1989

A STUDY OF CONCEPTUAL DATA MODELING IN DATABASE DESIGN: SIMILARITIES AND DIFFERENCES BETWEEN EXPERT AND NOVICE DESIGNERS

Dinesh Batra
Florida International University

Joseph G. Davis
Indiana University

A STUDY OF CONCEPTUAL DATA MODELING IN DATABASE DESIGN: SIMILARITIES AND DIFFERENCES BETWEEN EXPERT AND NOVICE DESIGNERS

Dinesh Batra
Department of Decision Sciences and Information Systems
Florida International University

Joseph G. Davis
Department of Operations and Systems Management
Indiana University

ABSTRACT

This paper explores the similarities and differences between experts and novices engaged in a conceptual data modeling task, a critical part of overall database design, using data gathered in the form of think-aloud protocols. It develops a three-level process model of the subjects' behavior and the differentiated application of this model by experts and novices. The study found that the experts focussed on generating a holistic understanding of the problem before developing the conceptual model. They were able to categorize problem descriptions into standard abstractions. The novices tended to have more errors in their solutions largely due to their inability to map parts of the problem description into appropriate knowledge structures. The study also found that the expert and novice behavior was similar in terms of modeling facets like entities, identifiers, descriptors, and binary and ternary relationships but was different in the modeling of unary relationships and categories. These findings are discussed in relation to the results of previous expert-novice studies in other domains.

1. INTRODUCTION

Developing the conceptual data model based on the detailed information requirements provided by users is a critical and demanding task in the overall database design. It is in this phase that the structure of the database to be implemented is captured along with the constraints. The conceptual model is usually easy to understand and can form the basis for communication with users. It does not include implementation details. This enables the users and designers to focus on specifying the properties of data without being concerned with file structures and storage details (CODASYL 1971; Elmasri and Navathe 1988). A good proportion of research in the area of conceptual modeling has been devoted to introducing additional formalisms for capturing greater meaning in the representations and the comparison of different data models for ease of representation from a human factors perspective.

The processes and expertise employed by designers in eliciting user requirements and representing them in a conceptual model has received comparatively little research attention. A deeper understanding of this process can provide useful insights for aiding pedagogy in this area, for building knowledge-based systems to support and, perhaps, for partially automating database design. With the advent of end user computing (traditional end users being engaged in application development and systems design with the availability of easy-to-learn and use hardware and software), database design is not restricted to well-trained, experienced designers. This necessitates greater diffusion of conceptual modeling and design skills across a cross -- section of end users. Such diffusion can be greatly facilitated if the skills and expertise involved in conceptual modeling can be subjected to thorough scrutiny and analysis. This paper attempts to enhance our understanding of the cognitive processes underlying conceptual data modeling through an exploratory investigation into the similarities and differences between experts and novices engaged in such a task. It employs protocol analysis, a process tracing methodology which has been successfully used in a number of other domains. We proceed to synthesize a process model of problem solving in this domain and use the model to illustrate expert-novice differences.

This paper is divided into six sections. The next section presents a survey of previous studies that have motivated this research. In the third section, the characteristics and distinctive features of conceptual data modeling as a problem solving task are outlined. The fourth section presents the methodology, profiles of the experts and novices who participated in the study, and a description of the task presented to the subjects. Section 5 is devoted to the analysis of data in the form of verbal protocols. The process model developed is also included in Section 5. The sixth section discusses the results obtained and synthesizes them with the findings concerning expert-novice differences in other domains. The final section outlines the implications and conclusions of this study.

2. PREVIOUS RESEARCH

A growing body of research has examined the differences in the problem solving processes employed by experts and novices in a wide variety of domains. Studying such differences can contribute to a deeper understanding of what the expert does differently from the novice to account for the generally observed superiority of the expert (Larkin et al. 1980). While some of these differences may be applicable only to the specific domains in question, it is also possible to account for some generalizable differences across domains. We proceed to review the prior research in a domain-specific manner and then to synthesize these results in a more general fashion to the extent it is possible.

The superiority of experts based on efficient processing of large amounts of information was one of the first expert-novice differences to be documented. In the game of chess, Chase and Simon (1973) found that chess-masters (comparable to experts) worked with familiar configurations of several chess pieces recognized as distinct "chunks" that could be evoked from memory with little effort. Performance differences between experts and novices in terms of their ability to recall the position of chess pieces on the board were not found when the pieces were randomly placed. Similar information-processing behavior by experts has been reported in the game of Go (Reitman 1976), bridge (Engle and Bukstel 1978) and in solving physics problems (Larkin et al. 1980). Larkin et al. also speculated that the greater speed with which the experts solved physics problems is related to their ability to execute the problem solving steps in compiled form as opposed to interpreted form. Chi, Feltovich and Glaser (1980) and Larkin (1983) also found that experts tended to categorize physics problems into standard types based on fundamental principles of the domain.

