS

The representations developed by the subjects were graded for correctness by one of the authors according to prespecified guidelines (Appendix E). The ER, DA and relational solutions are shown in Appendix B, C and D, respectively. For the ER group, the ER and relational representations (LRDM) were graded. The DA solutions were not graded since the DA representations did not capture connectivity information and grading schemes consistent with those for the ER model could not be applied. Only the relational representations developed by the DA subjects were evaluated. For the sake of convention, the relational representation prepared via the DA representation has been termed as DA-REL representation.

Table 1. Representation Conversion Performance

Facet	Mean DA-REL*	Mean LCDM	Sig. Level
Binary (One-Many) Relationships	48.7	51.4	0.85
Binary (Many-Many) Relationships	64.5	58.3	0.68
Ternary (One-Many-Many) Relationships	36.8	33.3	0.77
Ternary (Many-Many-Many) Relationships	21.1	27.8	0.56

*DA-REL refers to the relational representation developed via the DA

Table 2. ER Versus LRDM Representations

Facet	Mean ER	Mean LCDM	Sig. Level
Binary (One-Many) Relationships	81.6	51.4	0.027*
Binary (Many-Many) Relationships	92.1	58.3	0.009**
Ternary (One-Many-Many) Relationships	52.6	33.3	0.096
Ternary (Many-Many-Many) Relationships	44.7	27.9	0.055

* $p \leq 0.05$
** $p \leq 0.01$

Group scores were contrasted using ANOVA. Four sets of comparisons were made:

1. The relational representations developed via the LRDM and the DA model (DA-REL);
2. The ER model and the relational representation developed via the ER approach (LRDM);
3. The ER model and the relational representation developed via the DA approach (DA-REL); and
4. The DA model and the relational representation developed via the DA approach (DA-REL). Since DA representation was not graded, this comparison was qualitative only.

5.1 LRDM versus DA-REL

It was found that the relational representations developed by translating the ER and the DA models (ER-REL versus DA-REL were of similar quality). The mean modeling scores for each of the five facets considered did not vary significantly (Table 1). It was somewhat surprising that the errors were fairly similar. For example, the most commonly occurring errors in modeling binary relationships, regardless of methodology, were missing relationships and connectivity errors. In the case of the ternary relationships, incorrect degree specification and connectivity errors were prevalent. The mean scores for binary relationships (both one-many and many-many) were observed to be distinctly higher than for ternary relationships, although no formal comparison was done.

5.2 ER Versus LRDM

The comparison of the ER with LRDM representations was quite interesting (Table 2). While it was expected that ER scores would be higher than LRDM since errors could be introduced during the translation process, it was found that there was a significant loss in performance when the ER representations were converted to the relational counterparts. It seems likely that the translation from ER to LRDM was not viewed as a mechanical process. As noted in Table 2, significant differences were noted between scores for the binary one-many ($p=0.027$) and the binary many-many ($p=0.009$) relationships. Significant differences were not noted for ternary associations, although the ternary many-many-many relationship ($p=0.055$) approached significance. A larger sample size may be necessary to resolve this question. Given the significant drop in performance from the ER representations to LRDM representations, the errors introduced during the translation are discussed separately.

5.2.1 Binary One-Many and Binary Many-Many Relationships

A common mistake found in the relational solutions was the incorrect representation of connectivity. The ER solutions generally showed the correct connectivity, suggesting that subjects were aware of the concept of connectivity. The major problem was the representation of the same concept in the relational form. It seems that the subjects could not properly associate the notion of connectivity in the ER model with that of dependency in the relational model. For example, in a one to many relationship between two entities, subjects appeared to have difficulties inferring that the identifier of the entity on the "one" side was functionally dependent on the identifier on the "many" side, and that the identifier of the relationship was the identifier of the entity on the "many" side. Some subjects showed a concatenated key (suggesting a many to many relationship) or showed a separate relation without any primary key.

Another interesting finding was that a few subjects did show the relationships correctly in the ER form, but did not develop corresponding relations for them. On closer inspection, it was found that some of the subjects had not named these relationships in the ER representation. For example, a few subjects modeled the relationship between the entities EMPLOYEE and DEPARTMENT in the ER representation but did not assign a name to the relationship. During the translation to the relational representation, the relationship was ignored. It was also interesting that some subjects did not integrate the relations for entities which had common identifiers. For example, some subjects developed the following two relations:

EMPLOYEE (EMP#, <employee attributes>)
BELONGS (EMP#, DEPT-NAME)

The first relation corresponds to one entity and the second to a relationship. These relations should clearly have been merged. Subjects were shown how to integrate relations during the training session, but it seems that the distinction made between an entity and a relationship in the ER model obscured the integration process required for relational representation construction.

