

The Value of Conceptual Modeling in Database Development: An Experimental Investigation

Daniel Turk, Leo R. Vijayarathy, Jon D. Clark
Colorado State University, Fort Collins, Colorado, USA
vijayasa@colostate.edu

Abstract

It is generally accepted that models are useful in systems development and a variety of modeling tools and techniques are taught in information technology academic programs and used by software professionals. However, the recent emergence of rapid systems development approaches including Extreme Programming is challenging the value of modeling. Surprisingly, other than anecdotal evidence, there is little empirical data to support or refute the conventional wisdom that models are useful in systems development. This paper reports the results of an experimental study that examined the relationship between conceptual modeling and database schema development. The results suggest that the use of models helps improve the quality of the schema and reduces the time taken to complete it.

1. Introduction

It is generally accepted that models are useful in systems development and a variety of modeling tools and techniques are taught in information technology academic programs and used by software professionals. However, the recent emergence of “agile” systems development approaches including Extreme Programming and Agile Modeling is challenging the traditionally accepted value of modeling and its use in practice. To resolve the debate on the worth of modeling, empirical studies are absolutely necessary in attempting to understand the value of modeling, to determine when modeling is most useful, and in assessing how much modeling is beneficial. Surprisingly, other than anecdotal evidence, there is little empirical data to support or refute the conventional wisdom that models are useful in systems development, or to address the current claims that agile approaches are advantageous.

This paper reports the results of an experimental study that examined the relationship between conceptual modeling and database schema development. The results suggest that the use of models helps improve the quality of the schema and reduces the time taken to complete it.

2. Literature Review

Kent Beck, one of the earliest proponents of the eXtreme Programming (XP) / Agile Modeling (AM) movement, describes XP as “extreme” in the sense that it “takes commonsense principles and practices to extreme levels” (Beck, 2000, p. xv). Most agile methods (Agile Alliance, 2002; Ambler, 2002; Beck, 2000; Larman, 2001) are “extreme” in some way, and claim to be “lightweight” since they need to be able to adapt readily to changing requirements and resources in the real world. This is a reaction to the “heavyweight” and more formal traditional system development processes that have been followed for years, and to the current pressures to develop systems in “Internet time.” Running useful code is seen to be the deliverable of ultimate interest. Intermediate products (such as models and many forms of documentation), that are not seen to have measurable value for the customer, are only built if they provide some direct help to the developer, and then only so the developer can get on to building code. Models are not built and kept around for their own inherent value, since they are not considered to provide business value to the customer like running code does. Likewise, processes that do not add value to the deliverable that the customer wants are minimized, if not totally discarded.

Most agile methods apply the practice of pair programming (Williams, 2000, 2001) since development time and quality have been seen to improve dramatically with this approach.

The OMG (Object Management Group) is currently pursuing a Model-Driven Architecture (MDA) (MDA 2002) approach that could be construed as an attempt to counter the XP/AM approach. The MDA approach suggests that models are important in their own right, and that development that is built on models may be more reliable and cost-effective in the long run.

Unfortunately there is very little empirical work at present to support or disprove either side’s claims. Most claims are based on anecdotal evidence. This study is a small attempt to begin to add empirical understanding of the value of modeling.

3. Research Questions and Hypotheses

Our research agenda that is focused on examining the value of modeling is guided by the following questions:

1. Are software engineering models useful? (Are agile approaches “better” than non-agile approaches?)
2. What types of models are useful?
3. How are they useful? (What are the benefits? To whom?)
4. When are they useful? When are they not worth it? (What situations, domains? What stage(s) of development? What are the values of building, maintaining, and keeping models for their own purpose? To whom? What are the costs? When do they not provide any “customer” / bottom-line benefit?)

In the current study, we seek answers to some of the above questions by empirically examining the value of conceptual modeling to the development of database schema. The specific hypotheses tested are as follows:

H1: A database schema produced from a correct Entity Relationship Diagram (ERD) will be more correct than one produced without it.

H2: A database schema produced from a correct ERD can be accomplished in less time than without it.