Empirical studies of skill differences between experts and novices in the context of information systems have tended to focus on programming tasks. Gugerty and Olson (1986) investigated expert-novice differences in debugging computer programs written in LOGO and Pascal. They found that experts were faster and more successful at finding bugs and suggested that the experts' superiority could be traced to a more comprehensive understanding of the program and their consequent ability to generate and test high quality hypotheses concerning the bugs. It should be noted that the novices in this study did not do qualitatively different things as compared to the experts; it is just that the experts were able to tap into their knowledge structures to work faster and more correctly. These findings are consistent with the results obtained by Jeffries (1981, 1982). In her study, the experts spent a larger proportion of their time in the comprehension of the program to develop a more complete representation of the program besides remembering the details of the program better. These findings are also in accordance with the results obtained by McKeithen, Reitman and Reuter (1981), who found that expert computer programmers are able to process large amounts of relevant information compared to novices because of their ability to meaningfully organize information into useful chunks.

In studying expert-novice differences in the area of mathematical programming, Orlikowski and Dhar (1986) confirmed several results obtained in physics problem solving. They also proposed a differentiated model of knowledge organization between experts and novices. Experts' concepts were found to be more finely differentiated than those of the novices. This differentiation both at structural and semantic levels enabled them to categorize problems into standard formulations and facilitated the association of problem features with appropriate meaning.

The use of accounting information in financial decision making has been studied by researchers using protocol analysis (Bouwman 1982, 1984). Dillard (1984) provides a review of this research. Bouwman (1982) compared the decision making processes of experts and novices engaged in a financial analysis task. The verbal protocols obtained were split up into distinct decision making "processes," each consisting of a goal and one or more activities. Such processes constituted the basic units of the Problem Behavior Graph that provided the trace of the subjects' decision-making behavior. Bouwman reported that, at the global level, experts and novices used similar decision making processes. However, the relative frequencies of specific processes were significantly different. Experts tended to periodically summarize the results and formulate useful hypotheses. It was also reported that the financial analysis task can be decomposed into three non-contiguous phases: (1) examination of given information, (2) integration of observation and finding, and (3) reasoning. The most significant differences between experts and novices occurred during the reasoning phase (Bouwman 1982, 1984).

The review of the results of the expert-novice problem solving comparisons from a cross section of domains presented above cannot fully endorse the claim of Larkin, et al. (1980, p. 1336) that "expertness probably has much the same foundations wherever encountered" except at an abstract level.

The aspects that have been found to be common across most of such studies include (i) the ability of experts to process large amounts of information into meaningful chunks; (ii) the consequent facility to trigger such structures with little effort; and (iii) categorization of problems into standard types based on underlying domain principles. However, previous research is inconclusive as to whether the experts and novices use the same models or similar models differently (Bouwman 1982); or if they possess a differentiated model as compared to novices (Chi, Feltovich and Glaser 1981; McKeithen, Reitman and Rueter 1981; Orlikowski and Dhar 1986).

3. CONCEPTUAL MODELING AS A PROBLEM SOLVING TASK

Conceptual modeling involves the representation of the entire information content of the database being designed in somewhat abstract terms in relation to the way in which the data is physically stored (Date 1986). Essentially, it is the process of identifying entities, relationships between the entities, attributes, and categories. As suggested previously, a number of methodologies and data models have been proposed in the database literature for developing such a model. While they differ in the specific notations proposed, there are few basic "facets" common across these representation schemes. Batra, Hoffer, and Bostrom (1988) have identified the following facets as comprising the typical set of constructs that the modeler will need to work with in building the conceptual model against a set of given information requirements: entities, relationships qualified by degree and connectivity, identifiers, descriptors, and categories. Different instances of a facet have the same representation. Different facets have different representation. Figure 1 provides a tree representation illustrating the notion of a facet and showing that the conceptual model can be considered as composed of different facets which are shown as leaves in this tree.

