Comparing relational database designing approaches: some managerial implications for database training

Heng-Li Yang
Department of Management Information Systems, National Cheng-Chi University, Taipei, Taiwan

Keywords
Relational databases, Database packages, Databases, Decomposition method, User studies

Abstract
This research conducted two experiments to understand the performance (correctness and efficiency) of novice database designers, and perceptions of ease of use and preferences of two approaches for modeling relational databases: the semantic-oriented approach (top-down, e.g. using the entity-relationship model) and the logical-oriented approach (bottom-up, view decomposition, focusing only on the logical model). The findings indicated that in experiment 1, semantic-oriented treatments performed better in a complex, written-text case; logical-oriented treatments were better in a simple, tabular-form case. The same situation happened in experiment 2 though the differences were not statistically significant.

1. Introduction

Databases have become popular for decades of years (Anjard, 1994; Bracewell, 1988). In most applications, the relational data model (Childs, 1968; Codd, 1970) has also replaced earlier hierarchical and network data models. Nowadays, instructors at schools seldom teach those earlier data models. Though the object-oriented databases have been introduced for several years (Waltz et al., 1995; Zhang, 2001), the relational model is still prevailing in business.

Basically, there are two popular approaches for relational database design. The first approach heavily relies on relational database theories, especially normalization theory. The designer might collect the information requirements for databases into a large relational table, i.e. universal relation, and then decompose it into several smaller tables step-by-step to conform to first, second, third (or even higher) normal forms. Beginning with only one big universal relation to design a real database is not valid in practice[1]. But in the software industry, there is a popular and similar way to design databases - intuitively collecting information requirements into several relational tables and then decomposing them (Stamper and Price, 1990). This process is called view decomposition (Batra, 1997). Alternatively, this approach is called a bottom-up design that views relational database schema design strictly in terms of functional and other types of dependencies specified on the database attributes (Elmasri and Navathe, 2000). This approach only pays attention to the logical data model - the relational model. In this paper, this approach is abbreviated to "logical-oriented".

The second approach (hereafter abbreviated as "semantic-oriented") applies some semantic data model as an intermediate tool, begins with conceptual database design, and then transforms into the logical database schema (i.e. relational data model). The most popular semantic data model may be entity-relationship (ER) model, first proposed by Chen (1976). A variety of extended (or expanded) ER (EER) models have also existed (e.g. Elmasri et al., 1985). This approach is also named as a top-down design (Elmasri and Navathe, 2000). In this approach, designers may still apply the normalization theory to check the transformed relations. However, if the designer captures the concepts of entities and relationships adequately and transforms them to relations correctly, it would be unlikely that a violation of 4NF would occur in the resulting relations (Storey, 1988).

In the author's personal opinion, instructors should teach both approaches. However, some textbooks do not give adequate emphasis on semantic data models. For example, the famous Date's book has only less than seven pages of introduction to the ER model even in his fifth edition of his book (1990). Until 1995, Date first gave it one whole chapter space, but commented on it as "not so rigorous", "not objective". Similarly, in the famous Ullman's books (1982; 1989), only a few pages or, at most, one section, is used to introduce the ER model. Some practical database books (e.g. Stamper and Price, 1990) also claimed that the logical-oriented approach is more appropriate for collecting information requirements from business forms (e.g. product order form). The aim of this research discussed here was to compare these two approaches empirically. We conducted two experiments to understand the performance (correctness and efficiency) of novice database designers, and the
perceptions of ease-of-use and preferences to these two approaches. We also try to investigate the possibility of the fit between approaches and task formats.

2. Literature review

In literature, there are a number of empirical studies on database modeling. We can classify them as follows.

2.1 Category I. Improving modeling skills

Some have focused on:

- the improvement of conceptual database design through feedback (Batra and Davis, 1992);
- the use of heuristics in the modeling process (Batra and Antony, 1984; Srinivasan and Téeni, 1995); and
- the possibilities of improving through cooperative learning or self-efficacy (Ryan et al., 2000).

Others have explored similarities and differences between experts and novices in database design. For example, Prietula and March (1991) reported an empirical study of physical database design to compare the form and substance of reasoning used by the more experienced and inexperienced subjects. Batra and Davis (1992) reported the findings about experts and novices engaged in a conceptual data modeling task. In both studies, data were gathered in the form of think-aloud protocols.

2.2 Category II. Comparing models

Researchers have compared different models:

- comparison of traditional logical data models (e.g. hierarchical, network, or relational models);
- comparison of semantic data models (e.g. entity-relationship model, semantic object model, binary model) with logical data models; or
- comparison of semantic data models.

Based on the dependent variables, we can classify them into sub-categories here.

Sub-category II.1. Comparing effects of different data models on user comprehension

For example, early Brosey and Schneiderman (1978) presented two experimental studies of relational and hierarchical models to compare comprehension, the ability to use the database information, and schema memorization of undergraduate students. Jarvenpaa and Machesky (1989) conducted three experiments to compare logical data structure (LDS) (Carlisle, 1989) and relational data model. One of their compared dependent variables is subject's understanding of the notation. Palvia et al. (1992) examined three models:

1. data structure diagram (Bachman, 1969);
2. ER model; and
3. object-oriented model (Kroenke, 1995);

based on end-users' (business college students) comprehension (also considering comprehension time). They gave subjects three versions of a database (with actual and identical data in each) corresponding to those three models. They also solicited perceptions for each model from MBA students in their second study. Hardgrave and Dalal (1995) compared extended ER model (McFadden and Hoffer, 1994) and object modeling technique (OMT) (Rumbaugh et al., 1991) using three criteria:

1. model understanding;
2. time to understand; and
3. perceived ease-of-use.

