

## Developing a Data Modelling Tool to Visualize the Transformation of an ER Diagram into a Relational Schema

Elitsa Arsova

**Abstract:** *In this paper, we present the development of a data modelling tool that visualizes the transformation process of an “Entity-Relationship” diagram into a relational database schema. Our focus is the design of a tool for educational purposes and its implementation on e-Learning database course. A detailed description of system functionalities and algorithm for the conversion are proposed. Finally, a user interface and usage aspects are exposed.*

**Key words:** *E-Learning Course, Data Modelling Tool, Entity-Relationship Model, Relational Model.*

### INTRODUCTION

Data modelling is the most difficult and essential process that represents visually data organization in Database Management System (DBMS). A data model that is wrongfully formulated could make the implemented application ineffective and useless. That is why every database course begins with an introduction of data design concepts and gives a theoretical foundation of “Entity-Relationship” (ER) and relational models. However, some students, especially those involved in distance learning courses, find theoretical notions and rules difficult to understand. A better way for students to comprehend the theory is to put it in practice. Designing small learning units that contain a short theoretical basis, examples, and tasks to be performed appear to be most suitable in distance education [2] as well as in e-Learning education. Hence, the need of interactive data modelling tool for learning arises. Existing data modelling tools allow database designers to create, visualize, and save complex data structures, but they are not suitable for visual explanation of the design modelling principles and stages.

The idea is to design and implement an educational computer-tool that will illustrate the transformation of an “Entity-Relationship” diagram (ERD) into a relational schema. The tool is developed for educational purposes mainly and not as a powerful modelling tool for enterprise projects.

The commercial products [6,7,8,9] offer to users a wide range of functionalities that could be applied for a professional usage, rather than gaining fundamental knowledge. These products are very expensive for educational purposes.

On the other hand, researches in the field of designing data modelling tools for learning are conducted in some universities. A web-based tool is proposed in [5] which is developed to correct conceptual database schema automatically. A student has to design an ER schema and enter it into the system through a user friendly interface specifically designed for it. The tool corrects the design and shows detected errors giving advice on how to solve them. Another approach is an e-learning tool that allows defining and practising with e-learning exercises for event driven process chain modelling. That is a representative for graphical modelling techniques. The tool could automatically mark the student’s solutions by using temporal logic and model checking [3].

### THEORETICAL BACKGROUND

The process of designing a database begins with an analysis of the kind of information the database should hold and the relationships among components of that information. The first stage is to formulate ideas about the information that will be modelled and render them in the E/R model. Then, the abstract E/R design is converted to a schema in the data-specification language of some DBMS. Most commonly, this DBMS uses the relational model. If so, then by a fairly mechanical process the abstract

design is converted to a concrete, relational design, called a “relational database schema”. Fig. 1 suggests how the E/R and relational models are used in database design.

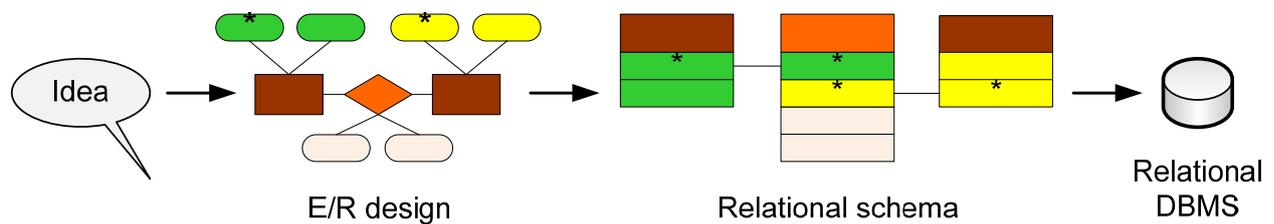


Fig.2 The database modelling and implementation process

In the E/R model, the structure of data is represented graphically as an “entity-relationship diagram” using three principle elements: entity sets, attributes and relationships. An entity is a “thing” which can be distinctly identified [1] and a collection of similar entities forms an entity set. It has associated attributes, which are properties of the entities in that set, and a relationship is an association among entities.

