Conceptual Data Modeling in the Introductory Database Course: Is it Time for UML?

James Suleiman
Information Systems Department
University of Southern Maine
Portland, ME 04104
suleiman@usm.maine.edu

Monica J. Garfield
Computer Information Systems Department
Bentley College
Waltham, MA 02452
mgarfield@bentley.edu

ABSTRACT

Traditionally, the typical undergraduate database course uses a form of Entity-Relationship (ER) notation when teaching conceptual modeling. While we have seen an increase in the academic coverage of UML in the database course, it is very rare to see UML as the primary modeling notation when teaching conceptual data modeling. However, outside of academe, there has been advocacy for the use of UML as an effective modeling tool for database design and for it to provide a unifying modeling framework. This paper examines the level of support for using UML vs. established ER notations for teaching conceptual data modeling in the introductory undergraduate database course. An analysis of textbook and tool support as well as a survey of what IS undergraduate programs are using in their introductory undergraduate database courses is included.

Keywords: Conceptual Data Modeling, Entity-Relationship, UML, Database Education

1. INTRODUCTION

As data modeling has evolved in the last 50 years we have seen a shift from hierarchical and network models to relational and object-oriented models. While the term "data modeling" may imply a variety of different meanings (Topi, et al., 2002), in information systems (IS) education, data modeling is consistently used to describe entities and relationships within a real world domain (Hoffer, et al., 2005). In the past 20 years relational data models have dominated the market but today the Unified Modeling Language (UML) has emerged as the software industry's dominant modeling technique for application development (Siau, et al., 2001). In the past few years there has been an increase in interest in the applicability of UML class diagrams in data modeling.

While there are a wide range of issues one must consider when selecting an appropriate data modeling language, the aim of this paper is not to pass judgment on or comment on which modeling technique is correct. It is to gain insight into the support for the different modeling techniques and the current state of data modeling in undergraduate database courses. As the paradigm governing modeling techniques evolves there comes a time when the academic environment may consider if the tipping point has been reached where the academic teachings in introductory database courses have the support to shift to UML. While we recognize that there are many theoretical and practical issues to consider when selecting an appropriate data modeling technique used in the classroom we have chosen to report on the current level of support for the use of ER modeling and UML class diagrams in undergraduate database courses.

This paper examines the viability of UML as a conceptual modeling notation for an introductory undergraduate database course by investigating the supporting issues, including: curricular fit; support materials (i.e., books and tools); and the use of UML in IS undergraduate programs. We then discuss the strengths and shortcomings of UML for teaching conceptual data modeling in light of these supporting issues. Finally, we highlight potential directions for future research and discuss conclusions and limitations of this study.

2. SUPPORTING ISSUES

In order to gain insight into the viability of using UML as a notation for teaching conceptual data modeling we examine some of the infrastructural supports for teaching UML in an introductory database course. First we discuss influence of the overall IS curriculum, specifically whether or not an object-oriented methodology is reinforced throughout the curriculum, on the readiness for teaching UML in database. We then examine the support of UML in eleven introductory database texts marketed to the academic community along with five popular software applications that support the diagramming of conceptual data models. Finally, we analyze the level of coverage of UML in current introductory database courses at nineteen undergraduate IS business schools in the United States.

2.1 Curricular Fit

When selecting the modeling technique for an introductory database course one must keep in mind the unique characteristics their academic environment provides. The database course is often part of an IS majors' curriculum and finding synergies between the courses in the curriculum may be a priority. While there may be multiple courses one needs to consider fit with, the most prominent course topics to consider are programming courses and systems analysis and design courses. If your program is using an object-oriented (OO) methodology and has embraced UML in the systems analysis, systems design and programming courses your students may have already been exposed to UML, thus potentially making the use of UML class diagrams a more natural fit and one that is more synergistic within your overall undergraduate IS curriculum. By giving the student multiple exposures to UML in different contexts the student will be able to get a more holistic view of systems design and integration.

Furthermore, the sequencing of the introductory database course needs to be considered. The earlier the database course falls in the sequence of required courses for the majors the easier it is for the database instructor to select the diagramming technique they are most comfortable with. However, if students have already been exposed to UML class diagrams, the database instructor may need to keep that in mind to reduce the confusion of the use of the various modeling techniques. The database course does not have to use UML if the systems analysis and design courses do but the database teacher may find that they need to address the differences in the modeling techniques in order to improve student comprehension of the integration between the various courses.

