Unified Modeling Language and Enhanced Entity Relationship: An Empirical Study

Manal Mahmoud Alkoshman

Manalalkoshman@yahoo.com

Abstract

A perfect design of the diagrammatic notations is necessary to communicate between the designer and the user at the requirements analysis stage. The Unified Modeling Language (UML) and Enhanced Entity Relationship (EER) are typically used for designing large systems and applications. On paper, diagrammatic notations can be used to develop or perform maintenance on the application. Any notations used, must represent business rules accurately. At the same time, the notations must be understandable to manager, user and programmer. ER and EER diagrams have been taught in colleges and universities for many years. In recent years, the Unified Modeling Language has appeared for the representation of relational databases. There are many articles in the literature discussing the efficacy of using the UML in modeling relational database systems. The choice between diagrammatic notations does not show any thoughtfulness for an understanding of the human reader. This study focuses on a comparison between the EER and the UML Class Diagram, the cause of the common notations and acceptance among system analysts and programmers. It was proven through an experiment on the sample of students participating in the experiment. The experiment takes the opinion of information technology students to determine which the best diagrammatic notation for them to use. These students will be programmers and system analysts in the future. The results identified through experiment are the favorite for graphic students, and the result can show any relational model closer to the user and manger.

Keywords: *Enhanced Entity Relationship, UML Class Diagram, and relational model.*

1. Introduction

There are many diagrammatic notations that are normally expressed through a variety of shapes (oval, rectangle, box.*etc*). These diagrammatic notations support configuration and design of the database system.

Database systems become more complex, so developers select the diagrammatic notations that accurately identify database system requirements. Not only are these diagrammatic notations different for each presentation in the database, but they also include a set of joint characteristics. The choice between diagrammatic notations depends on the system size and nature. Because of the common notations between system analysts and programmers, the study chooses the UML (Unified Modeling Language) and EER (Enhanced Entity-Relationship) from this notation.

There are several analytical studies conducted on the study of software engineering models and models of object-oriented programming. This paper compares between EERD and UML. The EER is used in conceptual level database design and UML Class diagram is used in OOP. The study reflects the best of diagram notations as determined by IT students. The study presents a paper to students containing a description of EER and UML, and a set of drawings for various database systems required for an information technology student. In addition to that it identifies business rules and determines schema relational database for each model.

The Students decide which of the relational database models explains more details. This study involves a group of students from Department of Information Technology at Mutah University.

This paper is structured as follow : Related work is presented in section 2,Section 3contains sample comparison between UML and ERD, section 4 converts EER representation to an Equivalent UML Class Diagram, section 5 describes the experiment method of the   study, and gives worked example about study. This example discusses how a student chooses from diagram notations, section 6 discuss the data, the limitation in section 7, conclusions and future work in section 8,   and references in section 9.

2. Related Work

There has been much research comparing between different relational designs. In [1] [2, 9] different notations for ER diagrams are compared. Some need more research in ER and EER modeling include the development of more modeling heuristics. EER diagram is extended to the entity relationship diagram continue to evolve to include new symbols to model object oriented concepts. In [2] addresses this notation comprehension issue by proposing an experimental methodology for determining which of two complete notations is easier to comprehend. This methodology has been applied to two types of entity relationship notations: our experiment requires subjects to indicate whether a supplied textual specification of objects and relationships matched each of a set of Chen [10], and SSADM [11] entity relationship diagrams. Bernadette Marie et al [16] comparative between UML and EER notations.

Researchers [12] have also proposed techniques using a UML class diagram that can be transformed into an equivalent EER diagram. Eunjee Song et al [5] have worked on techniques that convert a UML diagram to model relational database. The power of UML is not limited to object oriented. UML is a good language for successful integrating both worlds in [6]. In [16] extends existing approaches by enriching the generated (UML) conceptual schema with a set of OCL integrity constraints and derivation rules inferred from the database schema.

In [14,13,8] development tools that obtain the contextual information of each component in a conceptual database diagram. Shuyun Xu *et al* [13] describe the architecture of ER Draw and its implementation details to illustrate how such a tool can be developed. In [8] developing tool using extension the rational root tool for visualize UML and formal B notations in single environment.

In[3,4,15] aims to present Articulated Entity Relationship (AER) diagram ,which is an extension of Entity Relationship (ER) diagram , Dowming Yeha *et al* in [4] presented a method that extracts an EER diagram from a table-based database with little descriptions for the fields in its tables and no description for keys. N. Mfourga [15] Presented framework for extracting an entity-relationship schema from a set of form model schema of an operational relational database.

