

A CONCEPTUAL FOUNDATION AND INTEGRATION DATABASE DESIGNING MODEL

Mohamed Osman Ali Hegazi

Department of Computer Science, Faculty of Computer Science and Information Technology,
Alzaeim Alazhari University, Khartoum North, Sudan

Received 2013-07-19; Revised 2013-09-28; Accepted 2013-11-21

ABSTRACT

This study proposes a database designing model that works as a foundation for database designing phases and as integration for the database systems. The model starts by fragment the system from top to down and then integrate the different parts of the system using the bottom up approach. The proposed model is a graphic model, where it presents the fragmentations of the system in indexed binary tree and then the integration process follows the concept of tree traversing. The proposed model can be a scientific approach for solving the difficulties of database design in understanding the structure and the behaviors of the application and in integrating of the system parts.

Keywords: Conceptual Model, Database Designing, Binary Tree, Integration

1. INTRODUCTION

The goals of database design are to satisfy the information content requirements, provide a natural and easy-to-understand structuring of the information and support processing requirements and any performance objectives (Elmasri and Navathe, 2010). Database design is a difficult task because it requires full understanding of the application and translating the design requirements into a conceptual model (Huang, 2012). Conceptual modeling is widely recognized to be the necessary foundation for building a database that is well-documented and fully satisfies the user requirements. In particular, from the designer point of view the availability of a conceptual model provides a higher level of abstraction in describing the system process and its architecture in all its aspects. Typically conceptual models rely on a graphical notation that facilitates writing, understanding and managing conceptual schemata by both designers and users (Fettke, 2009).

Conceptual modeling deals with understanding the real world entity or object and representing it in such a way that it can be translated into a logical designing without describing the behavior or the structure of the system (**Fig. 1**). Therefore the designers may face problems in understanding the application structure and behavior, which may leads to difficulties in integrating the system parts, especially when the system is complex or it is a large scale database system.

This study proposes a model that adds two phases to database designing phases, the first phase works as a foundation for database designing where it fragments the system from top to down and demonstrates this fragmentation in graphic notation. This phase gives an understanding for the structure and the behaviors of the application. The second phase is an integration phase that can be used to integrate the system parts into a unified system. The proposed model is a graphic model, where it presents the fragmentations of the system in indexed binary tree and then the integration flow the concept of tree traversing.

1.1. Related Work

The researchers on conceptual modeling first try to present the structural and the behavioral of the application by using a semantic conceptual model (Sanyala *et al.*, 2011). Recent some researches try to solve this difficulties by finding a designing model for each application such as (Yao *et al.*, 2013; Cressent *et al.*, 2013; Yan and Ma, 2013). Yao *et al.* (2013) extended the ER model to enable it to capture time-varying information. While Yan and Ma (2013) introduced three levels of fuzziness into the conceptual data modeling to handling the fuzziness in engineering activities.