2.2 Support Materials

One way of assessing the level of support for teaching UML is to examine the availability of textbooks facilitating teaching of the language and tools supporting the notation used in the textbook. A total of eleven texts were examined for this study. The sample was restricted to texts geared toward the higher education market (i.e., academic texts) since they are most likely to be adopted in colleges and universities across the world. In order for a textbook to be considered, it needed to cover conceptual data modeling in some form. Many textbooks used modified formats of modeling notation, however, if the primary notation used in the text exhibited one of the signatures listed in table 1, and did not contain any other signatures from table 1, it was classified according to that signature. Classification by a simple signature is not without limitations. Whether or not the entire symbol set is supported and if the methodology of the language is adhered to was not considered. This should not be considered a serious limitation since the purpose of this classification was to analyze the modeling language that the notation in the text was derived from.

Of the texts analyzed, only two (Gillenson, 2005; Watson, 2004) did not fall into any of the classifications and were labeled "hybrid." Gillenson's notation used Chen's relationship diamonds to display the relationship name and Information Engineering's (IE's) crow's foot notation to represent cardinality and optionality. The author acknowledges that he "took the best ideas" from several modeling notations and added some of his own

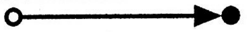
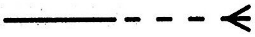
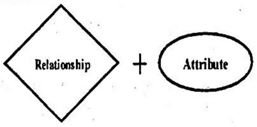
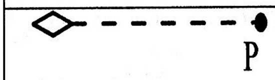
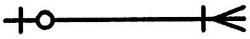
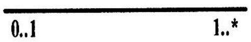
Signature	Classification	Description
	Bachman	Triangle represents cardinality, circle represents optionality
	Barker	Crow's foot represents cardinality, line style (i.e., solid or dashed) represents optionality
	Chen	Diamond shape relationship identifier and outside of the box oval attribute representation
	IDEF1X	Circle or diamond represent cardinality (i.e., many, one), letters (i.e., P, Z) represent optionality
	Information Engineering (IE)	Crow's foot and line represent cardinality, line and circle represent optionality
	UML	Cardinality and optionality are shown in numbers.

Table 2 - Classification of Notation

Citation	Text	Edition	Prevalent notation	Number of chapters covering data modeling	Total Chapters
(Connolly, et al., 2005)	Database Systems	4	UML	6	34
(Elmasri, et al., 2004)	Fundamentals of Database Systems	4	Chen E-R based	3	28
(Gillenson, 2005)	Fundamentals of Database Management	1	Hybrid (Chen and IE like)	2	15
(Hoffer, Prescott and McFadden, 2005)	Modern Database Management	7	Chen	4	15
(Kroenke, 2004)	Database Processing: Fundamentals, Design, and Implementation	10	IE Derived	1	15
(Mannino, 2004)	Database Design, Application Development & Administration	2	IE Derived	5	18
(Post, 2005)	Database Management Systems	3	UML	1	10
(Pratt, et al., 2002)	Concepts of Database Management	5	Access	1	9
(Rob, et al., 2004)	Database Systems: Design, Implementation and Management	6	Chen & IE	1	15
(Silberschatz, et al., 2002)	Database Systems Concepts	4	Chen E-R	2	27
(Watson, 2004)	Database Management: Databases and Organizations	4	Hybrid (LDS Like)	6	20

Table 3 - Summary of Database Textbooks

(Gillenson, 2005). Watson uses a notation that appears to be derived from Logical Data Structures (LDS) notation (Carlis, et al., 2000) but employs some different symbols.

For the introductory database text, there is no apparent standard conceptual modeling notation. Of the 11 texts analyzed, 4 used a Chen variant as the predominant notation, 3 used an IE type notation, 2 used UML and 2 used a hybrid notation. All of the texts included a chapter or appendix on object oriented databases and UML notation. In the practitioner market there are several books that use UML for conceptual data modeling with one of the more prevalent being UML For Database Design (Naiburg, et al., 2001). For a summary of textbooks analyzed, see table 2 above.