Baral Chitta *et al* [7] discusses multimedia database, we need to extend ER to take design of multimedia database.

3. Comparisons between a UML Class diagram and an EER Diagram

Dr. Peter Chen developed the ER diagram in order to build a unified data model. The EERD is an extension of the ERD, and EERD which contains all the properties of ER and adds some special concepts such as inheritance and aggregation.

UML is a standard language for building and documenting software. It is successful in modeling of large and complex systems. UML sponsor the team to communicate and explore possible analysis and implementations, and verification of the design of the programs, especially when the design is object-oriented.

UML focuses on Object Oriented Analysis and Design (OOAD), and contains more than one form. In this study, we will focus on UML class diagram. Both UML class diagram and chan's original ER diagram are notations used to implement analysis on real world through specific instance of an object.

3.1 Class vs. Entity:

The class is used to represent the entity in the UML class diagram. The diagrammatic notations are rectangle box. The rectangle box contains three sections. The class name on the top, the Attribute in the middle and the operation (behaviors) in the bottom of box . figure 1 shows class in UML class diagram.

Module (class name)
Module code: int Module title: string Level: int
Operation ()
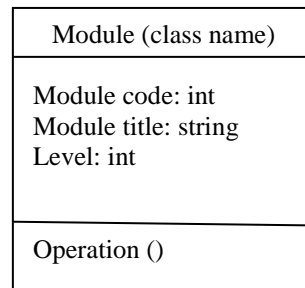
Figure 2. The UML Class Notation

The diagrammatic notations of entity in ER diagram are rectangle box. The rectangle box Contains name of entity and there is no place for attribute in box. Figure 2 shows Entity example.
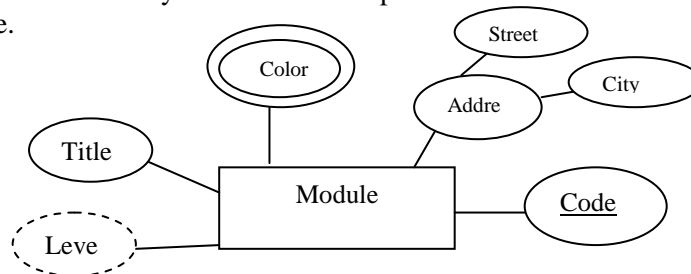

Figure 2. The Entity Notation

3.2 Attributes in ER diagrams vs. UML Class Diagrams:

Attributes in UML is similar to attributes in the ERD. The attributes in the class that describes the information class is shown as plain text in class rectangle (figure 1). The attributes of ERD appear linked to them through lines and circular forms. We can show types of attributes, such as compound, simple, derivative and multi-value denote as double cycle. (Figure 2).

3.3 Operations in ER diagram vs. UML Class Diagrams:

The operation is the behavior of the class. They express the specific work carried out by class .The work is shared among each object in that class. There is no definition of operations in ER model.

3.4 Cardinalities and Participation in ER Diagrams vs. UML Class Diagrams:

Both modeling techniques have the same level for showing cardinalities and participation between two entity/classes, but there are some differences, appearing in the example of the employee and the department.
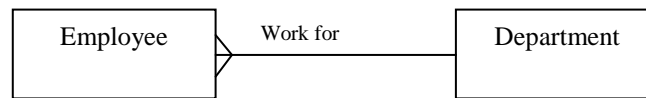
Figure 3. Cardinality and Participation Notation in UML Class Diagram

A cardinality constraint shows the maximum number of objects that can be connected between each other. The notation of cardinality (figure 3) using in UML class diagram means , more than one employee work for each object in department .The relation between employee and department becomes mandatory.
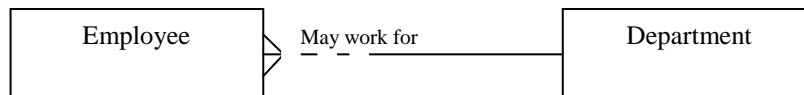

Figure 4. Cardinality and Participation Notation in UML Class Diagram

Figure 4 shows not all employees have work for an object of department class .The relation between employee class and department class becomes optional. Some UML notations use different ways to denote cardinality and participation.
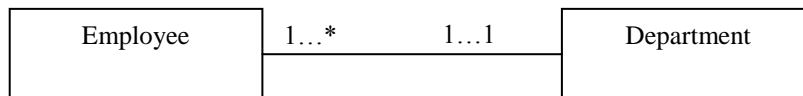