5.2.2 Ternary One-Many-Many and Many-Many-Many Relationships

Many of the errors noted in the binary relationships were also found in the ternary relationships. Some subjects could not translate the connectivity in the ER model as functional dependencies and identifiers in the relational model. One type of error, also found in binary relationships but with less frequency, was the absence of relations corresponding to unnamed relationships in the ER representation. While the inability of subjects to create such relations cannot be entirely attributable to the lack of discipline in naming the corresponding relationships, the simultaneity of the occurrence in the ER model and the absence of the corresponding relations in the relational model was noteworthy.

5.3 ER Versus DA-REL

This comparison was performed primarily for the sake of completeness. The mean scores and significance levels are listed in Table 3. Since the LRDM and DA-REL performances were so similar, the results predictably correspond to the ER versus LRDM comparison;that is, the ER scores were higher in all four cases than the DA-REL scores.

5.4 DA Versus DA-REL

For reasons mentioned earlier, the DA representation was not graded. The comparison between DA and DA-REL is qualitative only. The DA representations, however, seemed very poor in quality. It seemed that subjects did not use the DA representations to assist them in preparing the relational representation.

It was found that subjects had problems constructing relationships as aggregate objects. For example, the ternary fact EMP-PROJECT-SKILL (which can also be termed as EMP-SKILL- ASSIGNED-TO-PROJ) can be modeled as an aggregate of the EMPLOYEE, SKILL and PROJECT objects, or as aggregate of EMP-SKILL (which itself is an aggregate) and PROJECT, or in other ways. The plethora of modeling choices would suggest that subjects would find it easy to model the ternary fact. The results seem to contradict this inference; the numerous choices led to confusion. There were a number of errors in modeling these types of relationships. For example, some subjects modeled PROJECT as an aggregate of EMPLOYEE and SKILLS, others showed SKILL-USED as an aggregate of SKILL and TASK, where TASK was shown as an aggregate of EMPLOYEE, PROJECT and CITY.

6. DISCUSSION AND IMPLICATIONS

This study was an extension of the Batra, Hoffer and Bostrom (1990) study. The following observations are based on the quantitative and qualitative solution analysis.

Table 3. ER Versus DA-REL

Facet	Mean DA-REL	Mean ER	Sig. Level
Binary (One-Many) Relationships	48.7	81.6	0.020*
Binary (Many-Many) Relationships	64.5	92.1	0.022*
Ternary (One-Many-Many) Relationships	36.8	52.6	0.054
Ternary (Many-Many-Many) Relationships	21.1	44.7	0.087

* $p \leq 0.05$

- *Even for non-experts, the representation of a data modeling situation can lead to high performance provided a suitable conceptual modeling approach is used.* This study suggests that novices can learn the ER data model quickly and develop high quality solutions. It is assumed that the semantics of the situation have first been determined by effective elicitation techniques. The results obtained are fairly consistent with the Batra, Hoffer and Bostrom study.

- *The translation from one representation to another may not be "mechanical" for a non-expert designer.* The study by Batra, Hoffer and Bostrom concluded that the extended entity relationship (EER) representation was superior to the relational representation. However, the difficulties encountered in the present study in the translation of the ER solution to the relational solution raises questions about the utility of the previous study. The subjects could not readily extend the concept of connectivity into functional dependencies and identifiers. This finding has many implications. First, subjects do need considerable training in establishing associations between the connectivity of the relationships and the identifiers of the relations for these relationships. Standard pedagogy emphasizes training of ER and relational data models but not the equivalence and connections between the two. Second, it may be preferable to have an automated tool that takes the ER representation as input and produces the relational representation as output.

- *Subjects have a strong tendency to associate records/relations with entities, but not with relationships.* The lack of a "forcing" mechanism to name relationships may have been the reason that a number of subjects could not translate the ER relationships into relations. Thus, non-expert designers may perceive relationships as constructs which are important only at the conceptual data modeling stage but which can be discarded at the implementation phase. This finding also has many implications. Standard pedagogy should not only emphasize relationships at the conceptual data modeling stage, but also at the "translation" stage. This is especially important if the implementation is to be a relational model which uses foreign and concatenated keys. Naming a relationship should also be stressed and, in fact, can be forced if a CASE tool is being used. As suggested above, the automated tool could perform the translation to the relational representation.