Five popular packages that support the diagramming of conceptual data models were also examined. We found the least amount of support for Barker and Bachman notations and a high level of support for UML and IE (Chen and IDEF1X have moderate support) Results are summarized in table 3 below.

2.3 Current Use of UML in IS Undergraduate Programs

Finally, in an attempt to understand the level of support for UML teaching in undergraduate IS database courses we contacted schools that were identified as the top IS undergraduate business schools according to the U.S. News and World Report 2004 Undergraduate Rankings of Business Schools in MIS("America's Best Colleges," 2004). First we looked at the course descriptions of the database courses in the top listed schools and followed up by contacting the faculty that teach those courses.

Notation type	Visio 2003	Visible Analyst 7.5	ERWin Data Modeler 4	Oracle Designer 10G	SmartDraw 7
Bachman	no	yes	no	no	basic shapes (no templates)
Barker	no	no	no	yes	no
Chen	via templates	yes	no	no	yes
IDEF1X	yes	yes	yes	no	no
IE	yes	yes	yes	no	basic shapes (no templates)
UML	yes	yes	via Component Modeler	via JDeveloper	yes

Table 4 - Tool Support

Program	ER	UML
Massachusetts Inst. of Technology (Sloan)		
Carnegie Mellon University (PA)		x
University of Texas - Austin (McCombs)	x	
University of Arizona (Eller)		x
Univ. of Minnesota - Twin Cities (Carlson)	x	
Univ. of Maryland - College Park (Smith)	x	
University of Michigan-Ann Arbor	x	
University of Pennsylvania (Wharton)	x	
New York University (Stern)	x	
Georgia State University (Robinson)	x	
University of California-Berkeley (Haas)	x	
Indiana University- Bloomington (Kelley)	x	
Bentley College (MA)	x	
Purdue Univ. - West Lafayette (Krannert)	x	
Arizona State University (Carey)	x	
University of Georgia (Terry)	x	
University of Oklahoma (Price)	x	
University of Virginia (McIntire)	x	
University of Washington		x

Table 5 - Data Modeling Techniques in IS Undergraduate MIS Programs

As evident from table 4 very few schools have chosen to use UML as their primary modeling technique in their undergraduate introductory database course. However, many schools noted that they did introduce UML in one class when they cover OO databases. While many faculty members noted that UML is being used in their Systems Analysis and Design courses, a few remarked that they strongly believed it was inappropriate to use UML in the database course. There was no database management course offered in Sloan's online catalogue and our contact verified that such a course did not exist.

3. DISCUSSION

The support for UML modeling at the undergraduate database level does not appear to be very strong. While there appears to be synergistic reasons to consider the use of UML in database courses in some curriculum, the majority of schools have chosen not to use UML as the primary modeling technique in the undergraduate introductory database course. Academic textbooks do appear to be reflecting the market's movement towards UML and are placing the discussion of the technique primarily in appendices or in an OO database chapter. However, the primary modeling techniques emphasized in the majority of books we examine (9 of the 11) used ER related notation. Finally, the tools that support database diagramming all seem to have the capability to create UML diagrams. However, many of these tools are used to for a wide range of diagrams and therefore, the UML notation may be included for use in different components of the systems analysis and design phase of system building.

From a readability standpoint, UML class diagrams exhibit two major weaknesses. The UML notation for relationship cardinality (i.e., using numbers and asterisks) is less intuitive than ER notation and the representation of subtype boxes outside, rather than inside, supertype boxes can make it more difficult to establish what relationships a class is involved in, especially in more complex hierarchies (Simsion, et al., 2005). Both of these shortcomings are illustrated below in figure 1. The combination of the dominance of ER models, the primary focus on relational databases in introductory database courses and the complexity of UML class diagrams are compelling reasons to consider using ER notation in the class instead of UML.