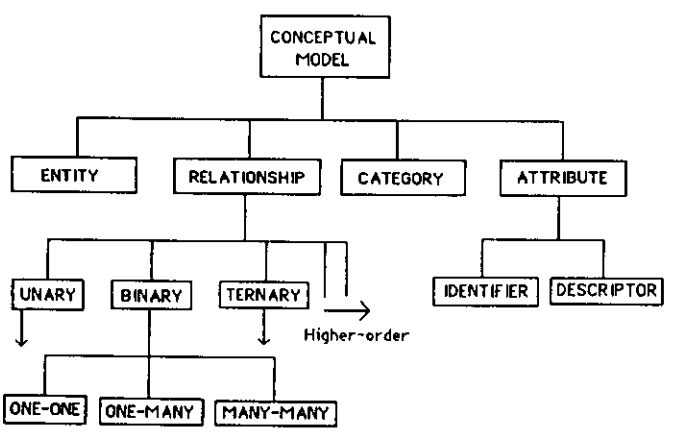

Figure 1. Components of Conceptual Model

In this view, conceptual modeling is the process of identifying these facets as they apply to the given situation and representing them in an interrelated fashion. The notion of a facet permits a more micro-level analysis of a conceptual data model than would be possible otherwise. This becomes useful in carrying out a finely-grained assessment of the performance of experts and novices engaged in a conceptual modeling task in which the subjects need to identify instances of such facets in the given situation and represent them in accordance with the convention with which they are familiar. These conventions could range from a relational representation to more semantic representations such as the extended entity relationship (EER) and object-oriented models. It should be noted that the modeling task in this study is somewhat different from the ones in which expert-novice problem-solving process differences have been studied previously. The idea of unique correct solutions is generally central to domains such as physics, linear programming, puzzle solving, and program debugging where the task usually involves focussed thinking in the narrowly defined problem spaces. The conceptual data modeling problem, in most realistic cases, involves a more open-ended problem space in which the modeler is forced to engage in both broader conceptual thinking as well as focussed problem-solving activities. It is more realistic to think of the solutions (conceptual models developed) in terms of degree of correctness than a unique correct solution. In this respect, the domain resembles the accounting problem studied by Bouwman (1982). We view the differences in domains as an opportunity to add to the available body of knowledge on expert-novice differences across multiple domains.

4. RESEARCH METHODOLOGY

4.1 Research Strategy and Procedure

As mentioned previously, the study employed the protocol analysis methodology for data gathering and analysis. Ericsson and Simon (1980) have convincingly argued that the verbal reports obtained provide the means to understanding the cognitive processes of subjects involved in performing tasks or solving problems in different domains. This approach becomes particularly useful in domains in which the task is relatively complex and open-ended since the verbal data enable us to study in greater detail the intermediate stages of such processes.

The focus of the protocol analysis methodology is on individual, and not aggregate, behavior. The sample size, therefore, is not an issue of concern. In fact, verbal reports collected from a single subject can provide a wealth of data. It is for this reason that this study selected only two subjects each for the novice and expert categories. However, care was taken in the selection process of the novices and experts since the quality of the verbal reports was critically dependent upon this step.

The subjects were provided with a case prepared for the study. They were asked to prepare a conceptual representation of the problem using the relational or any other model and to concurrently verbalize their thoughts as they proceeded with the task. They were expected to show all the rough work as they developed the solution. One of the authors acted as the user, so as to provide clarifications if needed. The subjects were given a sample demonstration of "thinking aloud" as the conceptual model was being developed. It was observed that the subjects quickly adapted to this mode of working. The verbal protocols provided by the subjects were audiotaped. These protocols were transcribed from the tapes into a text document by a trained secretary.

4.2 Subject Profile

Four subjects -- two novices and two experts -- participated in the study. The novices in our study were graduate students who had completed an introductory graduate database course but did not have extensive design experience. The novices had learned conceptual data modeling, and logical and physical database design, and had used standard relational and network DBMSs to define and manipulate data.

One of the experts was a doctoral candidate who was at the point of completion of his doctoral dissertation which dealt with logical database design. He had completed advanced courses in Database Management Systems and had reviewed the design of several large, real world databases. The second expert was an employee of a software company which specialized in the design of object-oriented databases. He had a Master's degree in Computer Science and had four years' work experience in database design.

4.3 The Task

The case that provided the information requirements for the conceptual modeling task for this study was prepared by the authors. The case description was adapted from a real application for which a system had been developed. One of the authors had served as a reviewer for the design of the system. The case included the following facets: entities, unary many-many, binary one-many, binary many-many, and ternary many-many-many relationships, categories, identifiers, and descriptors. The case was a semantically rich application that enabled the comparison of expert and novice problem-solving behavior.