- *The concept of an aggregate object representing a relationship for general business situations is confusing to a non-expert designer.* In this study, subjects generally developed poor quality DA representations. In particular, relationships were usually not shown as objects. Further, subjects did not seem to use DA representations to aid them in developing the relational representations. Although the present study considered only one problem, there are indications that there is no motivation to use DA as a conceptual modeling methodology for most business situations. The aggregation concept may, therefore, be useful only in certain domains where other approaches (e.g., object oriented data models) may be appropriate.

- *The Relational Model is not totally suitable for top-down analysis.* This study, as well as the study by Batra, Hoffer and Bostrom, arrive at the same conclusion: users tend to perform poorly in a conceptual modeling task using a relational model. However, the task description in both studies states a problem using natural language. It might be interesting to observe if and how these findings change if the problem were reframed as a case description which includes tabular user views along with the natural language description (e.g., similar to the cases found in McFadden and Hoffer [1989]).

7. CONCLUSIONS AND DIRECTIONS FOR FUTURE RESEARCH

This study reinforces the desirability of the ER data model for conceptual modeling performed in a top down fashion. For the novice designer, however, the translation of the ER representation to the relational representation is not a mechanical or trivial step. Relatively, the data aggregation concepts rank low in usability.

Future research can take various directions. An interesting extension to this study would be to examine the effect of feedback about the connectivity of a relationship from a computerized design aid. A novice designer might be asked to input the relations, one at a time, with the design aid interpreting the connectivity for the designer. Results from the study discussed in this paper suggest that such a mechanism might reduce the mismatch between the connectivity concepts in the ER model to the dependency concepts in the relational model. Other extensions to the research can be done by evaluating the usability of other data models. For example, the Semantic Data model (Hammer and McLeod 1981), the Functional Data Model (Shipman 1981), and Codd's (1979) extended relational model RM/T. Additional investigation is also needed with respect to the impact of possible intervening variables, such as task complexity, user characteristics, and application setting.

8. ACKNOWLEDGMENTS

The authors would like to thank the three anonymous referees for their comments on the original submission.

9. REFERENCES

Alavi, M., and Weiss, I. R. "Managing the Risks Associated with End-User Computing," *Journal of Management Information Systems*, Volume 2, Number 3, Winter 1985-86, pp. 5-20.

Batra, D.; Hoffer, J. A.; and Bostrom, R. P. "Comparing Representations with Relational and EER Models," *Communications of the ACM*, Volume 33, Number 2, February 1990, pp. 126-139.

Brancheau, J. C., and Wetherbe, J. C. "Key Issues in Information Systems Development," *MIS Quarterly*, Volume 11, Number 1, March 1987, pp. 23-45.

Chen, P. P. "The Entity-Relationship Model — Toward a Unified View of Data," *ACM Transactions in Database Systems*, Volume 1, Number 1, March 1976, pp. 9-36.

Cheney, P. H.; Mann, R. I.; and Amoroso, D. L. "Organizational Factors Affecting the Success of End-User Computing," *Journal of Management Information Systems*, Volume 3, Number 1, Summer 1986, pp. 65-80.

Codd, E. "Extending the Database Relational Model to Capture More Meaning," *Transactions on Database Systems*, Volume 4, Number 4, December 1979.

Elmasri, R.; Weeldreyer, J.; and Hevner, A. "The Category Concept: An Extension to the Entity-Relationship Model," *Data Knowledge Engineering*, Volume 1, Number 11, June 1985, pp. 75-116.

Hammer, M., and McLeod, D. "Database Description with SDM: A Semantic Datamodel," *ACM Transactions on Database Systems*, Volume 6, Number 3, September 1981, pp. 351-386.

Juhn, S., and Naumann, J. D. "The Effectiveness of Data Representation Characteristics on User Validation." In L. Gallegos, R. Welke, and J. Wetherbe (eds.), *Proceedings of the Sixth International Conference on Information Systems*, Indianapolis, Indiana, December 1985, pp. 212-226.