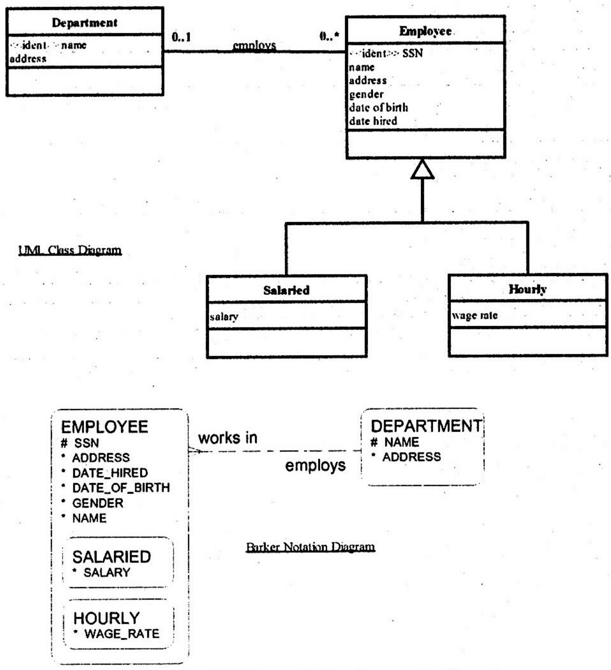


Figure 2 - UML and Barker notations

While the example in figure 1 may help support the argument that Barker notation of ER diagramming may be easier to read than UML, the same argument can be made against many other traditional ER notations. While widely used in higher education, Chen's notation is not as prevalent in practice simply because it puts too many objects on the page (Simsion and Witt, 2005). An examination of figure 2 shows an equivalent data model of an employee/department relationship using an extended form of Chen's notation to represent the subtyped entities (Salaried and Hourly) alongside an identical model using Barker's notation. One can see why notations representing attributes inside of the entity box, resulting in less space, have been preferred. Barker's notation was chosen to make this contrast as prior literature has identified it as an example of a formal graphical notation that is easily readable and well suited to data modeling (Hitchman, 2002).

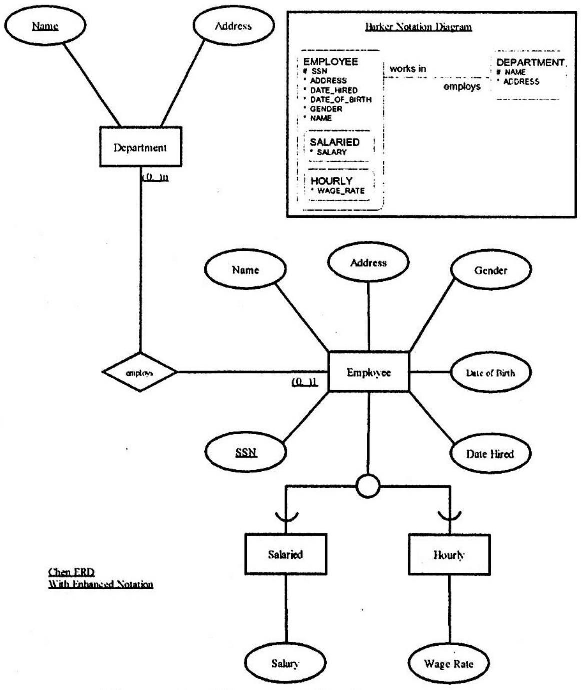


Figure 3 - Chen and Barker notations

While UML notation may be slightly harder to read than the Barker notation, the differences are negligible and present in other notations that are commonly used (e.g., Chen, IE, etc.). Therefore, it is hard to argue that readability is the main reason for the lack of use of UML in the classroom. It appears to be fairly clear to us that the lack of support for UML in the classroom may be due to: incompatibility between the goals of database modeling and the UML modeling technique; a lag in academic support for new database modeling techniques; or not enough of a significant difference to warrant a change.

As we look into the future we need to consider how UML relates to the database design effort. Because of advances in object based technology, it has become important to teach techniques for mapping object-oriented designs to relational, object-relational and object-oriented database systems (Urban, et al., 2003). Furthermore, the development of database applications requires close coordination between software developers and the database development team. Software developers increasingly are becoming accustomed to communicating application requirements via UML while the database designers build the database supporting the application. The overlap between these two functions is often the most challenging part of a software project. Using UML for data modeling can help resolve this challenge and in industry, it is becoming more clear that UML is increasingly being used in data modeling formalism (Wagner, 2005).

