Build Tool to Generate Relational Database from Class Diagram

Anfal A. Fadhil Assistance Lecture Dept. of Software Engineering University of Mousel, Iraq

ABSTRACT

UML has been accepted in both academia and industry as a de facto modeling language to develop software systems. As relational databases are widely used for developing software systems, thus, it is a challenge to derive relational tables from designed models ,In this paper, is building tool in C# language to designing a relational database system based on Object Oriented Analysis and Design that class diagram, by create the document xml that extract from class diagram, based on class name, attributes name , data type the relationships among the classes, multiplicity, Inheritance , And this concepts are not supported in the relational database Therefore , use is algorithm and rules to translate the relational database

Keywords

Class diagram, XMl, Relational Database ,Relationship C# language

1. INTRODUCTION

Database system is important to ensure the data can be stored, updated and retrieved for future use. Data modeling using the Entity Relationship Model has been introduced more than thirty years. However, designing a good database system is still an attractive issue particularly in system analysis and design because of very hard to do consistency checking between system design and database design[1],The power of modelbased engineering is the ability to visualize, analyze and design all aspects of a system. Being able to view and manage information and data alongside other models of a system provides great clarity and reduces the chance of error. [2]

The object oriented and relational technologies are originated in distinct paradigms. The object-oriented technology is universally accepted approach for the development of business applications since it offers more appropriate methods and facilities for modeling of the real world objects. Meanwhile, relational database is still used to preserve the persistence of enterprise data. Connecting these two paradigms is not a straightforward task and necessitates a systematic mapping from one to another. Mapping from object model to relational database demands special attention because it would yield an object-relational impedance mismatch[3]

The first task of a database designer is to produce a conceptual data model that reflects the structure of the information to be held in the database. A common approach to this is to develop an entity-relationship model, often with the aid of drawing tools. Another popular approach is the Unified Modeling Language. A successful data model will accurately reflect the possible state of the external world being modeled[4] in our research a relational database system based on Object Oriented Analysis and Design is presented. The database system is created by the schema table that extract from class diagram, It is based on special algorithms and rules that explain in this paper

2. RELATED WORK

Many research have worked on build relational database from class diagram, in the following section a brief explanation about their works

He, Ying, and Zhang, 2002 have done a research for designing and building a multimedia database. They proposed an idea to identify the classes from the user requirements. Then, they converted the class to schema class declarations using Object Definition Language. They used the Object Oriented Design (OOD) as the guide line to build the database. It was done manually. The key was choosing from the attributes in class and applies the principle base on relationship[5].

Alsaadi (Alsaadi, 2006) in his paper explain that class diagram can be used to produce a schema for system database. He discussed how to check the data structure or data integrities relating to database system using sequence diagram. He suggested a technique using binary class diagram and Object Constraint language (OCL) to ensuring data correctness from class diagram. To show the idea, he gave an example on a sequence diagram derived from the class diagram to ensure data correctness But he did not show the transferring technique to schema database[6] a research conducted by Ali et al. (Ali, Shukur, and Idris, 2007) in their work proposes a tool to check the consistencies of class diagrams. The idea was just to check the name and the numbers of classes, the number of relationship and checking the specification of class object. In their work, they show the technique how to formulate the rules to check the diagram[7].

Another research by Farzaneh et al. are show a formal automatic approach to extract relational tables from class diagrams, adopt on graph transformation systems. researcher have designed some graph transformation rules to derive necessary tables from UML class diagrams[8].

3. RELATIONAL DATABASE

Relational database concept was established in 1970 by Edgar F. Codd and grounded on the set theory to arrange data in terms of rows and columns ,Relational Database is considered as matured earlier than the object-oriented technology became the de facto standard for software development [9]. Most of business entities still rely on relational database management systems (RDBMS) such as Oracle, SQL Server, DB2, Sybase, Informix and etc. to organize, store and manage their enterprise data.

Relational database is based on entities and relationship among them. It is a collection of data items organized as a set of formally described tables from which data can be accessed easily. Relational database is considered as the most robust and reliable database management concepts because it enforce data normalization, where redundancy could be eliminated to ensure that there is only one source for each data element in the system, and therefore increase integrity through relationships [10].

4. THE UML CLASS DIAGRAM

In recent years, the Unified Modeling Language (UML) has emerged as the standard for the representation of software engineering diagrams, The UML class diagram contains classes, interfaces, collaborations, and dependencies, associations and interface relationships, A class is a description of a concept which may have attributes and operations associated with it, Classes are represented as rectangles and relationship between two classes is drawn as a line[11], Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application, it is also considered as the foundation for component and deployment diagrams, the diagrams are used also for

- 1. Describing the static view of the system
- 2. Showing the collaboration among the elements of the static view
- 3. Describing the functionalities performed by the system
- 4. Construction of software applications using object oriented languages

Classes are at the heart of any object-oriented system and describe the different types of objects for any system [12]

4.1 Transformation Process

In this research provided an alternative way on how to design optimum relational database system based on class diagram. the divide the research into two phases. The aim in the first phase is to produce the document xml. Then the document will be used as the input for normalize process in second phase, Where the document is read and then extract the basic information to create the relational database as shown in the figure(1)





When build proposer tool the properties of the class were used to form the relational database as follows

- 1. The name of the class association is the name of the table
- 2. Each attribute in the class is a field in the table with the same name and style as in the figure(2)