5. RESULTS

The verbal data in the form of transcripts and the written trace on worksheets were initially analyzed by the authors to identify and document the stages the subjects went through from the initial reading of the case to the completed conceptual data model. This was intended to provide a working model of the process which would permit a more detailed analysis of the data. Such a model becomes useful in analyzing lengthy protocols of 12 to 14 pages dealing with relatively open-ended tasks (Bouwman 1982). In identifying such a working (process) model, we were guided primarily by the ANSI/SPARC architecture which explicitly recognizes distinct levels of abstraction in conceptual database design and implementation. This study focused on the translation of user requirements (external view level) into a conceptual data model. The subjects seemed to operate at three distinct levels of abstraction and they iterated among these levels over the time they worked on the task. These levels were applicable to both experts and novices though the frequency and pattern of iterations and the total time spent at each level differed between experts and novices. Figure 2 presents the process model. The specific activities at each level are described below.

1. **Enterprise level:** During the enterprise phase, the subject would read, contemplate, comment, elicit user requirements (from the simulated user), seek clarifications, or establish connections. The focus at this level is on developing a reasonable understanding of the problem domain.

2. **Recognition level:** At this level, the subject would focus on some specific aspect of the user requirements and try to understand the sub-problem at hand. This would trigger the appropriate knowledge structures in his repertoire.

3. **Representation level:** The representation phase constituted the operationalization of the subject's understanding of the structure into a conceptual data representation using the relational (or any other data) model.

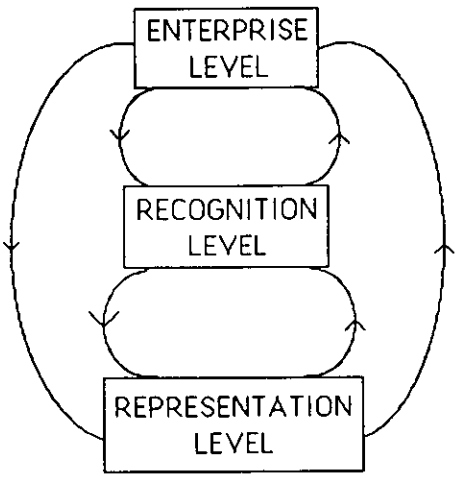

Figure 2. Process Model of Conceptual Database Design

The following examples illustrate the differences between the three levels. Suppose that one is modeling the fact that an equipment is composed of parts which in turn may be composed of other parts. At the enterprise level, one may expect a protocol segment of the form:

> Now, I am looking at parts. This example deals with centrifugal pumps...it is composed of body, impeller, and so on. Then, a part is made of smaller parts...I guess I have a question. Can a sub-part go into many parts?

During the recognition phase, the subject may articulate some key phrases which suggest his understanding of the underlying structure of the description under focus.

> So it seems that there is a recursive relationship. But it has to be captured in tabular form. I guess if I have a table for parts, one for the relationship between parts and sub-parts...and I guess, I need one for the relationship between parts and equipment.

The beginning of the recognition level may be marked by an expression of uncertainty about an aspect of user requirements. The recognition phase can be long and, at times, arduous if the subject has difficulty understanding the situation.

> I am having difficulty understanding this...it does not say how many levels of parts. But do we have to know that? I guess not. I must find a way to capture this irrespective of the number of the levels. Maybe, I should create another relation. This would connect equipment, part and sub-part....No, actually, there are two relations.

The representation level is usually a mechanical process where the designer gathers the complete information for the situation under focus and actually develops the representation.

> I am now creating a relation for the relationship between parts and sub-parts. I used part-code as the identifier for part...so this relation will have part-code and sub-part-code concatenated as the identifier...there are no other attributes for this relation.

The model in Figure 2 shows the three levels -- enterprise, recognition, and representation -- found in the protocols collected from novices and experts. It was found that a subject would typically stay at one level for some time before moving on to the next, and he could move from a given level to any other level. This is shown in the figure by the arcs which connect any pair of the three levels. At times, a subject would shift from the enterprise level to the representation level without going through the intermediate recognition level. This was especially the case when the easier aspects of conceptual modeling, for example, descriptors, were modeled. Conversely, one could return from the representation level to the enterprise level to seek more information or to validate the representation. One could also shift back from the representation level to the recognition level if, for example, it was detected that the hypothesized underlying structure was incorrect. Obviously, the representation level was absent if the subject failed to recognize a particular part of the structure. Further, the representation would be incorrect if there was an error in recognizing the underlying structure of the situation.