The first experiment, conducted by Kim and March (1995), also compared Teorey's extended ER model (Teorey et al., 1986) and Nijssen information analysis methodology (NIAM, a binary model) (Verheijen and Van Bekkum, 1982) in terms of model comprehension (measuring syntactic competence) and discrepancy-checking (measuring semantic competence) of graduate business students.

Sub-category II.2. Comparing the different final representations of different data models built by data designers

For example, Batra et al. (1990) compared modeling correctness of relational and Elmasri's extended ER models (Elmasri et al., 1985) developed by graduate students. Modeling correctness was measured in terms of six facets (unary, binary one-many, binary many-many, ternary one-many, ternary many-many-many relationships, and identifier). Designer's perceived ease-of-use perception was also collected. In the experiments of Jarvenpaa and Machesky (1989) (as mentioned in the above sub-category), accuracy of data models and completion time were measured. Bock and Ryan (1993) compared modeling correctness of senior/graduate students in developing conceptual data models for Elmasri's extended ER model (Elmasri et al., 1985) and Kroenke's object-oriented model. In the second experiment of Kim and March's research (1995) (as mentioned in the above sub-category), they compared modeling performance (semantic and syntactic) and perceived usefulness of extended ER model and NIAM. Lee and Choi (1999) compared the...
correctness of EER, Kroenke’s SOM, ORM (Halpin, 1985) and OMT (Blaha et al., 1988; Rumbaugh et al., 1991).

Sub-category II.3. Comparing the same final representations of different data models built by data designers

The designers in this sub-category were asked to develop the same final representations. For example, Shoval and Even-Chaime (1987) compared normalization and NIAM for designing the same final output – normalized relational schema. The result of the experiment revealed that the quality of the database designed using normalization was better than that designed using NIAM. It also concluded that normalization required less time to perform, and the analysts preferred normalization. However, there were only 26 graduate student subjects, and the quality was just measured in an overall score. Sarwar and Marshall (1993) compared object-oriented approach (Shaler and Mellor, 1988) to data-element approach (similar to our logical-oriented approach) for deriving the same normalized relations. They had 63 senior students as subjects and found that the method chosen did not affect task execution performance, but might be more influential in determining the designer’s perceived intrinsic value of the database design (the object-oriented approach was higher). Marshall and Gibson (1996) compared Kroneke’s semantic object approach to a technology-based approach (Sierra’s Analyst software). A total of 71 subjects could choose either approach or use a combined approach. They found that task performance was, in general, higher for the Kroenke group, but only significant (at the probability of 0.061) at key specification. In addition, the performance and perceptions with respect to type of support were not entirely consistent. For example, while the Kroenke’s method appeared to provide higher performance support, its perceived usability (i.e., simplicity in concept and ease-of-use) might be suspect.

2.3 Category III. Comparing modeling tasks

In the above studies, comparison focused on the data models but not the task. In the research of Lee and Choi (1998), they also tried to compare two tasks. They found that the better subjects in the natural language cases may be also better subjects in form cases in the entity/object/class attribute/entity (value) type, binary 1:1, modeling time and perceived ease-of-use. However, those two tasks were not only in different formats (natural language vs enterprise form), but also included different contents (hospital vs industry). Batra (1997) suggested that two characteristics of user views (which is either text description or reports) significantly affect designer performance in normalizing the views.

Observation

From the above literature review, we can find two weaknesses:

1. in Category II, no research directly compared top-down (ER) approach with bottom-up (logical-oriented) approach for designing relational databases; and

2. in Category III, the research only compared tasks, not the fit between models and task formats.

This motivates our research. Actually, our research belongs to the intersection between Categories II and III. We tried to compare the same final representations (relational schema) of different data modeling approaches, and also investigate whether any approach fit better with which format.

3. Research design

We adopted the laboratory experiment strategy with the questionnaire data collection method. Because of some interesting findings in the first experiment, we held the second experiment in the second year.

3.1 Subjects

Subjects of two experiments were sophomores in the MIS Department at National Cheng-Chi University, Taiwan. There were 101 students in the first experiment, 98 in the second. All(2) of them had taken requisite courses – basic computing course (six credits), introduction to MIS (three credits), data structure (three credits) in the past, and had been taking a database design course from the author, and system analysis and design course from another instructor for more than three months[3]. The database design score of the experiments would be counted as a material part of the semester credit. They could be surrogates of novice database designers.

3.2 Experiment procedures

Each experiment had three phases. Two weeks prior to each experiment, their background information was collected. Then, in each experiment, subjects were required to model database cases. The time limits of Experiments I and II were 180, and 90 minutes, respectively. Finally, they should fill in a post-experiment questionnaire about method ease-of-use and preference.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
3.3 Questionnaires

The collected background information included: sex, age, past grades of related computer courses, and their cognitive styles. The questionnaire of cognitive styles was taken from Slocum and Hellriegel (1983), translated into Chinese, and pre-tested to assure consistency of Chinese vs English version. Slocum and Hellriegel’s questionnaire was similar to the MBTI instrument (Myers and McCaulley, 1985), but was abridged. There were 16 questions in the questionnaire. The subjects could be classified into “sensing” and “intuition” styles according to information gathering factor; and “feeling” and “thinking” styles according to information evaluation factor. Because the information was already given in the experiments, we were more interested in information evaluation factor. According to Slocum and Hellriegel (1983), characteristics of “thinking” style would be:

- logic;
- organization;
- analysis; and
- rational problem solving, etc.,
and characteristics of “feeling” style would be:

- human interaction;
- feelings; and
- emotions.

Thus, we might expect that modeling performances of “thinking”-style persons would be better.