H3: The designer of a database schema will have greater confidence in the design when using an ERD than without it.

4. Method

The method chosen to test the importance of a graphical database modeling tool (i.e., ERD) involved the construction of a problem scenario of modest complexity, one that would be realistic enough in complexity and yet one that could be solved in a time span of about an hour. Three treatment sets were produced based on the hypothetical design problem. All three treatment sets were given a page long written description of the problem. Subjects in treatment set 1 were asked to produce a schema definition based on an ER model that the subjects were required to draw. Subjects in the second treatment set were given the correct ER model and from this were asked to produce the schema definition. Finally, subjects in the third treatment set were given only the written problem description and were asked to directly produce the schema definition, without any intermediate assistance of an ER or other type of model. The correctness and time required to perform each of the tasks was measured.

4.1. Treatment and Dependent Measures

The independent variable in this investigation is the treatment (i.e., no ERD, draw ERD, and given ERD). The dependent variables are time to design the database schema, the correctness of the schema, and confidence in the correctness of the schema.

4.2. Subjects

The study participants were students at a large US public university. The students, who were either Computer Science or Computer Information Systems majors, were recruited from upper-level courses. All of them had completed a minimum of one course in database development and were familiar with ER diagramming. Extra credit was offered as an incentive to encourage participation in the study

4.3. Experimental Procedures

The documents for the three treatment sets were color coded so that they could be administered in a consistent manner. The subjects were randomly assigned to one of the three treatment sets. Each of the subjects was first given a waiver form required by the university regarding rights and responsibilities, impact on course grade and other factors.

After the subjects had read and signed the waiver form, each of them was given the written description of the problem scenario with instructions to read and understand it. Upon completion of this phase of the exercise, subjects in treatment set 1 were asked to draw an ERD for the problem scenario. Subjects in treatment set 2 were given an ERD for the problem scenario and asked to develop the database schema. And, subjects in treatment set 3 were asked to develop the schema from the written problem scenario without the assistance of any graphical models. Subjects in treatment set 1, who had the additional task of drawing of the ERD, were instructed to develop the database schema after they had completed their drawing task.

For each subject, we timed the taken for a) reading and understanding the problem scenario, b) drawing the ERD (only for treatment set 1), and c) developing the database schema. Upon completion of the experimental task, the subjects filled out a short survey that collected data on their degree of confidence in the completed task, age, gender, and number of computer-related courses completed.

4.4. Data Analysis

Our subject pool consisted of 16 female and 34 male students (3 did not indicate their gender). The average age of the participants was 25.6 (median=22) and the

mean number of computer information systems and/or computer science courses completed by them was 5.8 (median=6).

Analysis of Variance (ANOVA) and post-hoc Scheffé tests were conducted to test for differences in performance quality, time taken to complete the schema, and confidence in the correctness of the schema solution. Performance quality was assessed by two research assistants who worked independently to grade each subject's database schema. A grading scheme (Table 1), similar to those used by Batra et al. (1990) and Lee and Choi (1998), was used to ensure objectivity and consistency. We provided adequate training to our assistants by first, jointly grading a schema with them, and then, providing extensive feedback on another schema that was independently graded by them. For our analyses, we used the average of the two graders' scores, which were found to be highly correlated (Pearson correlation: 0.894). Performance time was determined by calculating the difference between each subject's start and end times for completing the schema development task. To measure the last dependent variable, confidence in the correctness of the database schema, four items anchored by 1 (not at all confident) and 7 (very confident) were used. The four items loaded on a single factor and had a reliability coefficient of 0.89.

The results shown in Table 2 indicate that there are significant differences by treatment on measures of overall quality and schema development time. Specifically, the group that was given the ERD outperformed the group that did not use ERD by representing specialization and relationships more correctly in the database schema. Further, the group that was provided with the ERD had a significantly higher score on overall performance quality and took less time than the group that did not use ERD. In contrast, the group that was required to draw an ERD scored better on the facet of specialization and also took less time to complete the database schema than the group that did not use ERD.