5.1 Analysis of Data using the Process Model

Once the model was identified, each protocol was carefully studied and partitioned and classified into segments such that each segment corresponded to one of the three levels. This was achieved by identifying transition points between the levels. Each audiotape was rerun so that the transition points could be time-stamped. A typical subject had thirty to forty transition points. Thus, the duration of each instance of a phase (that is, the difference between adjacent timestamps) was found. Also, by aggregating the phase durations of all instances of a level in a given protocol, the total time spent in each phase was calculated. This analysis revealed some interesting similarities and differences between the novices and the experts.

While the process model introduced earlier was generally applicable to experts and novices alike, there were substantive differences in the emphasis placed by them on each of the levels and the patterns of iterations between them. For the most part, experts tended to work on one part of the case at a time, seek clarifications, integrate all the relevant information, recognize the similarities between the situation which triggered the appropriate knowledge structures (presumably from the long-term memory), and then proceed to represent the information. At the enterprise level, the experts' focus was on developing a holistic understanding of the problem by asking questions of the simulated user, if necessary. At the recognition level, the experts seemed to effectively categorize the information requirements into known abstractions. They then proceeded to represent the information based on the conventions they were most familiar with (relational for E1 and object-oriented for E2). They seemed to cycle through the levels, typically starting at the enterprise level and returning to the same level for a different part of the case without much iterative backtracking.

The novice behavior was considerably different. Their protocols did not reveal a focussed effort at integrating the information and filling the gaps by seeking clarifications at the enterprise level. In attempting to recognize certain requirements, the novices lacked the categorizing terminology and knowledge of the expert, albeit in degree. Much of the novices' effort was directed toward understanding the underlying structure and requirements. When they encountered a situation they were unfamiliar with, they would skip over to another part of the case. This resulted in a considerable amount of cycling between levels and backtracking. We now turn to a qualitative analysis of the data by each individual facet.

5.2 Qualitative Analysis at the Facet Level

The qualitative analysis revealed both similarities and differences between the novices and experts in modeling various facets in the case. One of the similarities pertained to the modeling of the entities and the identifiers and descriptors of the entities. In fact, in most cases, the subjects had little or no problem classifying an object as an entity or an attribute. Although the distinction between an entity and an attribute seemed intuitive, the general rule

that the subjects seemed to use to make this distinction was as follows:

> If an object has descriptive information, then treat it as an entity, else if an identifying name and number is adequate, treat it as a descriptor.

Another similarity that was found in the study is the relative ease with which they modeled binary relationships. In fact, the recognition phase seemed to be generally missing when the binary relationships were captured. For example, the following is an excerpt from one expert's protocol while the subject was engaged in defining the entity CUSTOMER and capturing the binary relationship between the entities EQUIPMENT and CUSTOMER.

> So I will use customer name as identifier. Since we need to link the equipments to the customer, we need the customer number or the customer name...Okay, I'll just call it a customer name.... Customer name as the key...it should be there also in the equipment.

In general, however, the representation of a binary relationship seemed so mechanical that the subjects did not spend much time at the recognition level.

There were similarities, too, between the way the novices and experts modeled the ternary relationships. There were three views included in the case which after normalization could be considered as two ternary relationships -- one between date, equipment, and mechanics, and the other between date, equipment, and parts. Depending on the way one understood the case, one could treat it as a four-way relationship between date, equipment, mechanic, and part. Since subjects generally used the relational model, the main task for representing these relationships was to determine the concatenated key. No significant differences were found between the processes of novices and experts as they modeled the ternary relationship. The general procedure that both experts and novices seemed to use to model ternary or higher degree relationships was as follows:

> If a user view involves identifiers or descriptors of more than two entities, then include the identifiers of the entities in a relation and exclude the descriptors of the entities. Next, find information about the connectivity of the involved entities from the user report and from querying the user. This information is used to determine the concatenated key of the relation. Finally, ensure that the descriptors are dependent on the primary (concatenated) key of the relation for the ternary (or higher degree) relationship.

There were, however, notable differences between the novices and experts in many respects. The novices did not show strengths at any of the three levels. They did not attempt to develop a complete understanding of the business and had considerable difficulty recognizing the key aspects of the user requirements. In particular they had difficulty in modeling unary relationship and categories. For example, the following excerpt revels one novice's struggle with the unary relationship which captured the description that parts are composed of sub-parts.