The post-experiment questionnaire included:

- five questions taken from Batra et al. (1990) to measure perceived ease of use (which were essentially adapted from the instrument developed by Davis (1985));
- two questions to compare perceived ease-of-use of two methods in each case(4);
- a closed-ended question on a seven-point Likert scale to measure method preference in general; and
- an open-ended question to ask reasons of preference.

Readers can see these questions in a later “Findings” section. The internal reliability, Cronbach alpha, of the ease-of-use questionnaire, were 0.91 in the first experiment, and 0.88 in the second experiment.

3.4 Experiment design

In both experiments, after randomly assigning students into treatments, we examined backgrounds among treatments and could not find significant differences of their past related course grades, sex, or ages (see Table I for their background information).

Both experiments were used to test two methods:

1. **Method 1** – the semantic-oriented approach; and
2. **Method 2** – the logical-oriented approach.

Actually, Method 1 in Case A of Experiment I would need extended ER (EER), since subjects should apply a generalization/specialization concept. There were two test cases in the first experiment: Case A was more complex, described in words and adapted from Batra et al. (1990); Case B was simple, presented in a form (as Figure 1, adapted from Stamper and Price (1990)(5)]. Each subject was required to model both cases, but use different methods. If he/she was first required to apply Method 1 to Case A, then he/she must apply Method 2 to Case B, and vice versa. When applying Method 1, subjects should first draw an ER diagram (see Figures 2 and 3), and then convert it to a normalized (3NF) relational schema. When applying Method 2, subjects should first write down one or several initial relations, then step-by-step decompose it into 1NF, 2NF and, finally, 3NF relations. Subjects must also record their beginning and end time points of modeling each case. Only when they turned in their answering sheets of the first case, would they receive answering sheets of the second case. Thus, they were not allowed to change their first part answers. To avoid ordering bias, we also considered different sequences of modeling cases. Thus, we had the Experiment I design as Table II. Students were randomly assigned into each of four treatments in a principle of keeping the respective numbers of “feeling”, “thinking”, “sensing”, “intuition” styles in each treatment almost equal.

There was only one case in Experiment II, but in two formats: Format A in text descriptions, Format B in two tabular forms (only format B is given here as Figures 4 and 5, its ER solution is shown as Figure 6). Subjects were required to use only one method. Thus, we had the Experiment II design as Table III. Similarly, we randomly assigned students into each of four treatments in the same principle of keeping their cognitive style distribution almost equal.

3.5 Dependent variables

The dependent variables in both experiments were modeling correctness, time, perceived ease of use, and preference. Modeling correctness had several facets. In the complex Case A of Experiment I, it included:

- unary, binary 1-1;
- binary M-N;
- ternary 1-M-N;

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Table I

Students’ profile in two experiments

<table>
<thead>
<tr>
<th>Background information</th>
<th>Experiment I</th>
<th>Experiment II</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex category</td>
<td>Male: 41; Female: 60</td>
<td>Male: 57; Female: 41</td>
</tr>
<tr>
<td>Age</td>
<td>Max: 26; Min: 19; Middle: 20.0; Average: 20.4 (n=101); Standard deviation: 1.58</td>
<td>Max: 29; Min: 19; Middle: 20.0; Average: 20.5 (n=98); Standard deviation: 1.78</td>
</tr>
<tr>
<td>Grade of basic computing course I</td>
<td>Max: 91; Min: 45; Middle: 71.0; Average: 71.0 (n=95); Standard deviation: 8.73</td>
<td>Max: 98; Min: 62; Middle: 80.0; Average: 79.1 (n=98); Standard deviation: 7.15</td>
</tr>
<tr>
<td>Grade of basic computing course II</td>
<td>Max: 95; Min: 60; Middle: 67.0; Average: 68.5 (n=91); Standard deviation: 8.24</td>
<td>Max: 97; Min: 60; Middle: 71.5; Average: 72.4 (n=96); Standard deviation: 9.34</td>
</tr>
<tr>
<td>Grade of introduction to MIS</td>
<td>Max: 88; Min: 60; Middle: 75.0; Average: 75.2 (n=97); Standard deviation: 5.89</td>
<td>Max: 93; Min: 60; Middle: 75.0; Average: 74.7 (n=97); Standard deviation: 7.29</td>
</tr>
<tr>
<td>Information evaluation style</td>
<td>All students Feeling: 89; Thinking: 12</td>
<td>All students Feeling: 76; Thinking: 22</td>
</tr>
<tr>
<td>distribution</td>
<td>Treatment I Feeling: 22; Thinking: 3</td>
<td>Treatment I Feeling: 19; Thinking: 5</td>
</tr>
<tr>
<td>Treatment II Feeling: 22; Thinking: 3</td>
<td>Treatment II Feeling: 19; Thinking: 5</td>
<td></td>
</tr>
<tr>
<td>Treatment III Feeling: 22; Thinking: 3</td>
<td>Treatment III Feeling: 19; Thinking: 6</td>
<td></td>
</tr>
<tr>
<td>Treatment IV Feeling: 23; Thinking: 3</td>
<td>Treatment IV Feeling: 20; Thinking: 5</td>
<td></td>
</tr>
</tbody>
</table>

Note: Because some students (e.g., transferring students) were exempted from the above basic courses, the numbers of students of these courses were not same.

Figure 1

The Case B in Experiment I

---

- ternary M-N-L relationships;
- entity;
- candidate key;
- attribute;
- normalization;
- generalization and specialization; and
- redundancy.