Fig 2: Class and the Corresponding Table

- 3. The operation in the class diagram not important in build database relationship between classes
- 4. Method for mapping relationship between classes, type of relationship are association, aggregation or composition. The following algorithm will be used to apply the transformation process from class diagram to relational database

Algorithm :create tables from relationship class diagram[13]

If type of relationship between classes=1:1

Then

create table from class and create new table

set new table name =name of two classes

set attribute name ,type and size of attribute in new table =primary key of tow tables

set key of new table ='PK' (All attribute)

else if type of relationship between classes =1:M then

create table from class

add new attribute in table (M) =primary key of table(1)

set new attribute in table (M)='FK'

If type of relationship between classes==M:M

Then

create table from class and create new table

set new table name =M+ name of two classes

set attribute name ,type and size of attribute in new table =primary key of tow tables

set key of new table ='FK' (All attribute)

else if type of relationship between classes =S:E(Super class and sup class)

then

create table from class

add new attribute in table (E) =primary key of table(S)

set new attribute in table (E)='FK'

end if

5. CASE STUDY

In this section, present class diagram an example of an application for hospital system

(See Figure 3) The tool is proposed in the Figure 4 explain how can any user build relational database from class diagram



Fig 3: Class Diagram for Hospital System



Fig 4: Tool Proposed in C# Language

After the application of the proposed tool on the hospital system, five tables(hospital, manager, person, patient, staff) were created with the same classes names, And The fields of these tables are the same as attribute in the class. Some fields are added in table as a result of the relationships between these classes such as

- 1. (id of person) in patient table, because The patient's class inherits the primary key attribute of the person's class and he become forging key.
- 2. (id of staff) in hospital table, because The relationship between hospital class and staff class 1:M and he become forging key.
- 3. (id of person) in staff table, because The staff's class inherits the primary key attribute of the person's class and he become forging key.

And three additional tables were created

- 1. hospitalmanager: Table because the relationship between the hospital table and the manager table is 1:1 the fields in this table are the primary key field of hospital table and manager table
- 2. mstaffpatient: Table because the relationship between the staff table and the patient table is M:M the fields in this table are the primary key field of staff table and patient table and they become forging key
- 3. mpersonhospital: Table because the relationship between the person table and the hospital table is M:M the fields in this table are the primary key field of person table and hospital table they become forging key

6. CONCLUSION

In this proposer tool presented framework to extract relational tables from UML class diagrams where designed different rules to transform classes to tables, attributes to columns, relationship between classes to new tables and solve inheritance among classes through translated few UML features such as qualifiers, association classes, compositions and generalizations into association relations with additional constraints based on the proposer algorithm and rules and this algorithm and rules reduce error, cost ,and time of translation because the system can run properly and fast without need to convert from class diagram to E-R diagram then create table in relational database, we note number of class and table must be equal unless case relationship between class type one-to-one and many-to-many relationship then create new tables recurring schedule, and the attributes in the class are equal the attributes in the table, additional the attributes used in the relationship between tables(Forging key ,primary key)

7. REFERENCES

- [1] S. Mohd ,R. Parit, Proposal for Constructing Relational Database from Class Diagram, Vol. 3, No. 2; May 2010
- [2] Sparx Systems, Database Models, 2017
- [3] A. Mohd,A.Nur, TRANSFORMING OBJECT ORIENTED DATA MODEL TO RELATIONAL DATA MODEL, International Journal on New Computer Architectures and Their Applications (IJNCAA) 2(3): 402-409 The Society of Digital Information and Wireless Communications, 2012 (ISSN: 2220-9085)
- [4] E.Rames , FUNDAMENTALS OF Database Systems SIXTH EDITIO,2011
- [5] He, Y.L., Ying, Z.L., and Zhang, Y.W. (2002). The Design and Implementation of a Multimedia Database with Object Oriented Analysis and Design Method. Proceeding of 6th International Conference 9 (ICSP '02). IEEE Press, 1035-1038.
- [6] Alsaadi, A, Checking Data Integrity via the UML Class Diagram,2006, Proceeding of International Conference on Software Engineering Advances (ICSEA 06), IEEE
- [7] Ali, N.H., Shukur, Z., and Idris, S, A Design of an Assessment System for UML Class Diagram,2007, Proceeding of 5th International Conference Computational Science and its Applications. IEEE
- [8] M. Farzaneh , R. Vahid, J. Shahram, From Class Diagrams to Relational Tables: A Graph Transformation-based

International Journal of Computer Applications (0975 – 8887) Volume 181 – No. 31, December 2018

Approach ,2011,PRZEGLĄD ELEKTROTECHNICZNY (Electrical Review), ISSN 0033-2097, R. 87 NR 8/2011

- [9] T.Connolly ,C. Begg, Database System A Practical Approach to Design, Implementation, and Management, Addison Wesley ,2010
- [10] Rob, Peter and Coronel, C ,Database Systems: Design, Implementation, and Management, Eighth Edition. Course Technology, Boston MA,2009
- [11] S. Vipin ,P.Ajay, Transformation of UML Class Diagram for Object Oriented Database System, International Journal of Computer Applications (0975 – 8887) Volume 26– No.10, July 2011
- [12] M.Russ , H. Kim, learning uml 2.0, 2015
- [13] K. Wilailuk, L.somchai, An Adaptive Translation of Class Diagram to Relational Database, IEEE, 2009