The relational model gives a single way to represent data: as a two dimensional table called relation. Names for the columns of the relation are called attributes and the rows of relations, other than the header row containing the attribute names, are called tuples. A tuple has one component for each attribute of the relation.

The two main rules, converting an E/R design to a relational database schema are:

- Turn each entity set into a relation with the same set of attributes, and
- Replace a relationship by a relation whose attributes are the keys for the connected entity sets.
  - For each entity set involved in relationship R, we take its key attribute or attributes as a part of schema of the relation for R.
  - If the relationship has attributes, then they are also attributes of relation R.

If one entity set is involved several times in a relationship, in different roles, then its key attributes each appear as many times as the number of roles[4].

### DESIGN OF THE DATA MODELLING TOOL

The data modelling tool is designed to offer simple and intuitive interface to the learner. The tool is used on a Databases course and its aim is to illustrate the transformation process of the two diagrams. The functionalities of the educational unit are presented on figure 2.

The learner could represent an entity by drawing a rectangle; an attribute by drawing an ellipse and a relationship with a rhomb. A line denotes the type of the relationship between these E/R elements. The correctness of a drawn E/R diagram (ERD) is validated during the generation of relational diagram (RD).

The learner could also save the drawn ERD and open it later. The saved file extension is .dat, and at this stage of implementation these files could not be opened by another data modelling tool or text editor.

After drawing the ERD or opening it, the learner could generate the RD. The tool draws the RD without any interaction of the user by analogy with ERD.

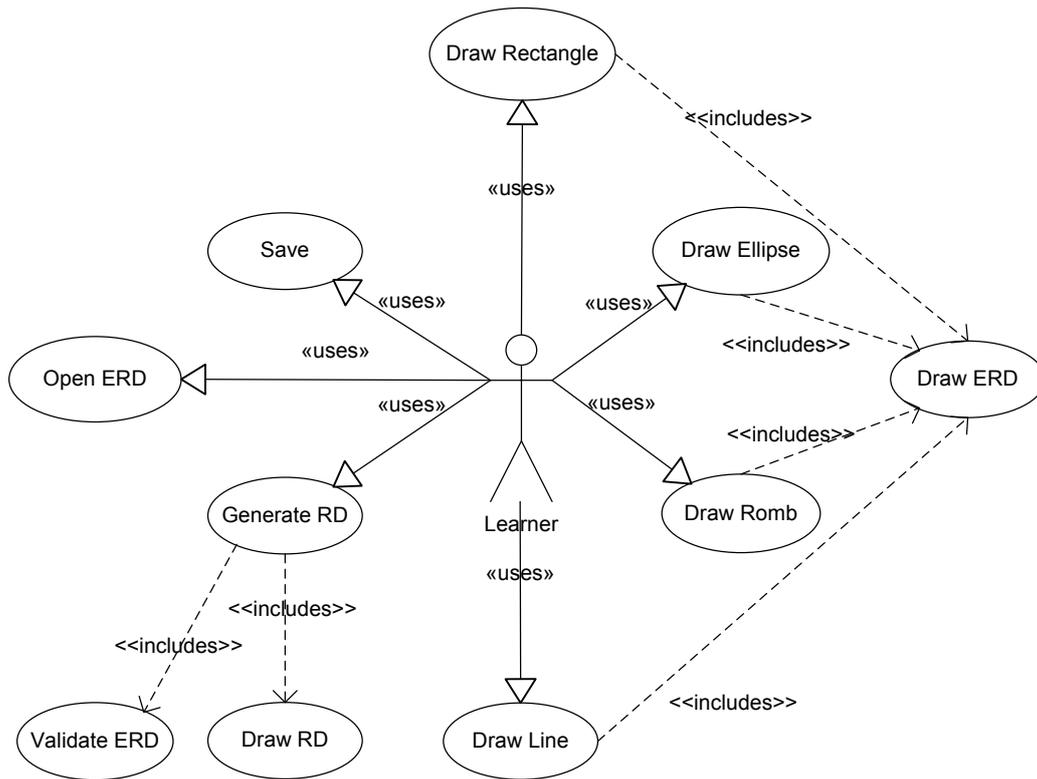


Fig.3 Use Case model of the educational tool

**TRANSFORMATION OF AN E/R DIAGRAM INTO A RELATIONAL SCHEMA**

The variables presented in Table 1 identify the ER and relational schemas. Each schema is described by two variables of type List: one of them belongs to the class Element and the other one to the class Relation.