McFadden, F. R., and Hoffer, J. A. *Data Base Management*. Menlo Park, California: Benjamin/Cummings Publishing Co., Inc., 1988.

Nijssen, G. M., and Halpin, T. A. *Conceptual Schema and Relational Database Design: A Fact Based Approach*. Englewood Cliffs, New Jersey: Prentice-Hall, 1989.

Peckham, J., and Maryanski, F. "Semantic Data Models," *ACM Computing Surveys*, Volume 20, Number 3, September 1988, pp. 153-189.

Peckham, J.; Maryanski, F.; Beshers, G.; Chapman, H.; and Demurjian, S. A. "Constraint Based Analysis of Database Propagation." In J. I. DeGross, J. C. Henderson, and B. R. Konsynski (eds.), *Proceedings of the Tenth International Conference on Information Systems*, Boston, Massachusetts, December 4-6 1989, pp. 9-18.

Shipman, D. "The Functional Data Model and the Data Language DAPLEX," *ACM Transactions on Database Systems*, Volume 6, Number 1, March 1981, pp. 140-173.

Smith, J., and Smith, D. C. P. "Database Abstractions: Aggregation," *Communications of the ACM*, Volume 20, Number 6, June 1977a, pp. 405-413.

Smith, J., and Smith, D. C. P. "Database Abstractions: Aggregation and Generalization," *ACM Transactions on Database Systems*, Volume 2, Number 2, June 1977b, pp. 105-133.

Teorey, T. J.; Yang, D.; and Fry, J. P. "A Logical Design for Methodology for Relational Databases Using the Extended Entity-Relationship Model," *ACM Computing Surveys*, Volume 18, Number 2, June 1986.

Teorey, T. J.; Wei, G.; Bolton, D. L.; and Koenig, J. A. "ER Model Clustering as an Aid for User Communication and Documentation in Database Design," *Communications of the ACM*, Volume 32, Number 8, August 1989, pp. 975-987.

Verheijen, G., and van Bekkum, J. "NIAM: An Information Analysis Method." In T. W. Olle, H. G. Sol, and A. A. Verryn-Stuart (eds.), *Information Systems Design Methodologies*. Amsterdam: North Holland, 1982.

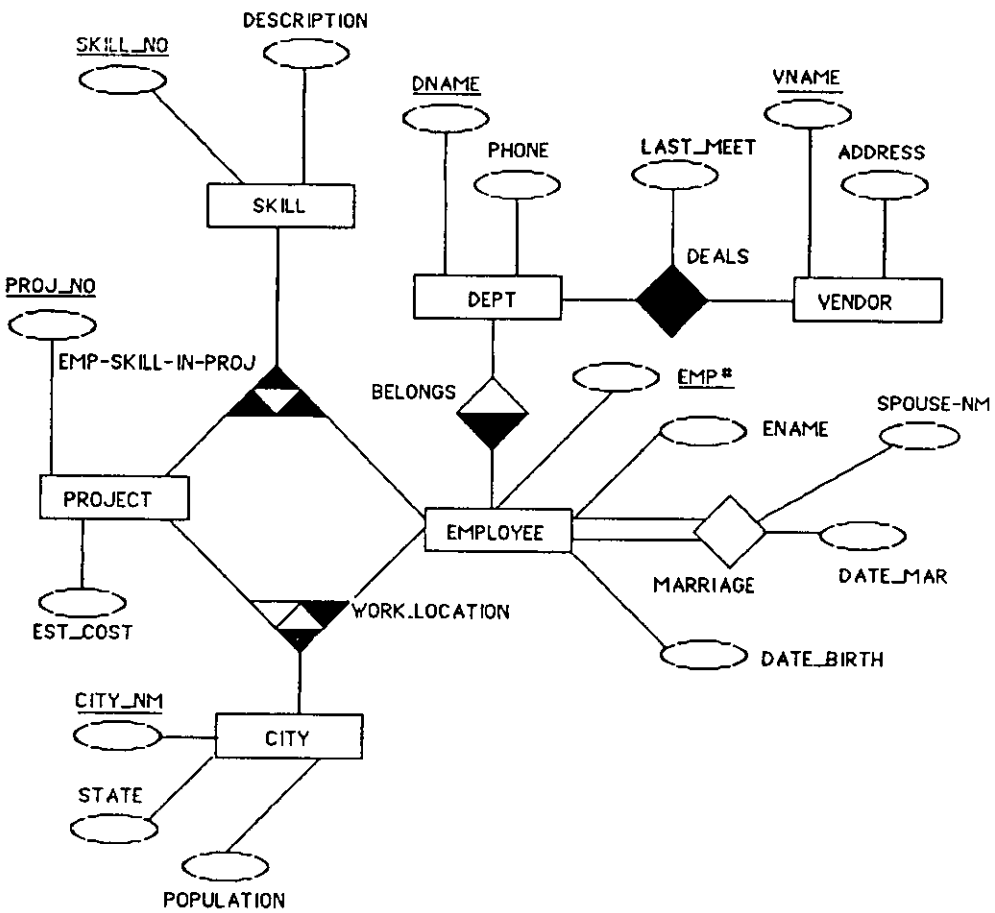
Wetherbe, J. C., and Leithheiser, R. L. "Information Centers: A Survey of Services," *Journal of Information Systems Management*, Volume 2, Number 3, Summer 1985, pp. 3-10.