> The equipment is comprised of parts which may consist of sub-parts. I am not sure how to put this in yet.
>
> [Later] Part bearing block...I don't understand what this part means yet. I'll skip this part and go to the next sentence.
>
> [Later] Instead of calling it sub-parts, we may be able to call it...just parts, and there should be descriptions of the parts, price, and weight. Types and parts would be the key. OK...OK...alright... wrong...part...part code...exclude these parts... OK...I think we should have another entity called sub-parts....That identifies the parts that make up the sub-parts.

The other novice showed weak recognition of the unary relationship but did not proceed to the representation level.

> Okay, each part...there looks like there has to be some kind of sub-part also. And it would probably have the part code as well as the sub-part code, I guess. I'm just going to hold that out.

This contrasted with the confident approach of the expert E1.

> [Enterprise level]: Any equipment is comprised of parts. Okay, now they are talking about parts.
>
> [Recognition level]: So this is a recursive relationship, but still we have to capture it in tables. And since we can order only the lowest level of parts, we are just interested in the lowest level. That is, here the lowest level is the sub-part which is the smallest part that can be ordered.
>
> [Representation level]: I'll create a new parts relation, and different equipment can have different parts....I want to create one more relation which is sub-part.

It seemed that the expert had come across similar situations in the past since, as soon as he recognized that the relationship was recursive, he immediately wrote down the solution. No rules or procedure followed by the expert could be captured. The representation of this problem appeared to be stored in the expert's memory in "compiled" form (Larkin et al. 1980), and the execution was, therefore, almost instantaneous without the usual intermediate steps.

The novice N1 did not appear to be familiar with recursive relationships and consequently had to work his way to a reasonable representation after some backtracking.

Differences were also found in the way the novices and experts modeled the categories. The case involved an entity EQUIPMENT which could be of the three types: centrifugal pumps, reciprocating pumps, and diesel engines. Each category had its own set of attributes. The Expert E1 seemed to know beforehand the correct representation for modeling categories. Again, the recognition phase preceded the representation phase:

> There are three different types of equipments: centrifugal pumps which....Looks like three sub-types: centrifugal and reciprocating pumps, and diesel engines.

The brevity of the recognition level protocol suggests the nature of expertise of the subject. The key phrase "looks like three sub-types" then seemed to trigger the internal knowledge structure of the category concept, and the representation was trivial after that.

It seemed that all the relevant pieces required to model the categories were internalized as a "chunk" in the expert. He used notions similar to the ones mentioned in Smith and Smith (1977) about the generalization concept. For instance, he identified equipment type as a "categorizing" attribute and created separate relations for the three categories. The Expert E2 used an object-oriented notation for the generalization concept. The novices did not seem to be aware of the category concepts. This was evident from the fact that they did not create relations for the subtypes of the equipment described in the case.

Both novices had problems modeling the categories. For example, one of the novices erroneously treated TYPE as an entity. It seems evident that the novice lacked the ability to differentiate between the entity concept and the category concept. In fact, this confusion led him to treat TYPE as an entity rather than as a "categorizing" attribute. The expert's concepts were found to be more finely differentiated, a finding which is consistent with that of Orlikowski and Dhar (1986).

It seems that the rules followed by the expert to model the categories were as follows:

> If an entity can be of different types such that each type has descriptive information in itself, then in the relation for the parent entity, include a descriptor whose values are the various types (categories). Next, prepare relations for each of the types (categories) and specify the descriptors of each type. Finally, use the identifier of the parent entity as the identifier of the various categories.

5.3 Quantitative Results

The quantitative results are presented to support the qualitative findings of the study. Since the sample size is not a significant issue given the methodology. The quantitative results do not have the statistical rigor of the conventional empirical studies. The strength of the protocol methodology is primarily derived from the quality of its verbal data.

One of the variables that was measured was the time taken to complete the task. The novices N1 and N2 took 68 minutes and 46 minutes, respectively (see Table 1). It may be mentioned that the novice N2 could not model a portion of the case and, therefore, the time taken pertained only to the completed portion of the task. The Experts E1 and E2 took 42 minutes and 55 minutes, respectively, which suggests that even though the experts, on an average, took less time than the novices, the difference was quite marginal.