Table 7 Schema describing variables

| Variable Type        | Variable Name     | Schema Type       |
|----------------------|-------------------|-------------------|
| List<Element>        | m_Elements        | E/R               |
| List<Relation>       | m_Relations       | E/R               |
| List<List<Element>>  | m_SchemeElements  | Relational schema |
| List<List<Relation>> | m_SchemeRelations | Relational schema |

An algorithm for the transformation of an ERD into a relational schema is presented on fig. 3 and described in the next paragraph.

1. Read ER diagram elements (m\_Elements);
2. Create a new RD element of type List<Element> in m\_SchemeElements. This element is a table that will be formed in the next steps.
3. Check the type of element, if it is a rectangle.
4. If so, the title of rectangle becomes a title of created RD element i.e. the table. Subsequently, read all elements in m\_Relations that refer to the rectangle. If the element is not an entity, then go to step five.

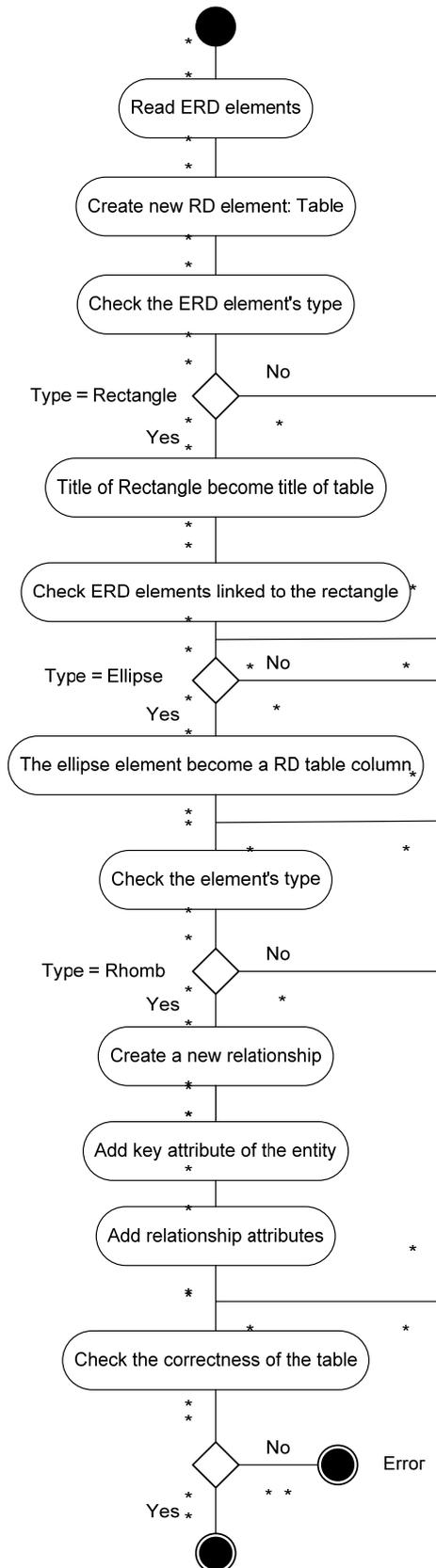


Fig.4 An algorithm for transformation of an E/R diagram into a relational schema

5. Check ERD element which is linked to the rectangle.

6. If its type is an ellipse, then the title of ellipse becomes RD table attribute. Add one more element to the m\_SchemeElements.

7. If the element is not an ellipse, then go to step seven.

8. Check the type of ERD element, if it is a rhomb.

9. If so, describe a new RD element of type List<Relation> in m\_SchemeRelation and a new table is created. Subsequently, read all elements in m\_Relations that refer to the rhomb.

10. If the element is not a relationship, then go to step 11.

11. Add key attributes of the linked entities to the relationship entity.

12. Add relationship attributes.

13. Validate the correctness of ERD table.

The conditions validating the correctness of ERD table are the following:

- Each element of type Rectangle should be linked to elements of type ellipse which number could be between two and eight;
- Each element of type Rectangle should be linked to at least one element which title consists of symbol “\*”. Thereby, key elements of RD are defined;
- Each element of type Rhomb should be linked to two or three elements of type Rectangle.