The Case B of Experiment I and both formats of Experiment II were simpler, so the facets of their modeling correctness did not include unary, ternary relationships and generalization and specialization. Besides, though Batra et al. (1990) did not suggest to add up the scores of facets, we still had four...
"overall indicators" in each case just for references. These were:

- total score (summing up all of the above facet scores);
- all relationships (summing up all of the above relationship facet scores);
- overall correctness; and
- overall correctness and process.

Overall correctness was a general evaluation on the final relations. We did not give equal weights to each facet, but give more emphasis on entity, relationship, and normalization. The "overall correctness and process" was a further aggregate score, which not only included the variable "overall correctness" (i.e. the final relations satisfying 3NF), but also considered correctness during the modeling process – correctness of ER diagram of the Step 1 in method 1 or the 1NF, 2NF of the Steps 1-2 in method 2. It was the score given to the student as a part of his/her course credit.

3.6 Grading

The answering sheets were graded for correctness comparing with correct solutions. Each facet was graded separately. Errors were classified as incorrect, major, medium, and minor, and 100 per cent, 75 per cent, 50 per cent, and 25 per cent were deducted, respectively[6]. We considered a situation if a subject got more relations, the possibilities of making mistakes in finding candidate keys and assuring normalization were greater. To avoid this kind of punishment, the facets of "candidate key" and "normalization" were graded on a relative base. For example, if a subject got 12 or 16 relations, the full scores of his two facets were 120 or 160, respectively. Finally, we normalized the score of each facet to the base 100.

3.7 Hypotheses

In Experiment I, we tried to test the following null hypotheses:

H1. Modeling methods had no significant effect on modeling correctness, time, perceived ease-of-use, or preference.

H2. Information evaluation style had no significant effect on modeling correctness, time, perceived ease-of-use, and preference.

Because of the findings in Experiment I, we are more interested in the case formats. In
Experiment II, we tried to test the following hypotheses:

H1. The significant effects of modeling methods on modeling correctness, time, perceived ease-of-use, and preference would depend on case formats. The semantic-oriented approach would be superior on the format of text description; the logical-oriented would be superior on the format of tabular form.

H2. Information evaluation style and case format had an interaction effect.

4 Findings

4.1 Experiment I

From Tables IV and V, we can find the following:

- Information evaluation style had no effect in Case B, but had significant effects on three variables of Case A (t-tests): binary M-N relationship, candidate key (significant at less than 0.05), and ternary M-N-L relationship (significant at less than 0.1). “Thinking” style subjects had higher scores. However, if we considered the method effect at the same time (ANOVA tests of Table IV), these three variables were significant only at level of 0.1.

- Modeling method had significant effects on seven variables of Case A (t-tests): binary M-N relationship, entity, candidate key, generalization/specialization, overall correctness and process (these five were significant at less than 0.05); ternary 1-M-N relationship, and total score (these two were significant at less than 0.1). Their significant probabilities did not have any change even if we considered the information evaluation style effect (ANOVA tests of Table IV). Treatments applying Method 1 (i.e. the semantic-oriented) scored higher in all of these seven variables except the ternary 1-M-N relationship. In fact, the semantic-oriented treatments also scored lower in the ternary M-N-L relationship.

![Image of a database diagram](image-url)

**Figure 3**

The corresponding ER diagram of Case B in Experiment I

**Table II**

<table>
<thead>
<tr>
<th>Experiment I design</th>
<th>First do case A</th>
<th>First do case B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apply Method 1 to text descriptions</td>
<td>Treatment I: (A1, B2)</td>
<td>Treatment II: (B2, A1)</td>
</tr>
<tr>
<td>Apply Method 2 to table descriptions</td>
<td>Treatment III: (A2, B1)</td>
<td>Treatment IV: (B1, A2)</td>
</tr>
</tbody>
</table>

**Note:** A1 denotes “applying Method 1 (semantic-oriented) to Case A”; B2 denotes “applying Method 2 (logical-oriented) to Case B”; others are similar.
though not significantly. But, the semantic-oriented treatments still took somewhat less time to model this case (80.22 minutes vs 85.58).

- Modeling method had also significant effects on seven variables of Case B (t-tests): binary M-N relationship, attribute, normalization, total score, overall correctness, all relationships, overall correctness and process (all of them were significant at less than 0.05). Neither would their significant probabilities change if we considered the information evaluation style effect (ANOVA tests of Table V). Treatments applying Method 2 (i.e. the logical-oriented) scored higher in all of these variables. The logical-oriented treatment took insignificantly longer time to complete (59.94 minutes vs 54.45).

The wording and scores of questions (4) and (5) in Table VI were reversed from the original questionnaire for giving the same direction as other (1) to (3) questions. From Table VI, we can find the following:

- Generally speaking, all subjects perceived that the semantic-oriented approach might be not so difficult to use (in question (5) of Table VI, average was 3.79, a little toward disagreement to that question). They significantly disagreed that it would not be clear and understandable as the logical-oriented approach, and showed preference to the semantic-oriented in both cases (though only Case A was significant). However, they strongly felt that using the semantic-oriented approach required more mental effort.

- Treatments III and IV more disagreed that the semantic-oriented approach was cumbersome, frustrating, not clear or understandable, difficult, or required more mental effort in both cases. They also presented more preferences to the semantic-oriented. However, we have seen that the performance of treatments III and IV was inferior to the other two treatments. This finding is similar to Marshall and Gibson (1996) that performance and perceptions of support were not entirely consistent. Treatments III and IV were required to apply the logical-oriented approach to Case A, and the semantic-oriented approach to Case B. To them, the logical-oriented approach
Figure 5
The Format B in Experiment II (continued)

X Company Shipment

<table>
<thead>
<tr>
<th>Customer#</th>
<th>Customer name</th>
<th>Address</th>
<th>Telephone</th>
<th>Shipment#</th>
<th>Date</th>
<th>Order#</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Product#</th>
<th>Description</th>
<th>Order Quantity</th>
<th>Shipped Quantity</th>
<th>Undeliverable Quantity</th>
<th>Unit Price</th>
<th>Amount Owing to Shipment</th>
</tr>
</thead>
</table>

Total Amount: ________
Discount: ________
Payable Amount owing to this Shipment: ________

Notes: 1. Each shipment has its unique Shipment#: 2. Each customer order may be partitioned into several shipments; 3. In the case that an order is partitioned into several shipments, if the original order is offered 10% discount, each shipment will also have the same discount.