Figure 5: UML Example Mandatory Relationship for an Employee To Work for a Department.
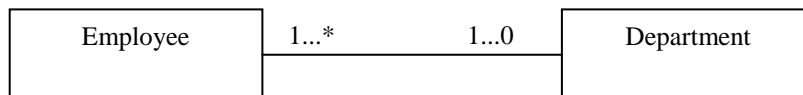

Figure 6: UML Example Optional Relationship for an Employee to Work for a Department

In figure 5 and 6 minimum number is the first number. If the first (minimum) number is one (1) the denote mandatory participation .if zero (0) this denote partial participations. The second (maximum) number is the number of object can take participations. If it is many can use *.
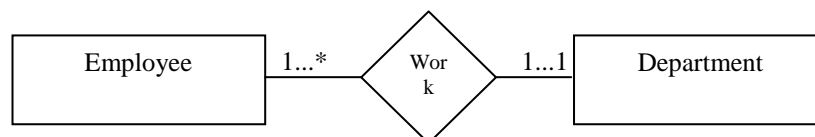

Figure 7. ER Diagram in Elmasry and Navathe Notations

In figure 7 shows Elmasry and Navathe notations use (1..1) , (1..*) ,(M..1) and (1...N) to represent participations and cardinality in binary relation. M or N or *means many object .if first number is one (1) means use one object at least (mandatory relation). If first number is zero (0) means partial relation.

In Chan's original ER notations using double line to represent total cardinalities and using single line to represent partial cardinalities in ER diagram.
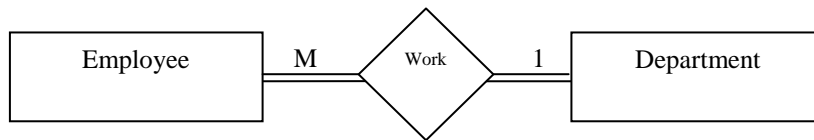
Figure 8: Chan's Original ER Notations

In figure 8, represent in each department   many employees work, and   all employees must work in only one department .The   participation of employee in department   is mandatory.

3.5 Association vs. Relationship in:

Association in UML class diagram is the relationship between the two Classes to determine how each of them affects other. The relationship between two Classes or more. In figure 3 and 4, the association represent in UML through a simple line. The   relation name placed on behind line.

The EER model is represented relations through the Diamond-shaped, where write inside the Diamond-shaped. It appears in the figure of 8 and 7.

4. Convert EER Representation to an Equivalent UML Representation

All EER parts find its equivalent in UML class diagram. Accepted the operation and weak entity.

The EER model focuses on analysis phase and conceptual level of database. The operation is not defined in implementation phase, so operation does not exit in EER model. The weak entity may solve this problem as much way:

4.1 Weak Entity

A weak entity is described as when entity depended on the other entity (parent) in the relationship, primary key of the weak entity is combination between primary key from the parent entity and a partial key of the weak entity.

Song et al [1] compare between ten entity relation modeling notations .Discuss of The weak entity in five notations fully, and three notations partially.
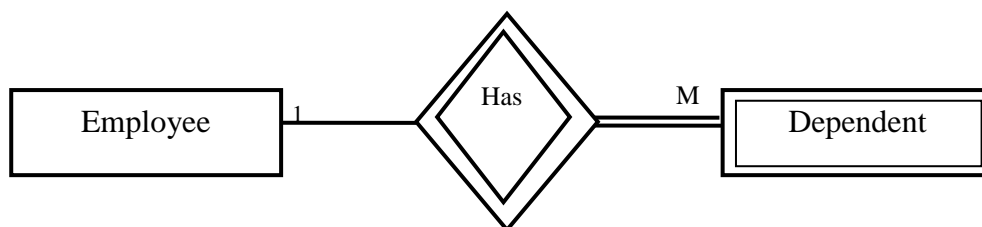

Figure 9. ERD of Dependant Weak Entity – Elmasiry & Navathe Notation

Figure 9 shown an example of weak entity .dependent weak entity cannot be determined without having an employee record .a weak entity represent as rectangle inside rectangle. A weak entity does not direct equivalent represent in UML class diagram. To solve this we can