If one of these conditions is not true, the data modelling tool will alert for an error in the sketched ERD.

### IMPLEMENTATION OF THE DATA MODELLING TOOL

The data modelling tool is experimental and its educational aim is to demonstrate the transformation of an ERD into a RD. A user interface is presented on fig. 4 and fig 5. It is divided into three sections. The learner could design the ERD at the upper right field. The upper left field shows to the user the ERD elements' title and type. This information will help him/her to follow the designing process and also correct or delete its elements. The result of transformation of the ERD is visualized on the section below.

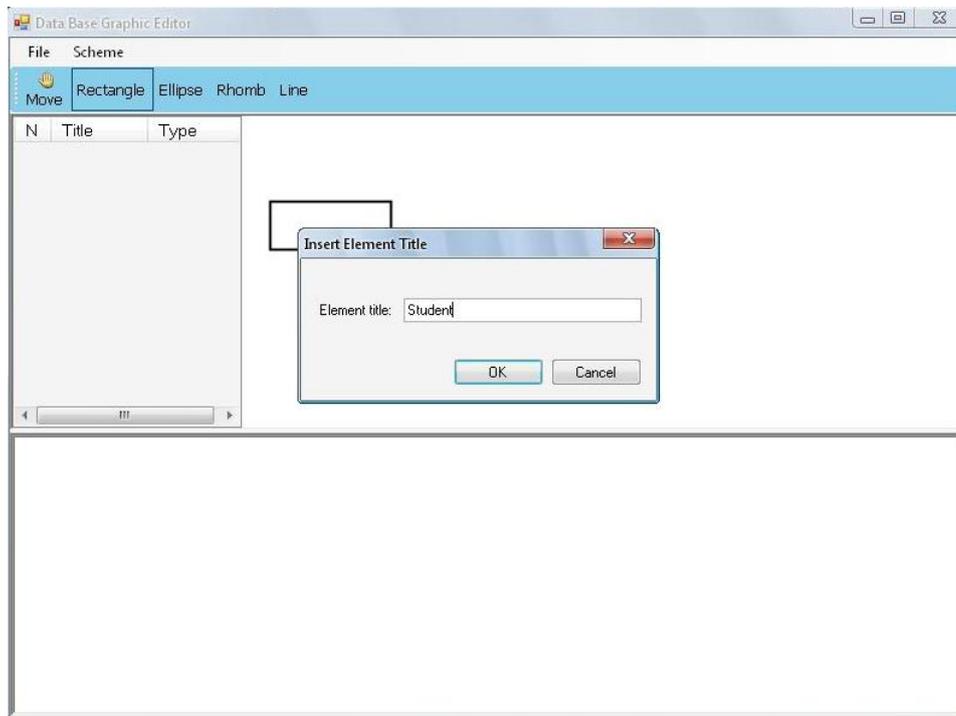


Fig.5 A user interface of the data modelling tool

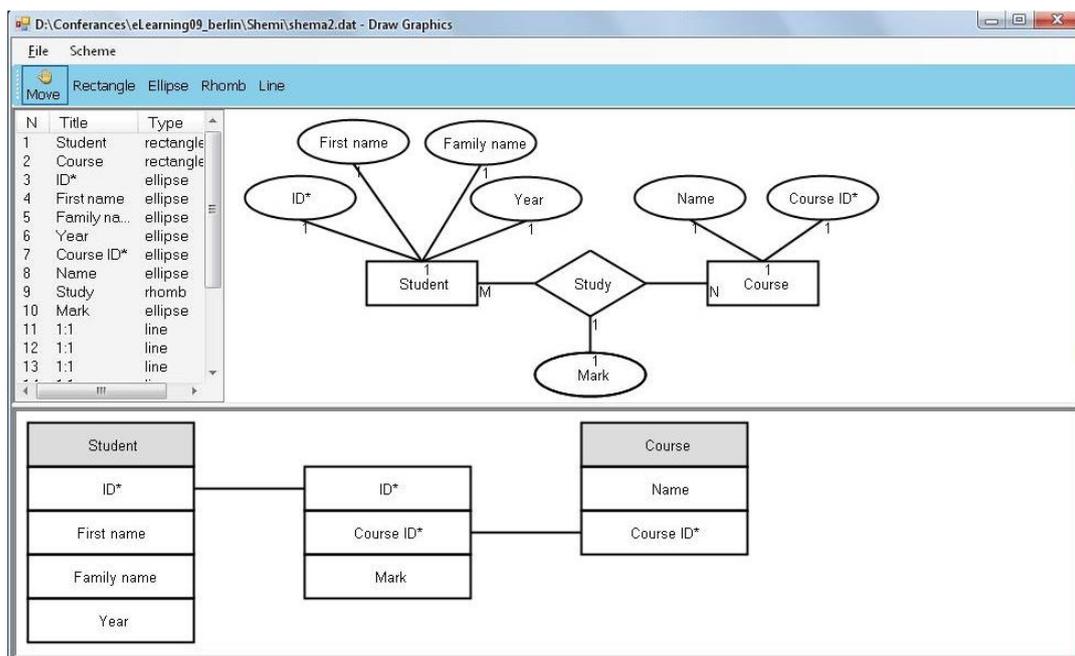


Fig.5 Generation of the relational diagram

The learning unit offers two functional toolbars horizontally designed. There are two menus at the first toolbar: File and Schema. The File menu consists of the following functions: New, Save and Open, which relate to the ERD. The other toolbar consists of five buttons: Move, Rectangle, Ellipse, Rhomb and Line. To draw one of these shapes, we should choose it and then point where to put the element in the upper right field. The next step is to write a title of the element in a pop-up window, which appears after drawing the element (fig. 4).

After designing the ERD, the learner could save it in a file or generate the relational schema. To do the last operation, he/she should choose from the Schema menu the option Generate. The view of the two diagrams is presented on fig. 5.

### **CONCLUSIONS AND FUTURE WORK**

An algorithm for transformation of an E/R diagram into a relational schema is proposed. Its educational purpose is to visualize the theoretical rules. The algorithm is used for the implementation of program modules which are contained in the data modelling tool.