While industry has been evolving, IS educational curricula have been evolving as well. Introductory programming courses, which are often a prerequisite to an introductory database course, are increasingly moving to object-oriented languages (Neubauer, 2002). Since students may already be familiar with UML notation from their analysis & design or programming courses, teaching an additional notation can become confusing to students. Furthermore, some feel that the central aspects of E-R schemata can be found in UML Class Diagrams (Gogolla, et al., 1991). However, UML may not be the diagramming model that is most beneficial to the database design effort. While UML is the de facto standard for object oriented development it is not clear if it is the best fit for diagramming data design needs.

4. LIMITATIONS AND DIRECTIONS FOR FUTURE RESEARCH

This paper examines the support and use of UML as a conceptual modeling notation for introductory undergraduate database courses. There is no definitive answer as to what is the "best" notation to use for data modeling in undergraduate database courses, this study is intended to start a discussion that has been ongoing in industry. Future research and more in depth discussions are necessary to gain additional insight into the questions of what type of modeling to use in an undergraduate database course. This study is not without its limitations, which include: sample selection for schools and the use of books to represent the level of support.

The U.S. News and World Report ranking is not a representative sample of introductory database courses. It also only considers management information systems programs in business schools. To get a better picture of the use of UML in database courses one would need to increase the breadth of school coverage as well as the depth (e.g., by including computer science and other technology disciplines).

Furthermore, we choose to look at textbooks and diagramming tools as a proxy for support for UML in teaching database courses. While textbooks support classroom efforts, they do not dictate what occurs in any particular classroom. The text book market may well reflect an increasing demand for knowledge about using UML for data modeling, however, this does not necessarily imply that the chapters that cover UML or ER are being used in the teaching of the database classes that use the specific books. Furthermore, the use of current texts does not consider planned revisions to future editions of texts and what is prepress. Many of the textbooks included in this paper have a chapter or appendix that discusses UML notation, it is quite possible that that chapter alone will become the dominant modeling chapter within a book. Future research should also include the volume of sales of these books which may give us a better idea of how many students are using each of the books.

5. CONCLUSIONS

For the introductory database course, there appears to be more support for ER modeling (in terms of textbook availability) than for UML modeling and most of the IS programs examined use traditional ER modeling techniques as their primarily modeling technique in their undergraduate introductory database courses, While many introduce UML in a few class periods, their primary modeling technique is an ER variant. If there were a notation that was clearly superior for teaching conceptual data modeling, the academic textbook market should reflect this – which it does not.

In industry, there are vocal advocates for using UML as a replacement to ER notation for conceptual data modeling. The prevailing argument is that since UML is the de facto standard for application development and databases are typically one component of an application, UML provides a unifying framework or holistic view of the application environment. One of the reasons why there is no apparent parallel advocacy in academe is that this argument is controversial and has not been resolved.

It is expected that object-relational and object-oriented database technology will become more prevalent. Practitioner-oriented learning materials have already addressed the need to provide support for data modeling with UML and it is expected that academic support materials will address this market need as well. While UML enjoys wide tool support, the current academic textbook market does not strongly support the use of UML in database text books and very few of the IS programs examined have embraced it in their database courses.

Does a potential change from ER notation to UML notation for conceptual data modeling warrant discussion? For those who are considering the use of UML for conceptual data modeling this article should help them understand the current status of the use of UML in database courses and provide a starting point in a conversation that has already been taking place in industry. Before we can fully address the benefits and problems with using UML for conceptual data modeling, we need to better understand what is being used in industry and what is being taught in the classroom. This paper makes the point that this argument warrants further discussion. It also shows that if one were to replace ER with UML notation in an introductory course, it would require a pioneering effort (i.e., there is minimal academic support for doing so).

6. ACKNOWLEDGEMENTS

We would like to thank the faculty from participating institutions that responded to our queries regarding use of UML in conceptual data modeling courses.

7. REFERENCES

"America's Best Colleges." *U.S. News and World Report* 2004, 82.

Carlis, John and Joseph Macguire (2000) Mastering Data Modeling: A User Driven Approach, 1st Edition, Addison-Wesley Longman, Boston.

Connolly, Thomas and Carolyn Begg (2005) Database Systems: A Practical Approach to Design, Implementation, and Management, 4th Edition, Pearson Education, Essex.