TABLE 1: TIME SPENT AT DIFFERENT LEVELS IN MINUTES

Subject	Enterprise	Recognition	Representation	Total
Novice N1	9.54 14.6%	22.24 33.1%	35.22 52.3%	67.4
Novice N2	9.46 21.1%	13.48 29.8%	22.42 49.1%	46.1
Expert E1	22.27 55.3%	2.50 7.0%	15.15 37.7%	40.32
Expert E2	18.47 34.0%	11.28 20.7%	25.00 45.3%	55.15

The quantitative analysis of the protocols revealed that there were notable differences in the proportion of time spent by the novices and the experts at the three levels. The novices spent most of the time at the representation level (about 50 percent), a fair proportion of the time at the recognition level (about 30 percent), and minimal time at the enterprise level (about 20 percent). The experts, on the other hand, spent most of time at the enterprise level (55 percent and 34 percent) and representation level (about 40 percent), and very little time at the recognition level (7 percent and 20 percent).

The above data suggests that although there were no notable differences in the total time taken by novices and experts, there were critical similarities and differences in the way they apportioned the time into the enterprise, recognition and representation levels. Both experts and novices seemed to spend a good proportion of their time

at the representation level. This was expected since this level involved the actual conceptual representation of the database and it required pulling together all details of the data included in the case and writing them down. Differences were found, however, in the time spent by experts and novices at the enterprise and the recognition levels. Experts seemed to be more concerned with getting the requirements right. As a result, they spent a lot of time asking questions, seeking clarifications, and trying to put the various pieces of the domain together. They spent little time at the recognition level. They seemed to quickly relate the underlying structure of parts of the case to knowledge they already possessed. The novices were not as concerned about getting the requirements right, but had to try hard to recognize and represent the data structures in the case since such processes were not automatically triggered. They lacked the experts' finer differentiation of conceptual modeling concepts and had to struggle to come up with the correct representation of some of the facets.

While the quantitative analysis lends some support to the process model presented in the previous section, the results should be treated with some caution. The results are, however, generally consistent with the findings from the qualitative analysis of the protocols.

6. DISCUSSION

In this section, we attempt to place the study in perspective by comparing our results with those obtained in previous studies on expert-novice differences. The critical issues identified by prior research are listed and the corresponding findings based on our study are discussed.

1. Experts and novices tend to apply qualitatively different (process) models to the task. This differentiation is reflected in the structure of the model as well as the meaning they assign to concepts (Chi, Feltovich, and Glaser 1981; Larkin 1983; Orlikowski and Dhar 1986). Alternatively, Bouwman (1982) has shown that, in the accounting domain, problem-solving processes of experts and novices can be captured by the same model, but the model is applied differently. Our results are more in accordance with that of Bouwman which may be attributed to the relatively open ended nature of the tasks involved in the two studies.

2. It has been found in most studies in this area that experts exhibit richer vocabulary and relative ability to categorize problem descriptions into standard abstractions. This was corroborated in our study. At the recognition level, the mapping of the case to knowledge structures was triggered often for the experts whereas the novices were unable to achieve this for some of the facets.

3. Hinsley, Hayes, and Simon (1983) have reported that experts have demonstrated the ability to automate some aspects of the problem solving process while the novices often have to work from "first principles." This result was partially confirmed with respect to some aspects of the overall conceptual model.

4. Several studies have argued that experts are able to process and meaningfully organize large amounts of information (Chase and Simon 1973; Jeffries 1981, 1982; McKeithen, Reitman and Rueter 1988). This result was not directly supported; however, the experts in our study demonstrated a more detailed and systematic approach to information gathering before addressing the representation aspects.

5. Most previous studies have underscored the greater preponderance of misconceptions and errors by novices. The novices in this study tended to have more errors in their conceptual models than experts largely due to their inability to map parts of the problem description to appropriate knowledge structures.

7. IMPLICATIONS AND CONCLUSIONS

The findings reported in this paper have important implications for training, research, and development in database design. It has explicated a general process model of conceptual data modeling and explored the differentiated application of this model by experts and novices. The relative superiority in expert performance suggests that the strategies they employed could be profitably used in training novice users and designers. The experts' modeling strategy is characterized by decomposing the overall problem description into meaningful parts, gathering and organizing all relevant information concerning each part, mapping this to appropriate knowledge structures prior to actual representation. While the necessary knowledge structures and facets are emphasized in standard pedagogy, the process of relating them to the detailed problem description needs to be emphasized.