Figure 6
The corresponding ER diagram (of both formats) in Experiment II
might be difficult in the complex Case A; on the other hand, it might seem that the semantic-oriented was easier in the simpler Case B. However, treatments III and IV did not draw the ER diagrams correctly[7]. Probably because the experience of applying the semantic-oriented approach to the Case B, treatments III and IV had similar favorable attitude toward applying the semantic-oriented approach to the Case A. • In general, “thinking” style subjects perceived the semantic-oriented approach more easy-of-use and had more preference
than the "feeling" style, but the differences were not significant. Interestingly, the "thinking" style significantly perceived the ER more easy-of-use in Case A, but they had insignificantly negative attitude to the ER ease-of-use in Case B. We debriefed these students. Some stated that they were less confident of their ER solutions to the seemingly simple Case B.

In the open-ended question, subjects expressed their reasons of preference. Based on content analysis, they were summarized below:

- Out of 41 writing reasons of preference for the semantic-oriented approach, 20 explained that ER was easier to understand, the relationships among application objects were more obvious, its abstraction level of concepts was higher; 13 stated that it was easy to understand and use by drawing the ER diagram.
- Out of 30 writing reasons of preference for the logical-oriented approach, nine explained that it was simpler, easy to understand; eight claimed that it was procedure-driven, logical and orderly; four stated that it was hard to find relationships in the ER; another four said that the ER diagram was hard to draw.

Thus, we found that only in the complex and written-text Case A, information evaluation style had limited effects. Therefore, it seems that we can reject the H1, but reject H2 only partially. The sentence descriptions in Case A might give subjects more hints in correctly finding entity, candidate key, binary M:N relationship, and generalization/specialization while they drew the ER diagram. However, Cases A and B were different in terms of not only complexity but also format. We could not jump to conclusions that the semantic-oriented approach is suitable for text descriptions, the logical-oriented approach suitable for tabular form. This motivated our Experiment II in which we designed the tasks that had the same complexity, but different formats.

### 4.2 Experiment II

First, let us only consider the method variable (the left part of Table VII):

- With respect to Format A (text descriptions), the semantic-oriented treatment (treatment I) obtained higher scores of almost all correctness variables (except normalization variable) of Table VII. The semantic-oriented treatment also needs less modeling time (78.63 minutes vs 84.64). But, the

<table>
<thead>
<tr>
<th>Table VI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Post-experiment t-test in Experiment I</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ease-to-use and preference variables</th>
<th>Modeling method</th>
<th>Information evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>All subjects</strong></td>
<td>Treatments I, II scores</td>
<td>Treatments III, IV scores</td>
</tr>
<tr>
<td>1. I found the semantic-oriented approach more cumbersome to use than the logical-oriented approach</td>
<td>3.76</td>
<td>4.24</td>
</tr>
<tr>
<td>2. Using the semantic-oriented approach was more frustrating than the logical-oriented approach</td>
<td>3.74</td>
<td>4.20</td>
</tr>
<tr>
<td>3. Using the semantic-oriented approach required more mental effort than the logical-oriented approach</td>
<td>4.34**</td>
<td>4.72**</td>
</tr>
<tr>
<td>4. To me, the semantic-oriented approach is not as clear and understandable as the logical-oriented approach</td>
<td>3.63**</td>
<td>3.82</td>
</tr>
<tr>
<td>5. Overall, I found the semantic-oriented approach more difficult to use than the logical-oriented approach</td>
<td>3.79</td>
<td>4.14</td>
</tr>
<tr>
<td>6. I prefer to the semantic-oriented approach, not the logical-oriented approach</td>
<td>4.35**</td>
<td>4.18</td>
</tr>
<tr>
<td>7. To case A (Projects Inc. personnel database), I found the semantic-oriented approach easier to use</td>
<td>4.33*</td>
<td>4.16</td>
</tr>
<tr>
<td>8. To case B (Granger Ranch invoice database), I found the semantic-oriented approach easier to use</td>
<td>4.23</td>
<td>4.02</td>
</tr>
<tr>
<td>9. Averaging (1) to (5), the semantic-oriented approach was more difficult to use than the logical-oriented approach</td>
<td>3.8688</td>
<td>4.2245</td>
</tr>
</tbody>
</table>

**Notes:** 1. Please reference Table II for each treatment’s work requirement; 2. The interval scale of each item was from one (strongly agree) to seven (strongly agree); middle was four; 3. The * marks are significant at 0.10; ** marks are significant at 0.05. Those marks in the second, third, fourth, sixth and seventh columns indicate the significant difference from the middle rating four.
differences were only significant in attribute and total score.

- With regard to Format B (tabular forms), the logical-oriented treatment (treatment IV) scored higher in seven correctness variables of Table VII, lower in candidate key, normalization, redundancy and overall correctness and process. But, only the difference of redundancy was significant. In contrast to Experiment I, the logical-oriented treatment had insignificantly lower score in the normalization facet. It might be mainly because they did not detect the redundant information while constructing the initial relation(s). The logical-oriented treatment also need significant less modeling time (51 minutes vs 73.64).