4.1.1 First Possibility: In figure 10 if employee is deleted, we do not need to keep the depend
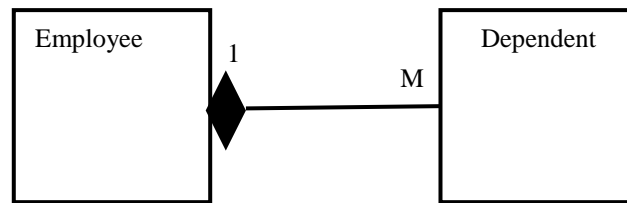
Figure 10: Possible Notation for Weak Entity

4.1.2 Second possibility: Ignoring the weak entity from UML class diagram, and using regular entity to represent association and cardinalities between them. Shown that in figure 11
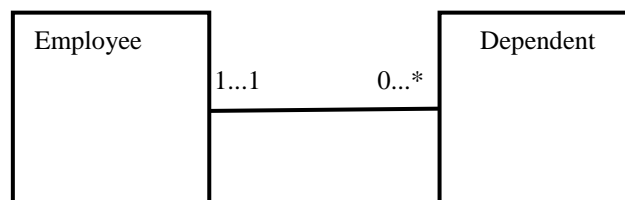

Figure 11. Ignore the Weak Entity and Use Regular Entity with 1:m Relationship

5. The Experiment

An experiment to discern the opinion of developers and users of databases in both models, and determine the extent of knowledge of the differences between the two models. The experiment shows the model fits over the analysis phase, and gives more details about database systems. Any model used in analysis, should be a reference for all stages to develop the database system.

5.1 Experiment Aims

The aims of providing a summary of the models are used to analyze the database. The way the experiment is through involving groups of IT students.

The first aim of this experiment is to determine which famous notations between designers are used to analyze database system. This aim enhances the possibilities that more than one notation can be used in the analysis stage. All of these are good notations to analyze the database systems with some differences. This aim checks through the opinion of involving groups of IT students.

The second aim is trying to find UML notation is equivalent the EER notation.

5.2 Experiment Method

The experiment contains a collection of database examples. Each example includes some different characteristics. The study designs all diagrams on tutorial sheet using UML notation and EER notation. There are some important characteristics on the experiment:

- All examples in tutorial sheet have a binary relation.
- The tutorial sheet contains a simplified explanation of the notations, but does not explain basic concept of database. Participating students must have the basic concepts of databases. Students must be able to identify cardinality of relationship and participation constraint and other basic concepts.

- Participating groups are given their opinion on notations submitted to the databases examples on the tutorial sheet, and in some examples required drawn model of UML or EERD model of business rules.

5.3 Experiment Materials

5.3.1 The experiment paper: The experiment was performed by taking a sample of 15 students. At the beginning of the study the goal was explained to them. The goal was to determine what is the preferred model analysis of database systems and customary among designers and users. The majority of the answers before distributing paper revolve around two models: UML and EERD.

The distribution of experiment paper will be the next step, where the experiment paper contains five systems. Each system contains the different characteristics from the other. Each system has a range of questions concerning the clarity of relationships, attributes, participation and cardinality constraint

5.3.2 The experiment diagrams: The systems that entered in the experiment are the employee system, car ownership system, the university system, the bank system and training center system. I mention here to training center system:

• Training center system:

Any course has a number, name, timing and the type of course. The course can take several sessions during the year. Each session has supervisor. The session has a number, cost, and start date. Any supervisor can supervise more than one session. The supervisor has number, name and address. Each session is followed by a number of participants. Each participant has the following information recorded the name, number and address.

- **Main Entity/class on training center system**

  Course:
   Takes in several sessions during the year.
  Session:
   Must be supervised by supervisor.
   The number of participants more than one.
   More than one session per course during the year.
  Supervisor:
   Supervised one session or more.
  Participation:
   Allows each participant to take only one session in one time.

Figure 12. Explain for Training Center System

A figure 12 represented the business rule for figure 13 and 14.The figure 14 represent ER diagram and figure13 represent UML diagram for training center system .Some of the questions asked of the participants on figure 13 and figure14 :
   1. What are the UML and EER?
   2. Why do we need to analyze the system when the systems development?