Yao, S. *Principles of Database Design: Volume 1, Logical Organizations*. Englewood Cliffs, New Jersey: Prentice-Hall, 1985.

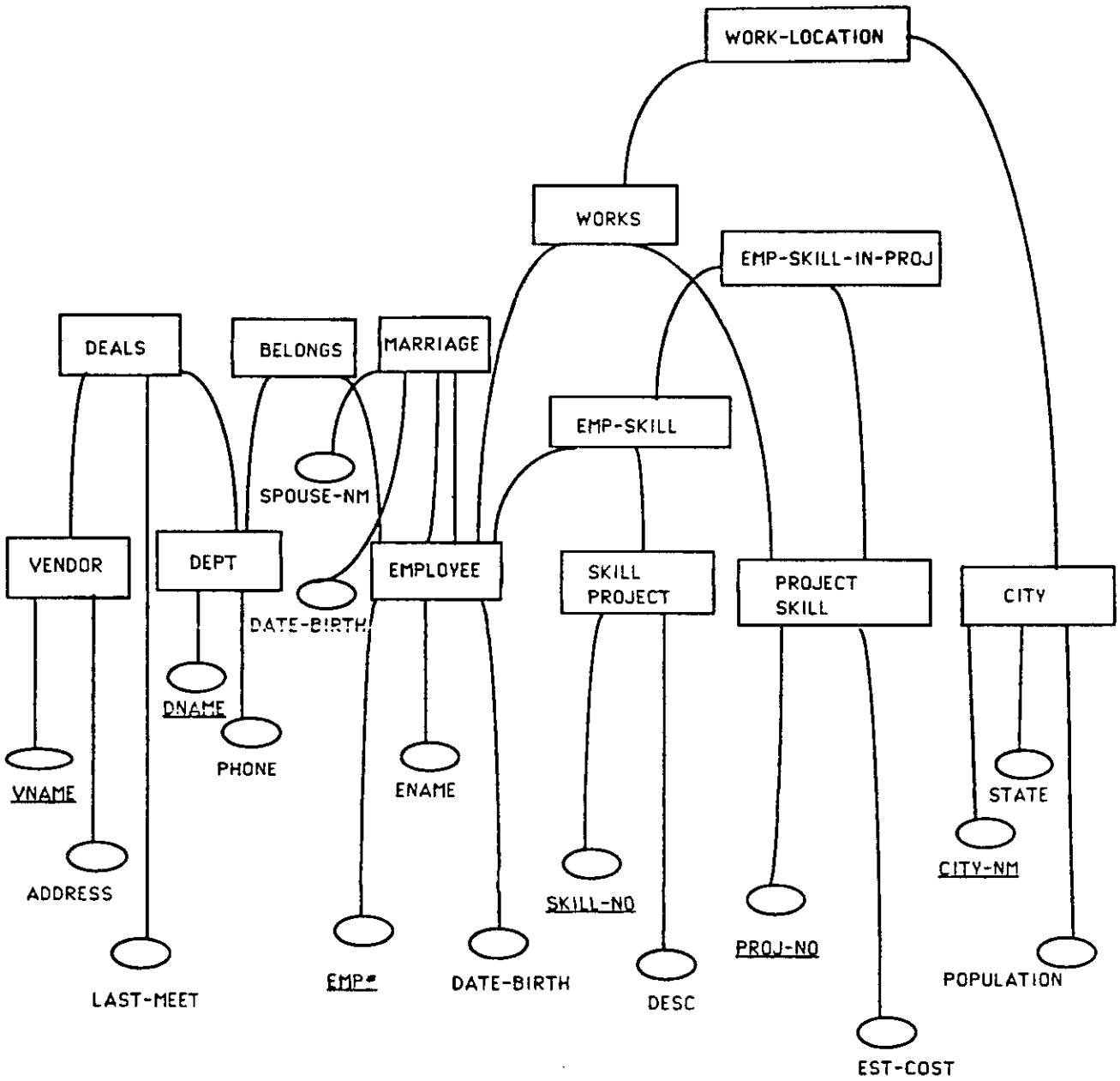
Appendix A. Experimental Task

The ABC Company wants to develop a database. Each ABC Company Employee has a unique ID Number assigned by the company. Other data which must be stored includes the employee"s name and date of birth. If an employee is married to another employee of the ABC Company, the name of the individual they are married to and the date of the marriage is to be stored. No record of marriage needs to be maintained if an employee is married to a non-employee. An employee cannot be married to more than one person. Each employee belongs to only one department. Each department is identified by name and has a different telephone number. Each department deals with many vendors when procuring equipment. A vendor typically deals with many departments. Data about which departments deal with which vendors is to be stored. Relevant vendor data includes vendor name and address. Many employees can work on many projects, but cannot work on more than one project in any given city. The employee can, however, work on the same project in many different cities. For example, Vicky can work on the SUPERCHEM Project in New York and the MAXIOIL Project in Minneapolis. She cannot work on the SUPERCHEM Project in New York and the MAXIOIL Project in New York (two projects in the same city). She can, however, work on the MAXIOIL Project in New York and the MAXIOIL Project in Boston (same project in different cities). For each city, the name of the city, the state in which it is located, and the population of the city must be stored in the database. In this case, the name is adequate to serve as an identifier. A project is identified by project number. The estimated cost of each project must be stored. An employee can have many skills. The number of employee skills applied varies from project to project. For example, Vicky may prepare requisitions and check drawings for the SUPERCHEM Project in New York, conduct inspections for the MAXIOIL project in Minneapolis, and prepare drawings and conduct inspections for the MAXIOIL Project in Boston (that