- If not making a distinction of formats, the semantic-oriented treatments (treatment I and II) performed better in all correctness variables except entity and took longer time in modeling. But only redundancy, overall correctness and process, and time had significant differences.

Then, consider the variables of method, format, and information evaluation together (the right part of Table VII).

- Information evaluation style had no individually significant effect on any modeling correctness variable or time.
- The only variable on which method had individually significant effect was overall correctness and process.
- Subjects modeling Format B performed significantly better than Format A with respect to redundancy and spent significantly less time.
- Method and format had an interaction effect on attribute and time. The semantic-oriented treatment scored significantly higher in attribute and took (insignificantly) less time while modeling Format A; the logical-oriented treatment scored higher (but insignificantly) in attribute and spent significantly less time while modeling Format B.
- Information evaluation and format had an interaction effect on binary M-N relationship, candidate key, total score, and total relationships. Considering about different formats, “thinking” style subjects performed (insignificantly) better in all these variables while modeling Format B; “feeling” style significantly

| Table VII |
| Tests on modeling correctness and time in experiment II |

<table>
<thead>
<tr>
<th>Dependent variables</th>
<th>Treatment I vs Treatment III</th>
<th>Treatment II vs Treatment IV</th>
<th>Treatments I, II vs Treatments III, IV</th>
<th>Method</th>
<th>Format</th>
<th>Information evaluation</th>
<th>Interaction of method and format</th>
<th>Interaction of ANOVA</th>
<th>Interaction of format and Information evaluation</th>
<th>Interaction of all three</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Binary 1-1 relationship</td>
<td>0.109</td>
<td>0.292</td>
<td>0.722</td>
<td>0.775</td>
<td>0.163</td>
<td>0.977</td>
<td>0.106</td>
<td>0.924</td>
<td>0.123</td>
<td>0.930</td>
</tr>
<tr>
<td>2 Binary M-N relationship</td>
<td>0.288</td>
<td>0.629</td>
<td>0.641</td>
<td>0.950</td>
<td>0.596</td>
<td>0.850</td>
<td>0.292</td>
<td>0.659</td>
<td>0.032**</td>
<td>0.869</td>
</tr>
<tr>
<td>3 Entity</td>
<td>0.202</td>
<td>0.141</td>
<td>0.728</td>
<td>0.909</td>
<td>0.231</td>
<td>0.138</td>
<td>0.271</td>
<td>0.491</td>
<td>0.720</td>
<td>0.398</td>
</tr>
<tr>
<td>4 Candidate key</td>
<td>0.193</td>
<td>0.947</td>
<td>0.358</td>
<td>0.722</td>
<td>0.314</td>
<td>0.511</td>
<td>0.259</td>
<td>0.528</td>
<td>0.039**</td>
<td>0.498</td>
</tr>
<tr>
<td>5 Attribute</td>
<td>0.075*</td>
<td>0.365</td>
<td>0.454</td>
<td>0.275</td>
<td>0.359</td>
<td>0.721</td>
<td>0.051*</td>
<td>0.348</td>
<td>0.076*</td>
<td>0.551</td>
</tr>
<tr>
<td>6 Normalization</td>
<td>0.618</td>
<td>0.337</td>
<td>0.777</td>
<td>0.691</td>
<td>0.745</td>
<td>0.838</td>
<td>0.682</td>
<td>0.725</td>
<td>0.445</td>
<td>0.467</td>
</tr>
<tr>
<td>7 Redundancy</td>
<td>0.327</td>
<td>0.049**</td>
<td>0.052*</td>
<td>0.154</td>
<td>0.022**</td>
<td>0.165</td>
<td>0.345</td>
<td>0.540</td>
<td>0.196</td>
<td>0.365</td>
</tr>
<tr>
<td>8 Overall correctness</td>
<td>0.638</td>
<td>0.773</td>
<td>0.886</td>
<td>0.999</td>
<td>0.732</td>
<td>0.964</td>
<td>0.396</td>
<td>0.856</td>
<td>0.693</td>
<td>0.461</td>
</tr>
<tr>
<td>9 Total score &lt; summing-up (1 to (7))</td>
<td>0.095*</td>
<td>0.666</td>
<td>0.342</td>
<td>0.489</td>
<td>0.981</td>
<td>0.927</td>
<td>0.174</td>
<td>0.959</td>
<td>0.073*</td>
<td>0.901</td>
</tr>
<tr>
<td>10 All relationships &lt; (1) + (2)</td>
<td>0.121</td>
<td>0.334</td>
<td>0.609</td>
<td>0.836</td>
<td>0.641</td>
<td>0.888</td>
<td>0.103</td>
<td>0.814</td>
<td>0.022**</td>
<td>0.873</td>
</tr>
<tr>
<td>11 Overall correctness and process</td>
<td>0.114</td>
<td>0.434</td>
<td>0.090*</td>
<td>0.095*</td>
<td>0.676</td>
<td>0.818</td>
<td>0.420</td>
<td>0.599</td>
<td>0.872</td>
<td>0.568</td>
</tr>
<tr>
<td>12 Modeling time</td>
<td>0.351</td>
<td>&lt;0.001**</td>
<td>0.089*</td>
<td>0.366</td>
<td>&lt;0.001**</td>
<td>0.811</td>
<td>0.005**</td>
<td>0.196</td>
<td>0.990</td>
<td>0.939</td>
</tr>
</tbody>
</table>

Notes: 1. The * marks are significant at 0.10; ** marks are significant at 0.05; 2. Treatment I applied method 1 to format A, Treatment II applied method 1 to format B; Treatment III applied method 2 to format A, Treatment IV applied method 2 to format B

[161] Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
performed better in all these variables while modeling Format A.