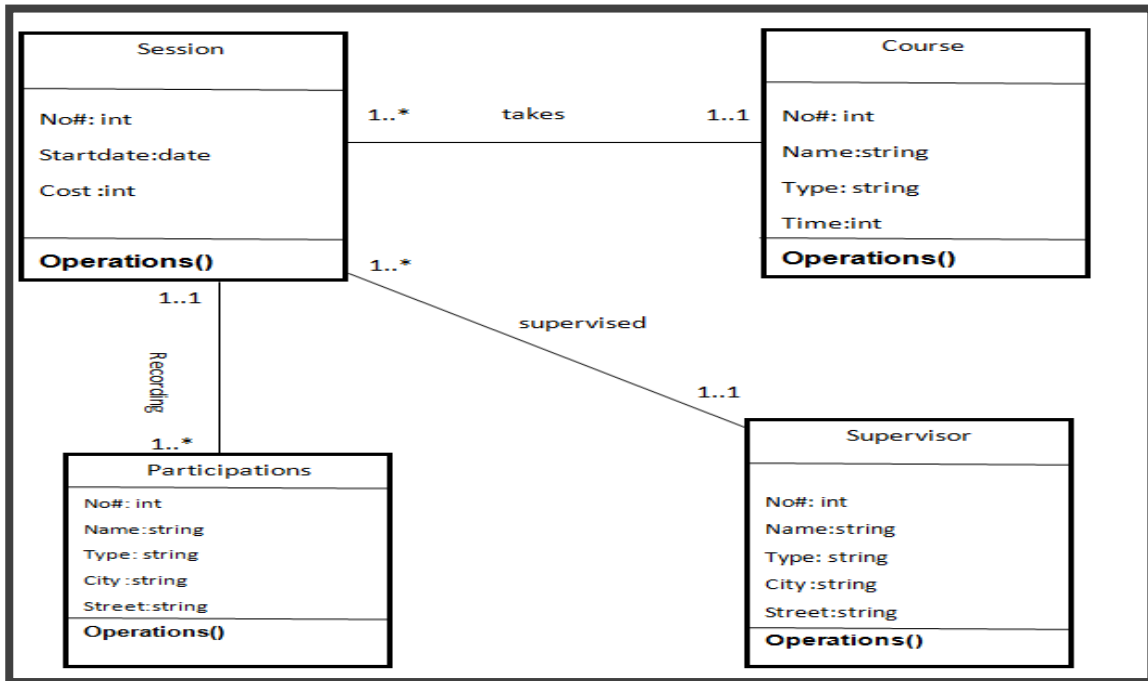3. Is it clear relation in the figure 12 and figure13?

Figure 13. UML Diagram for Training Center

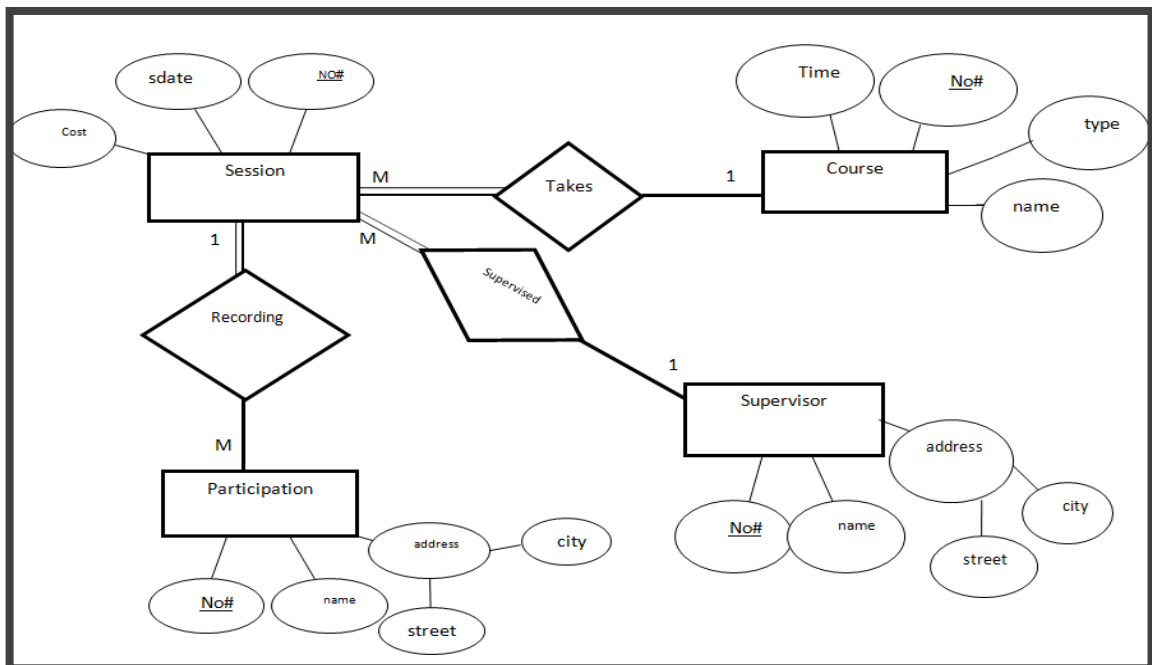4. Which model is preferred to be used in system analysis? And why?