is, an employee may use the same skills in different projects or different skills may be applied in each project). Although it is possible that all of an employee"s skills could be employed in the projects with which the employee is involved, it is more likely that not all of the skills will be used in any one project. Each skill has a code associated with it, which is to be stored along with a brief description of the skill.

Appendix B. ER Solution



Appendix C. DA Solution



Appendix D. Relational Representation Solution
(Adapted from Teorey, Yang and Fry 1986)

EMPLOYEE (EMP#, ENAME, DATE-BIRTH, SPOUSE#, SPOUSE-NM, DATE-MAR, DNAME)
DEPT (DNAME,PHONE)
VENDOR (VNAME, ADDRESS)
DEALS (DNAME,VNAME, LAST-MEET)
SKILL (SKILL#, DESCRIPTION)
PROJECT(PROJ#, EST-COST)
EMP-SKILL-IN-PROJ (EMP#,SKILL#,PROJ#)
CITY (CITY-NM, STATE, POP)
WORK-LOCATION (EMP#,CITY-NM, PROJ#)

Appendix E. Grading Guidelines

The grading was done using a scheme similar to the one used by Batra, Hoffer and Bostrom (1990). Errors were classified as *incorrect*, *major*, *medium* and *minor*, and scores of 0, 0.25, 0.50 and 0.75 were awarded, respectively. A correct representation resulted in a score of 1. The following guidelines were used:

1. An error in the degree of a relationship was classified as incorrect. For example, if a ternary relationship was shown as two binary relationships, the error was treated as incorrect. Missing relationships were, obviously, treated as incorrect, too.

2. An error in the connectivity of the relationship where the degree had been modeled correctly was classified as medium error. For example, if a one-many relationship was shown as a many-many relationship, the error was treated as medium.

3. If the relation for a relationship was not integrated with another relation when the two relations had common identifiers, the error was treated as minor. For example, the relation for a one-many relationship should be integrated with the relation for the entity on the "many" side of the relationship.

4. If the degree of a relationship was correct and the connectivity was incorrect, the error was classified as major if the identifiers of the entities involved in the relationship were not named correctly.