### **REFERENCES**

- [1] Chen, P. The Entity-Relationship Model: Toward a Unified View of Data. is ACM on Database Systems, Vol. 1, No. 1, March, 1976.
- [2] Ivanov, S., J. Peneva. Distance Learning Courses in Computer Science – Initiation and Design. Proceedings of CompSysTech'2007, Ruse, 14-15 June, 2007.
- [3] Markus, S., R. Lackes, C. Börgermann. Using Model Checking to Automatically Mark Graphical E-Learning Exercises. In Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications. Chesapeake, 2008 (pp. 5302-5307).
- [4] Molina, H., J. Ullman, and J. Widom. Database Systems: The Complete Book. USA, Prentice Hall, New Jersey, 2002.
- [5] Prados, F, I. Boada, J. Soler, J. Poch. A Web-Based Tool for Entity-Relationship Modeling. International Conference on Computational Science and its Applications, Glasgow ,2006.
- [6] Avolution. <http://www.avolution.com.au/products.html> (Last accessed 15.07.2009).
- [7] Data modeling from CA. <http://www.ca.com/us/data-modeling.aspx#overview> (Last accessed 15.07.2009).
- [8] DeZign for Databases Database Modeling Tool. <http://www.datanamic.com/dezign/index.html> (Last accessed 15.07.2009).
- [9] SmartDraw – The World's Most Popular Business Graphics Software. <http://www.smartdraw.com/> (Last accessed 15.07.2009).

### **ABOUT THE AUTHOR**

Elitsa Arsova, PhD Student, Dept. of Computer Systems and Technologies, University of Ruse, Phone: +359 (0)82 888 827, E-mail: EArsova@ecs.ru.acad.bg.

### **ACKNOWLEDGEMENT**

This work was partially supported by the European Social Fund and the Bulgarian Ministry of Education and Science under the Operation programme “Human Resources Development”, Grant# BG051PO001/3.3-02/7.