Figure 14. UML Diagram for Training Center

6. Analysis

The experiment conducted on students showed convergence in the responses. The result is that EERD is the best and was rated 80 percent. The justification for this rate is the EERD has the ability to represent all elements of the relational database system that go into the analysis phase. Without going into details, we do not need it in the analysis phase, but it can be mentioned in the other stages of database system development.

The EERD is directly known in entity dependent on other entity in existence .This entity called a weak entity. It is not known operations involved in the implementation phase. While the UML class diagram is no direct definition of weak entity, and knows the operations that should be known in the implementation phase.

On the other hand, the experience has shown the proportion of students who are able to understand the systems analyst through ERD is 70 percent, and the percentage of students who are able to understand the systems represented in the analysis phase through UML class diagram was 20 percent. Table1and chart 1 show the result of experiment**.**

Table 1. Result of Experiment

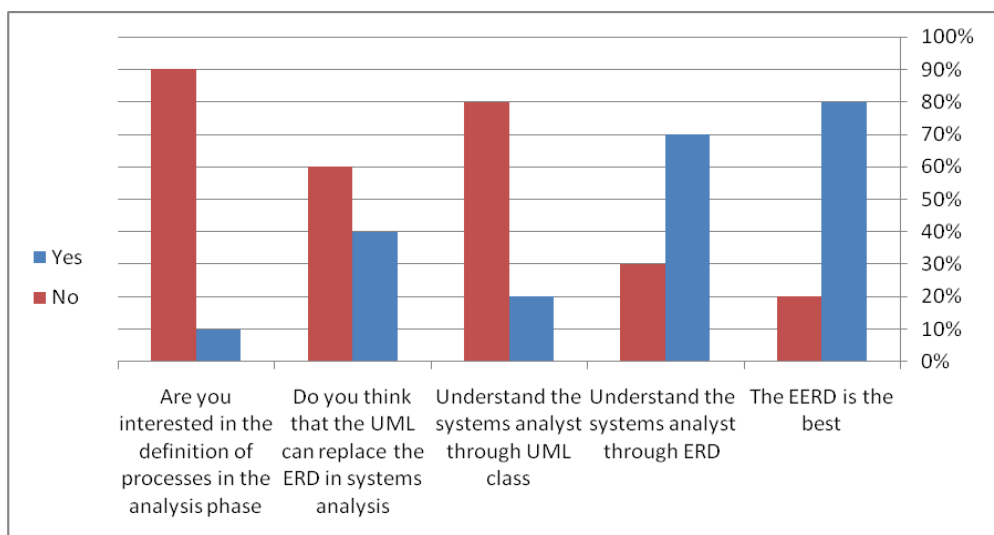| Question	Yes	No
The EERD is the best	80 %	20 %
Understand the systems analyst through ERD	70 %	30 %
Understand the systems analyst through UML class	20 %	80 %
Do you think that the UML can replace the ERD in systems analysis	40 %	60 %
Are you interested in the definition of processes in the analysis phase	10%	90%

Chart 1. Result of Experiment

7. Limitations

The constraints that were in the study are the number of students participating in the experiment. We need many of participants in order to get more accurate results.

8. Conclusion and Future Work

There are many differences between using UML class diagram and EER diagram during analysis and implementation phase .The UML Class Diagram fits the implementation phase more than the analysis phase, because UML Class diagram contains a place that defines behaviors. The details can be defined on the database behaviors such as deletion and insertion and modification on entities. These details are included in the implementation phase. The EERD represents conceptual level to database design; it is not entered in implementation specific detail. Indeed, the UML class diagram mainly designed for Object Oriented Programming (OOP), and object oriented concepts is different from database concepts. In OOP it has properties and methods, can be define as public and private. UML Class Diagram is more programming tool than database administrator tool.