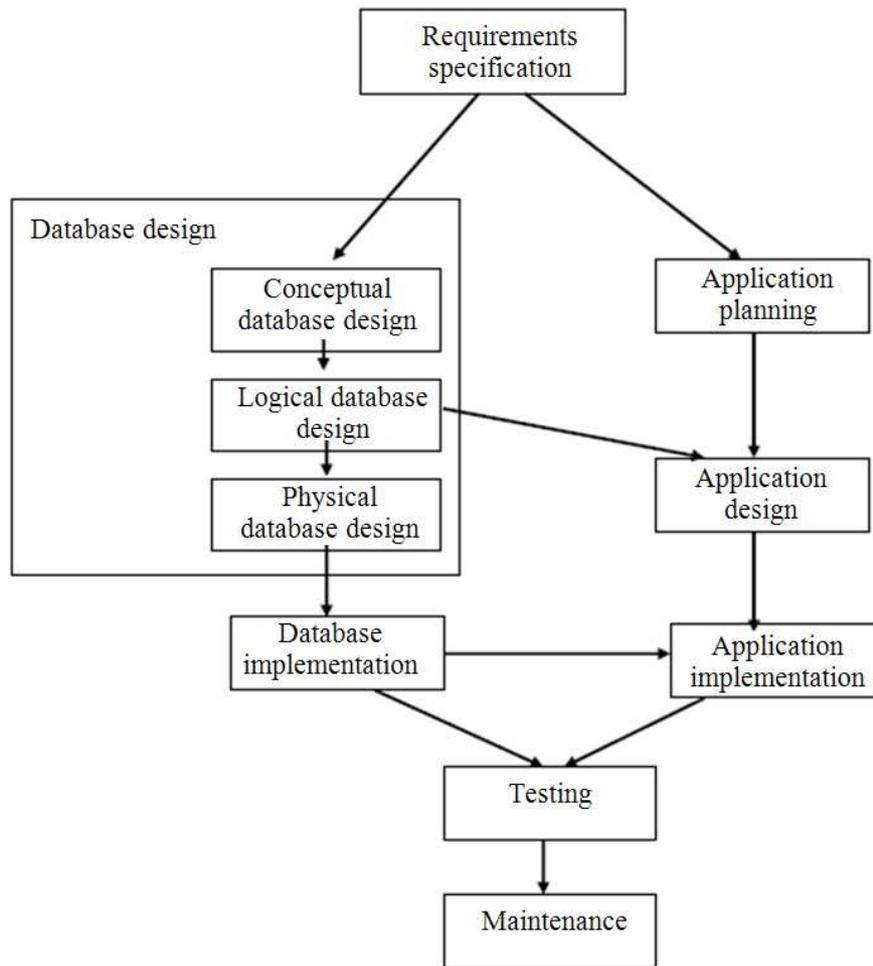


Fig. 1. Database design process (Elmasri and Navathe, 2010)

While other researchers present a fully automated algorithm that eliminates the need for subjective human judgment to understand the complexities inherent in large ER diagrams that restricted the effectiveness of their use in practice (Tavana *et al.*, 2007).

1.2. The Suggested Model

The suggested model adds new two phases to database designing phases:

- The foundation phase: Is an initial stage in database designing phases, it starts before the beginning of the conceptual model; it works as a foundation for the conceptual model without any restriction upon what kind of models can be used. In this phase the system is fragmented from top to down and present this
- fragmentation in indexed binary tree. Therefore this phase presents the structure and behavior of the system
- The schema integration phase: Is the phase in which the system's parts are combined starting from the bottom and moving towards the top

1.3. The Algorithm of the Model

The two phases of the suggested model follow the following steps:

- Step1: Determine the size of the smallest subsystem, which can be done by determining the numbers of parameters, the size of the data, or by determining the atomic system that has no branches or subsystem.
- Step2: Recursively divide the system from top to bottom until reaching the smallest subsystems.

Step3: Each one of the smallest subsystems can then be solved individually.
 Step4: Combine solved subsystems of the same level to obtain the system on the top level. This process will finally produce the complete solution for the main system.

Step 1 to 3 are for the fragmentation phase (foundation) while step 4 for the integration phase.

1.4. The Foundation Phase (System Fragmentation)

In the foundation phase we start fragment the system to sub-systems using the top down approach and present this fragmentation in indexed binary tree.

1.5. The Indexed Binary Tree

The suggested model is a graphic model; it presents the fragmentations of the system in indexed binary tree as shown in Fig. 2.

Figure 2 shows the graphical presentation of the system fragmentation, which could be further explained as follows:

- Each node in the tree is represents a main system or a subsystem
- The main system is on the root of the binary tree and it has the index of 1

- Each node in the sub-trees is a subsystem and it has a unique value that represents the index of its subsystem (the node number preceded with a sequence of numbers separated by points to represent the level and the relation of this node with the other nodes)
- The leaves of the tree represent the final subsystems that result from dividing (fragmenting) the system and therefore they are the result of the phase

At the end of this phase we will reach two important achievements:

- The first achievement is specifying the smallest subsystems (the leaves) with their relationship specified clearly by unique values (indexes). The importance of these subsystems is that; it incorporates all the designing and implementation processes
- The second achievement is specifying the structure and the hierarchy of the system components together with their relationships. The importance of this result is that:
 - It solves the difficulties of understanding the structure and the relationship of the application
 - It makes the integration process easier because all the integration processes are based on the structure of the system and the relations between its parts (the subsystems)

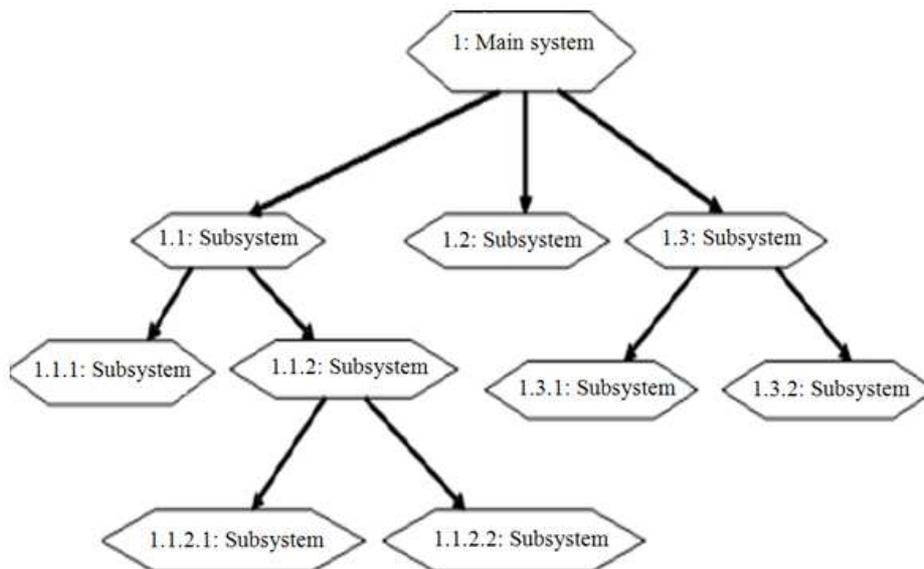


Fig. 2. The indexed binary tree (system fragmentation)

After this phase the designer can start designing the small systems only (the leaves) using any designing models needless to be aware about the components of the system or any understanding of the application and the design requirements of the main system.

In designing any one of the smallest subsystem (the leaves) the indexed value must be involved in all steps of the designing process. This can be done by adding this value to any entity, class or table to specify the belongings. Within this study we also suggest making a level 2 of the indexed tree (Fig. 3 for ER designer and Fig. 4 for object oriented designer).

1.6. The Integration Phase (Schema Integration)

After designing the databases of the subsystems (the leaves), a methodology for integrating the schemas into a global database schema is required. According to the indexed binary tree, the integration process can follow the concept of tree traversing.

Hence, one can start merging the subsystems database from bottom towards the top. This way makes the suggested model suitable for applying any one of the three integration strategies including Binary Ladder Integration, N-array Integration, or Binary Balance (Elmasri and Navathe, 2010). The reason is that, these strategies are based on tree concept; however, the binary balance strategy can be the default integration strategy unless there is no special relation among the parts of the system due to the fragmentation process used in this study's approach. As shown in the above section, the subsystems in each sub-tree are brothers and they are completely related to each other and connected. Figure 5 shows how binary balance strategy can be implemented in our model.

The final database designing approach according to the findings of this study could be put together as shown in Fig. 6. As we can see in this figure there are two additional phases added to database designing phases.