The findings provide guidelines for the development of knowledge-based support tools for the conceptual data modeling task. The existing tools such as View Creation System (Storey and Goldstein 1988) can be augmented to incorporate the generic strategies and heuristics employed by experts and to assist novices in avoiding some of the common pitfalls.

This study has addressed one aspect of data modeling that has not been researched previously: the process of conceptual design as performed by experts and novices. The similarities and differences between experts and novices provide some understanding of the nature of expert

knowledge and experience in this domain. Further work involving a real world data modeling task would contribute to developing domain specific expert and novice models besides discovering the precise heuristics used by experts.

8. REFERENCES

Batra, D.; Hoffer, J. A.; and Bostrom, R. P. "A Comparison of User Performance Between the Relational and the Extended Entity Relationship Model in the Discovery Phase of Database Design." *Ninth International Conference on Information Systems*, Minneapolis, 1988.

Bouwman, M. "The Use of Accounting Information: Expert Versus Novice Behavior." In G. Ungson and D. Braunstein, Editors, *Decision Making: An Interdisciplinary Inquiry*. Boston: Kent Publishing Company, 1982, pp. 134-167.

Bouwman, M. "Expert Versus Novice Decision Making in Accounting: A Summary." *Accounting, Organizations and Society*, Volume 9, Numbers 3/4, 1984, pp. 325-327.

Chase, W. G., and Simon, H. A. "Perception in Chess." *Cognitive Psychology*, Volume 4, 1973, pp. 55-81.

Chi, M. T. H.; Feltovich, P. J.; and Glaser, R. "Categorization and Representation of Physics Problems by Experts and Novices." *Cognitive Science*, June 1981, Volume 5, pp. 121-152.

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

Date, C. J. *An Introduction to Database Systems*. Volume 2, Fourth Edition, Reading, Massachusetts: Addison-Wesley, 1986.

Dillard, J. E. "Cognitive Science and Decision Making Research in Accounting." *Accounting, Organizations and Society*, 1984, Volume 9, Numbers 3/4, pp. 343-354.

Elmasri, R., and Navathe, S. B. *Fundamentals of Database Systems*. Menlo Park, California: Benjamin Cummings Publishing Company, 1988.

Engle, R. W., and Bukstel, L. "Memory Processes Among Bridge Players of Differing Expertise." *American Journal of Psychology*, Volume 91, 1978, pp. 673-689.

Ericsson, K. A., and Simon, H. A. "Verbal Reports as Data." *Psychological Review*, Volume 87, Number 3, May 1980, pp. 215-251.

Gugerty, L., and Olson, G. M. "Debugging by Skilled and Novice Programmers." *CHI 1986 Proceedings*, April 1986, pp. 171-175.

Hinsley, D. A.; Hayes, J. R.; and Simon, H. A. "From Words to Equations: Meaning and Representations in Algebra Word Problems." In Carpenter and Just, Editors. *Cognitive Processes in Comprehension*, Erlbaum, 1978.

Jeffries, R. "Computer Program Debugging by Experts." Paper Presented at the Meetings of the Psychonomic Society, 1981.

Jeffries, R. "A Comparison of the Debugging Behavior of Expert and Novice Programmers." Paper presented at the Meetings of the American Educational Research Association, 1982.

Larkin, J. "Problem Representation in Physics in Mental Models." In D. R. Gentner and Stevens, Editors, *Mental Models*. LEA, 1983.

Larkin, J. H.; McDermott, J.; Simon, D. P.; and Simon, H. A. "Expert and Novice Performance in Solving Physics Problems." *Science*, Volume 208, 1980, pp. 1335-1342.

McKeithen, K. B.; Reitman, J. S.; Rueter H. H.; and Hirtle, S. C. "Knowledge Organization and Skill Differences in Computer Programmers." *Cognitive Psychology*, Volume 13, 1981, pp. 307-325.

Orlikowski, W., and Dhar, V. "Imposing Structure on Linear Programming Problems: An Empirical Analysis of Expert and Novice Models." *Proceedings of the Fifth National Conference on Artificial Intelligence* (AAAI-86), Philadelphia, Pennsylvania, August 11-15, 1986.

Reitman, J. S. "Skilled Perception in Go: Deducing Memory Structures from Interresponse Times." *Cognitive Psychology*, Volume 8, 1976, pp. 336-356.

Storey, V. C., and Goldstein, R. C. "A Methodology for Creating User Views in Database Design." *ACM Transactions in Database Systems*, Volume 13, Number 3, September 1988.