Similarly to Experiment I, from Table VIII, we can find the following:
- Generally speaking, all subjects significantly perceived the semantic-oriented approach easy to use, clear and understandable as the logical-oriented approach, but requiring more mental effort. They also more significantly preferred it to the semantic-oriented approach.
- "Thinking" style subjects significantly more disagreed that the semantic-oriented approach was cumbersome, not clear and understandable, or required more mental effort. They also more preferred the semantic-oriented approach than the "feeling" style though the difference was not significant.
- Treatment I applied the semantic-oriented approach to Format A and Treatment IV applied the logical-oriented approach to Format B. If that implied a good fit, the subjects might be more comfortable. Table VIII combined the perceptions of Treatments I and IV to compare with the perceptions of Treatments II and III. There was no significant difference between these two combined groups. However, these four treatments did have some differences of perceptions. If we run ANOVA to consider the variables of method and information evaluation style together, method would have significant effects on the (4) and (5) questions. Looking at the Table VIII, Treatment II significantly disagreed that the semantic-oriented approach was not clear or understandable, more difficult to use. Treatment II applied the semantic-oriented approach to Format B. Checking their scores, we found that they got significantly lower score in finding the correct binary N-M relationship. Again, similar to Experiment I, we found that performance and perceptions of support were not consistent.

Subjects also expressed similar reasons of preference in the open-ended question.

Thus, since only few tests of variables were significant, we got partial supports of H1 and H2 in Experiment II.

5 Conclusions and suggestions

5.1 Contributions

This research conducted two experiments to compare the semantic-oriented approach with the logical-oriented approach. The number of subjects in our experiment was larger than the previous research in literature. In addition, comparing to the previous research that had only a short period of training (e.g., eight hours in Bock and Ryan (1993)), our 42 hours training would be more appropriate and subjects were more homogeneous. We also tested possible effects of information evaluation styles and different task formats, which were not considered in the past.

5.2 Conclusions

From these two experiments, we make the following three conclusions:

1. Experiment I indicated that the semantic-oriented treatments performed better (especially in binary M-N relationship, entity, candidate key, and generalization/specialization) in a complex, written-text case; the logical-oriented treatments were better (especially in binary M-N relationship, attribute, and normalization) in a simple, tabular-form case. In Experiment II (keeping the same task complexity), though, roughly speaking, the semantic-oriented treatment was also superior in the written-text format and the logical-oriented treatment was superior in the tabular-form format, most of the statistically significant differences disappeared. (The semantic-oriented treatment only scored significantly higher in attribute; the logical-oriented treatment only significantly spent less time.)[8]

2. Experiment I indicated that the information evaluation style of subjects had limited effects on modeling correctness – no effect on a simple, tabular form case; had significant effects on three variables (binary 1-1 relationship, candidate key, and ternary M-N-L relationship) in a complex, written-text case ("thinking" style was better). In Experiment II, we found the interaction effect: considering about formats, with respect to binary M-N relationship and candidate key, "thinking" style subjects were (insignificantly) better while modeling the tabular form format; "feeling" style performed significantly better while modeling the written-text format.

3. Both experiments indicated that most subjects perceived the semantic-oriented approach easy to use, and more preferred to it. But they also complained that it required more mental efforts. Besides, we found that subject's performance and perception of support were not consistent.
<table>
<thead>
<tr>
<th>Ease to use and preference variables</th>
<th>Information evaluation</th>
<th>Modeling method</th>
<th>Significant level of difference</th>
<th>Treatment I scores</th>
<th>Treatment II scores</th>
<th>Treatment III scores</th>
<th>Treatment IV scores</th>
<th>Treatments I and IV scores</th>
<th>Treatments II and III scores</th>
<th>Significant level of difference</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. I found the semantic-oriented approach more cumbersome to use than the logical-oriented approach</td>
<td>3.88 4.03 3.36** 0.095*</td>
<td>3.75 4.0</td>
<td>3.76 4.0</td>
<td>3.88 3.88</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. Using the semantic-oriented approach was more frustrating than the logical-oriented approach</td>
<td>3.93 4.03 3.59 0.224</td>
<td>4.083 3.583</td>
<td>4.2 3.84</td>
<td>3.96 3.90</td>
<td>0.838</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. Using the semantic-oriented approach required more mental effort than the logical-oriented approach</td>
<td>4.50** 4.68** 3.86 0.026*</td>
<td>4.833** 4.625*</td>
<td>4.16 4.4</td>
<td>4.61* 4.39</td>
<td>0.472</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. To me, the semantic-oriented approach is not clear and understandable as the logical-oriented approach</td>
<td>3.63** 3.82 3** 0.030**</td>
<td>3.75 2.875**</td>
<td>4.2 3.68</td>
<td>3.71 3.55*</td>
<td>0.606</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. Overall, I found the semantic-oriented approach more difficult to use than the logical-oriented approach</td>
<td>3.69* 3.80 3.32** 0.208</td>
<td>4.25 3.125**</td>
<td>4.08 3.32**</td>
<td>3.78 3.61*</td>
<td>0.612</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. I prefer to the semantic-oriented approach, not the logical-oriented approach</td>
<td>4.28* 4.14 4.73** 0.123</td>
<td>3.958 4.625*</td>
<td>4.04 4.48*</td>
<td>4.22 4.33</td>
<td>0.748</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Averaging (1) to (5), the semantic-oriented approach was more difficult to use than the logical-oriented approach</td>
<td>4.06 4.12* 3.85 0.079*</td>
<td>4.12 4.005</td>
<td>4.105 4.015</td>
<td>4.07 4.06</td>
<td>0.937</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes: 1. The interval scale of each item was from one (strongly disagree) to seven (strongly agree); middle was four; 2. The * marks are significant at 0.10; ** marks are significant at 0.05; Those marks in the second to fourth, and sixth to eleventh columns indicate the significant difference from the middle rating four
5.3 Discussions and suggestions