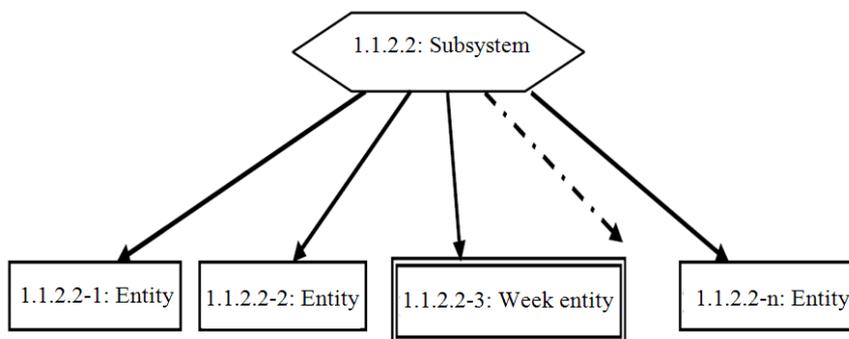


Fig. 3. The indexed tree level 2 for ER

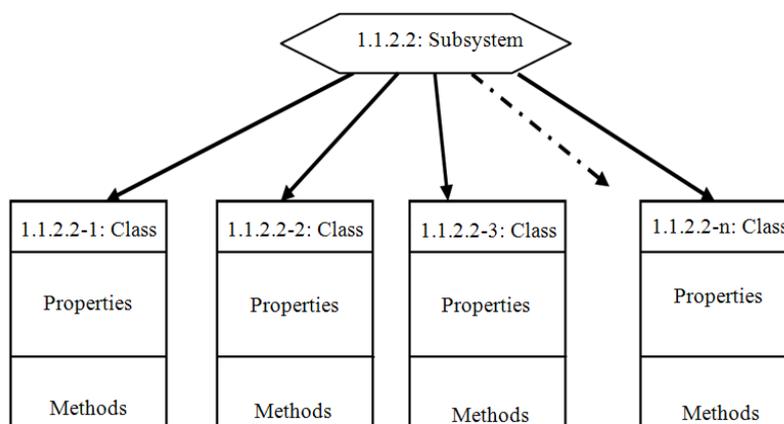


Fig. 4. The indexed tree level 2 for object oriented

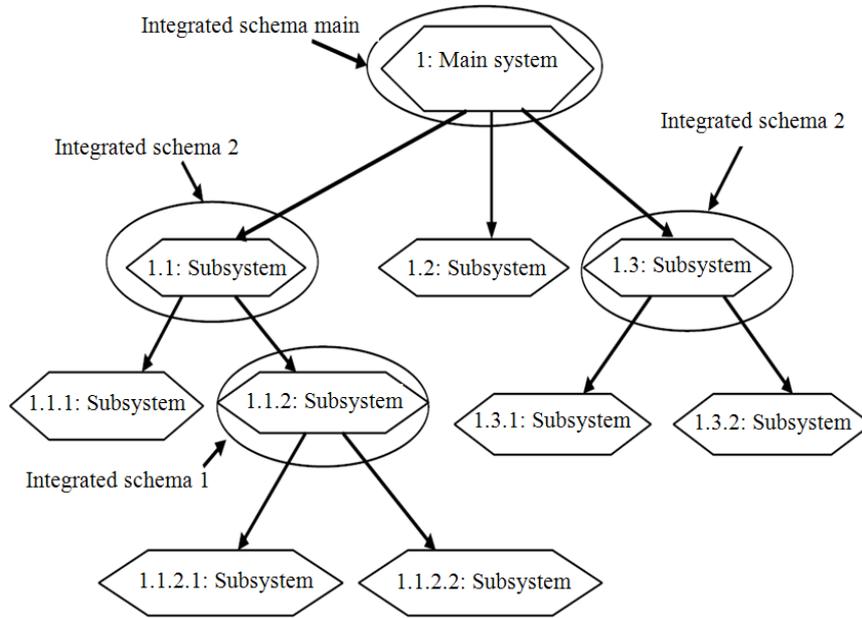


Fig. 5. System integration using binary balance strategy

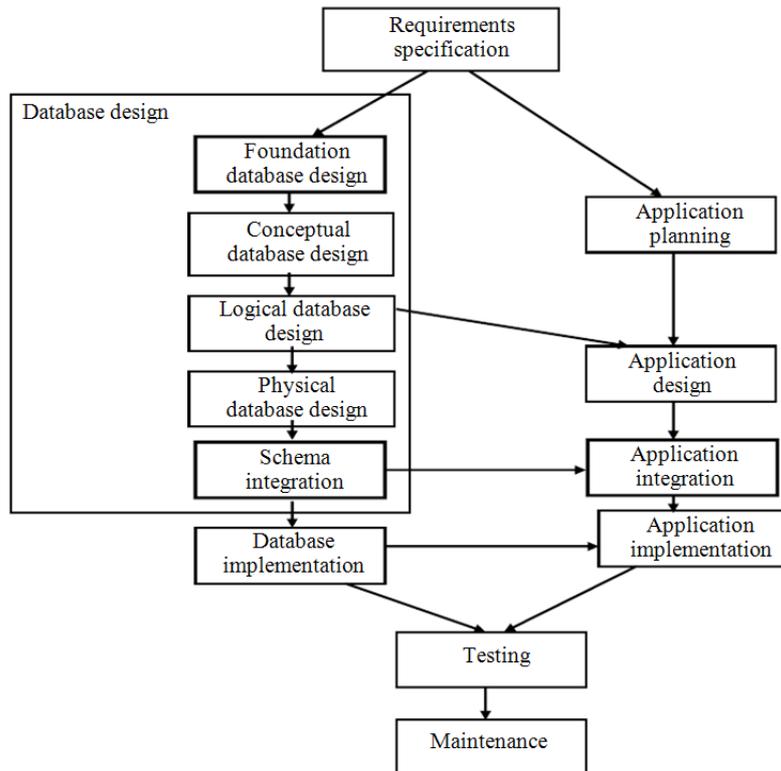


Fig. 6. Database designing process according to our suggested model

2. CONCLUSION

Database design is not easy task especially when the system is complex or contains a large amount of data, where it is often difficult to take all the system data as one unit. This difficulty has made designers and researchers seek solutions and have taken the character of the problems or solutions for special database systems mainly they generate a designing models for each application according to the behaviors of that application, without using standard solution that can be a suitable for all kinds of applications.

In this study, a model that works on all types of database systems is proposed, introduced and described. It is based on a scientific approach that links all the database designing phases and makes it possible to foresee all system data, even if the data are hierarchical and complicated. It relied on the fragmentation of the system systematically and then re-integrates the fragmented parts of the system into one unified system after assuring individually designing the subsystems.