Elmasri, Ramez and Shamkant B. Navathe (2004) Fundamentals of Database Systems, 4th Edition, Pearson Addison Wesley.

Gillenson, Mark L. (2005) Fundamentals of Database Management Systems, 1st Edition, John Wiley & Sons, New York.

Gogolla, Martin and Uwe Hohenstein (1991), "Towards a Semantic View of an Extended Entity-Relationship Model." ACM Transactions on Database Systems, Vol. 16, No. 3, pp. 369-416.

Hitchman, Steve (2002), "The Details of Conceptual Modeling Notations Are Important - A Comparison of Relationship Normative Language." Communications of the Association for Information Systems, Vol. 9, pp. 167-179.

Hoffer, Jeffrey A., Mary Prescott and Fred McFadden (2005) Modern Database Management, 7th Edition, Pearson Prentice Hall, Upper Saddle River, NJ.

Kroenke, David (2004) Database Processing: Fundamentals, Design, and Implementation, 9th Edition, Prentice Hall.

Mannino, Michael V. (2004) Database Design, Application Development & Administration, 2nd Edition, McGraw-Hill/Irwin, New York.

Naiburg, Eric J. and Robert A. Haksimchuk (2001) UML for Database Design, Addison Wesley.

Neubauer, Bruce J. (2002), "Data Modeling in the Undergraduate Database Course: Adding UML and XML Modeling to the Traditional Course Content." Journal of Computing in Small Colleges, Vol. 17, No. 5, pp. 147-153.

Post, Gerald V. (2005) Database Management Systems: Designing & Building Business Applications, 3rd Edition, Mc-Graw Hill, New York.

Pratt, Philip J. and Joseph J. Adamski (2002) Concepts of Database Management, 4th Edition, Thomson Course Technology.

Rob, Peter and Carlos Coronel (2004) Database Systems: Design, Implementation, and Management, 6th Edition, Thomson Course Technology.

Siau, Keng and Qing Cao (2001), "Unified Modeling Language (UML) - a Complexity Analysis." Journal of Database Management, Vol. 12, No. 1, pp. 26-34.

Silberschatz, Abraham, Henry F. Korth and S. Sudarshan (2002) Database Systems Concepts, 4th Edition, McGraw-Hill, New York.

Simsion, Graeme C. and Graham C. Witt (2005) Data Modeling Essentials, 3rd Edition, Morgan Kaufmann Publishers.

Topi, Heikki and V. Ramesh (2002), "Human Factors Research on Data Modeling: A Review of Prior Research, an Extended Framework and Future Research Directions." Journal of Database Management, Vol. 13, No. 2, pp. 3-19.

Urban, Susan D. and Suzanne W. Dietrich (2003), "Using UML Class Diagrams for a Comparative Analysis of Relational, Object-Oriented, and Object-Relational Database Mappings." ACM SIGCSE Bulletin, Vol. 35, No. 1, pp. 21-25.

Wagner, Paul. "Teaching Data Modeling: Process and Patterns." Paper presented at the Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education, Capacrica. Portugal, June 2005.

Watson, Richard T. (2004) Data Management: Databases and Organizations, 4th Edition, John Wiley & Sons. New York.

AUTHOR BIOGRAPHIES

James Suleiman, Ph.D., is an assistant professor of information systems at University of Southern Maine's School of Business and a senior research associate at the Center for Business and Economic Research. He received his B.S. in Finance from Lehigh University, his M.B.A. from University of South Florida and his Ph.D. in MIS from University of Georgia. He was a consultant for Cap Gemini Ernst & Young's division of Telecommunications Media and Networks, worked for IBM and consulted for various Fortune 500 firms. His research interests include information systems education and computer supported cooperative work.

Monica J. Garfield, Ph.D., is an Assistant Professor in Computer Information Systems at Bentley College. Her research focuses on the use of IT to enhance creativity as well as the socio-technical issues that impact telemedicine systems. Her work has appeared in such journals as Information System Research, MIS Quarterly, Communications of the ACM and Journal of Management Information Systems. She is also the editor of ISWorld's Database page (http://www.magal.com/iswn/teaching/database/).

iscap

Information Systems & Computing
Academic Professionals

EDSIG
Serving Information Systems Educators

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 1055-3096