Database design is part of system development. Strictly speaking, any system development should begin with system analysis. The results of system analysis are information requirements, which could be represented as texts, forms, diagrams, etc. In our two experiments, the text descriptions are closer to these information requirements. However, the tabular forms still remain the original enterprise forms. It could be conjectured that designers might get clues of entities and relationships from the written sentences of information requirements given by system analysts. However, the top-down approach implies semantic “data modeling”. As discussed by Srinivasan and Téeni (1995), data modeling includes cognitive activities, which need mental efforts. If the designer is a novice, the design can be error-prone (Storey et al., 1985). As argued by Whitley (1998) empirically, information system development methods are not being used in the way their developers intend them, but rather that pre-understanding is necessary for successful method use. On the contrary, the bottom-up approach has formal decomposition algorithms and can be easily learned by novices. A novice might have good performances in simple database design jobs, in which designers can directly get the attribute ideas from business tables or forms and then apply view decomposition procedures logically. This explains why ER modeling has not been prevailing in Taiwanese software houses, since their jobs are simple.

The fit phenomena disappeared in the second experiment. It was partly because first, in the second year, we had adjusted some database course lectures to teach students the often-made mistakes in both approaches and, second, the task is not complex in the second year.

If database design is a pure logical exercise, we might suggest that the “thinking” style persons are more suitable and they should prefer the logical-oriented approach. However, though in a complex, written-text case of Experiment I, “thinking” style did perform better than “feeling” style, there was no differences in Experiment II. It might be explained as the teaching effect of the second year. However, from the experience of Experiment II, if a “feeling” person wanted to model a database, it seems that he/she would perform better if system analysts would provide him/her with the written-text formats. The original tabular forms might be too complex for them.

From the above, we give managers some suggestions to database design training as follows:

- Database designers should cooperate with system analysts. The information requirements should either be written or drawn in some diagrams. Even if the enterprise forms are available, system analysts should provide text descriptions as complements. Database designers might easily check the original forms, derive modeling objects from those information requirements (and discussing with system analysts if needed). On the other hand, database concept is also the most important technical skill to system analysts (no matter whether experienced or not) as shown by Nord and Nord (1997).
- Both modeling methods are needed. From the author’s personal teaching experiences, novices might not adequately catch the ER constructs (especially ternary or higher order of relationships), though they look simple at first sight. In addition, errors in the ER model can result in non-normalized table structures (Bock, 1997). The focus of normalization theory could help them to check whether they had the correct ER diagrams (no partial, transitive dependencies, etc.). That is what we might call “normalized ER diagram”. On the other hand, the ER concept could help students to check whether they got some semantically reasonable relations while they applied the bottom-up procedure.
- There are also some implications for managers when recruiting new designers. Although logical persons are usually welcome in designing information systems, a “feeling” person could be trained to perform as well as a “thinking” person in this field. Besides, it seems that a database design job still contains part of arts rather than pure science, though some researchers (e.g. Storey, 1988) tried to improve this situation. It is possible that some characteristics of “feeling” style persons, such as human interactions, could help complement weaknesses of “thinking” persons. However, the cooperation with system analysts in defining information requirements could become more important if hiring “feeling” designers.
- In order to design a good ER model, we need to further study the modeling cognitive process and accumulate more heuristic rules as discussed in Srinivasan and Téeni (1995).
Notes
1 In fact, the universal relation concept has been just a historic note in the software development world.
2 Transfer students might be exempted.
3 Up to the experiments, the subjects had taken about 42 hours database training: basic concepts (three hours), ER model (nine hours), relational model and SQL (six hours), functional dependencies (five hours), normalization (five hours), database design (two hours), data dictionary and transaction (three hours), concurrency control and recovery (six hours), network and hierarchical model, and others (three hours).
4 Experiment II did not have these two questions because its subjects only apply one method.
5 Owing to the space limitation, the original Case A descriptions are not given here, interested readers might directly browse the paper in Batra et al. (1990). Instead, we give our ER modeling solutions (as shown in Figures 2 and 3) to Cases A and B.
6 “Overall correctness” and “overall correctness and process” had some detail grading criteria, but are not shown here owing to space limitation.
7 The most often made mistakes were that they did not recognize invoice as an entity to model its own attributes (such as selling terms) and keys (invoice number), but straightforwardly treat it as a simple relationship directly connecting the customer and product entities. So, the transformed relations were not correctly normalized.
8 Since both our ER approach and the object-oriented approach in Sankar and Marshall (1989) belong to the conceptual data model, and both our logical-oriented approach and their data-element approach focus on only the logical-data model, it might be interesting to compare two study findings. They found that the method did not affect task execution performance. Our Experiment I findings were different, but we had almost the same results in Experiment II.

References
Carls, J.V. (1985), Logical Data Structures, Computer Science Department, University of Minnesota, Minneapolis, MN, TR 85-23.
Date, C.J. (1986), An Introduction to Database Systems, 6th ed., Addison-Wesley, Reading, MA.
Davis, F.D. (1985), "A technology acceptance model for empirically testing new end-user information systems: theory and results", doctoral dissertation, Sloan School of Management, Massachusetts Institute of Technology, MA.
Elmasri, R., Weeldreyer, J. and Hevner, A. (1985), "The category concept: an extension to the


**Further reading**