3. REFERENCES

- Cressent, R., P. David, V. Idasiak and F. Kratz, 2013. Designing the database for a reliability aware model-based system engineering process. *Reliab. Eng. Syst. Safety*, 111: 171-182. DOI: 10.1016/j.ress.2012.10.014
- Elmasri, R. and S. Navathe, 2010. *Fundamentals of Database Systems*. 6th Edn., Addison-Wesley, ISBN-10: 0136086209, pp: 1200.
- Fettke, P., 2009. How Conceptual Modeling Is Used. *Commun. Assoc. Inform. Syst.*, 25: 571-592.
- Huang, I.L., 2012. An empirical analysis of students' difficulties on learning conceptual data modeling. *Acad. Inform. Manage. Sci. J.*
- Sanyala, A., A. Sarkarb, S. Choudhury and W.D. Baed, 2011. Graph-semantic based web data model: Conceptual design to logical representation. *J. Comput. Methods Sci. Eng.*, 11: 77-88. DOI: 10.3233/JCM-2011-0379
- Tavana, M., P. Joglekar and M.A. Redmond, 2007. An automated entity-relationship clustering algorithm for conceptual database design. *Inform. Syst.*, 32: 773-792. DOI: 10.1016/j.is.2006.07.001
- Yan, L., and Z.M. Ma, 2013. Conceptual design of object-oriented databases for fuzzy engineering information modeling. *Integr. Comput.-Aided Eng.*, 20: 183-197. DOI: 10.3233/ICA-130427
- Yao, C., F. Fan, L. Shen and X. Li, 2013. Extending temporally ER model for designing temporal databases with multiple time granularities. *J. Theoretical Applied Inform. Technol.*, 53: 458-458.