The weak entity type has difficult representation on UML Class Diagram .There are ways to solve this problem, but any notations used must be agreed upon, and is known for both the designer and the user. The solutions of weak entity are not agreed between everyone.

It was concluded through this study experience. The students' choice for the best notation depends on how the student understands the system, and the student's knowledge of notation. Students, who participated in the experiment, are more familiar with EER.

As a results , we found that teaching Database designing course would be helpful for the student if it was based on EER model .The student must be taught UML but in implementation phase (logical design) .In the future we will compare between UML and EERD , especially when relations are of the type   many- to- many .and we will use online system and increase number of students participating to perform the experiment .

Acknowledgements

References

[1]   I. Y. Song, M. Evans & E. K. Park, "A comparative analysis of entity-relationship diagrams", Journal of Computer and Software Engineering, vol. 3, no. 4, **(1995)**, pp. 427-459.

[2]   H. C. Purchase, R. Welland, M. McGill and L. Colpoys, "Comprehension of diagram syntax: an empirical study of entity relationship notations", International Journal of Human-Computer Studies, vol. 61, no. 2, **(2004)**, pp. 187-203.

[3]   P. S. Dhabi, M. S. Patwardhan, A. A. Deshpande, M. L. Dhore, B. V. Barbadekar and H. K. Abhyankar, Articulated entity Relationship (AER) Diagram for Complete Automation of Relational Database Normalizaation, **(2010)**.

[4]   D. Yeh, Y. Li, and W. Chu, "Extracting entity-relationship diagram from a table-based legacy database", Journal of Systems and Software, vol. 81, no. 5, **(2008)**, pp. 764-771.

[5]   E. Song, S. Yin & I. Ray, "Using UML to model relational database operations", Computer Standards & Interfaces, vol. 29, no. 3, **(2007)**, pp. 343-354.

[6]   D. Shah and S. Slaughter, "Data modeling and UML. Unified Modeling Language: Systems Analysis", Design and Development Issues: Systems Analysis, Design and Development Issues, vol. 43, **(2000)**.

[7]   C. Baral, G. Gonzalez and T. Son, "Conceptual modeling and querying in multimedia databases", Multimedia Tools and Applications, vol. 7, no. 1-2, **(1998)**, pp. 37-66.

[8]   A. Mammar and R. Laleau, "A formal approach based on UML and B for the specification and development of database applications", Automated Software Engineering, vol. 13, no. 4, **(2006)**, pp. 497-528.

[9] B. M. Byrne, & Y. S. Qureshi, "The Use of UML Class Diagrams to Teach Database Modelling and Database Design", University of Sunderland, vol. 11, **(2013)**.

[10] P. P. S. Chen, "The entity-relationship model—toward a unified view of data", ACM Transactions on Database Systems (TODS), vol. 1, no. 1, **(1976)**, pp. 9-36.

[11] P. Weaver, N. Lambrou and M. Walkley, "Practical business systems development using SSADM: a complete tutorial guide", United Kingdom, Harlow: Pearson Education, **(2002)**.

[12] S. S. S. Cherfi, J. Akoka and I. Comyn-Wattiau, **(2003)**, "Conceptual modeling quality-from EER to UML schemas evaluation", Conceptual Modeling, **(2002)**.

[13] S. Xu, Y. Li, and S. Lu, "ERDraw: An XML-based ER-diagram Drawing and Translation Tool", Computers and Their Applications, **(2003)**, pp.143-146.

[14] F. Batmaz and C. J. Hinde, "A diagram drawing tool for semi–automatic assessment of conceptual database diagrams", **(2006)**.

[15] N. Mfourga, "Extracting entity-relationship schemas from relational databases: a form-driven approach", Reverse Engineering, Proceedings of the Fourth Working Conference, **(1997)**.

[16] V. Cosentino and S. Martinez, "Extracting UML/OCL Integrity Constraints and Derived Types from Relational Databases", 13th International Workshop on OCL, Model Constraint and Query Languages, **(2013)**.

Author

Manal Mahmoud Al Khoshman, she is a postgraduate student at Mutah University finishing her Master's degree. She works at Al Blaqa' University as assistant lecturer, and a researcher in the field of database analysis. She Lives of Al Shoubak - Ma'an, Jordan.