Although the means for the confidence measure were higher for the given ERD and drew ERD treatment groups in comparison to the no ERD group, the differences were not statistically significant.

These results show support for Hypotheses 1 and 2, and suggest that conceptual modeling assists database designers develop better schemas in less time.

5. Discussion and Implications

The value of modeling is either taken for granted or discounted as unnecessary without empirical evidence to support either of these two divergent views. Results from our study suggest that modeling does have a positive impact in systems development. Specifically, when the

task is to develop a database schema, performance in terms of correctness of solution and time taken to complete the solution are better when an ER model is provided.

Although our results are based on a small sample of student subjects, it contributes to the growing debate on the value of modeling by offering some empirical evidence. The implications of our findings are that modeling has a role in the systems development process and it may be premature to abandon the teaching of models in academia and/or neglect its application in industry.

6. Limitations

There are several limitations to the current investigation including the use of student subjects, a small sample size, the testing of only one modeling tool, and an experimental task that did not include the implementation of a database schema, but rather only a conceptual representation of it.

7. Directions for Future Research

Empirical evidence from our study suggests that modeling is useful in systems development. In addition to addressing the limitations of our study, future research can examine the value of models in system modification and programming tasks. Further, the impact of moderators on the relationship between the use of modeling and systems development performance can also be studied. Some of these moderators could be task-related such as complexity and requirements completeness and clarity. Others may be based on individual differences such as gender, experience, cognitive capabilities, and approaches to problem solving.

8. References

- Agile Alliance. (2002). <http://www.agilealliance.org>. Visited 2002 Aug 9.
- Ambler, S. (2002). *Agile Modeling: The Official Agile Modeling (AM) Site*. <http://www.agilemodeling.com>. Visited 2002 Aug 9.
- Batra, D., Hoffer, J.A., and Bostrom, R.P. (1990) "Comparing Representations with Relational and EER models," *Communications of the ACM* 33 (2), 126-139.
- Beck, Kent. (2000). *Extreme Programming Explained*. Boston: Addison-Wesley.
- Larman, C. (2001, 2nd ed.). *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and the Unified Process*. Prentice-Hall.
- Lee, H.; & Choi, B.G. (1998) "A Comparative Study of Conceptual Data Modeling Techniques," *Journal of Database Management* 9 (2), pp: 26-35.

Williams, L.; Kessler, R.; Cunningham, W.; & Jeffries, R. (2000). "Strengthening the Case for Pair Programming." *IEEE Software*, 17:4 (July/August), 2000, pp. 19-25.

Williams, L.; & Upchurch, R. (2001). "In Support of Student Pair-Programming." *Proceedings of the ACM Special*

Interest Group on Computer Science Education (SIGCSE 2001) Conference, Charlotte, NC, USA, February, 2001, pp. 327-331.

Table 1. Grading scheme

Facet	Error Classification and Points			Maximum Possible Points
	Incorrect (-1.0 Points)	Medium Error (-0.5 Points)	Minor Error (-0.25 Points)	
Entity	<ul style="list-style-type: none"> Missing Represented as an attribute 	<ul style="list-style-type: none"> Duplicate 	<ul style="list-style-type: none"> Extra entity 	5
Identifier	<ul style="list-style-type: none"> Missing Identifier is different from the one specified in the case 		<ul style="list-style-type: none"> Not underlined or specified with PK notation 	5
Specialization	<ul style="list-style-type: none"> Missing 	<ul style="list-style-type: none"> Incorrect Inheritance Attributes of subtype shown as attributes of super type 	<ul style="list-style-type: none"> Relationship to super type through foreign key not shown 	1
Relationship	<ul style="list-style-type: none"> Missing Incorrect degree 	<ul style="list-style-type: none"> Incorrect connectivity 	<ul style="list-style-type: none"> Unary entity names instead of identifiers for foreign keys 	6

Table 2. ANOVA and Scheffé test results

Facet	Treatment									ANOVA Results	
	No ERD			Drew ERD			Given ERD				
	Mean	SD	N	Mean	SD	N	Mean	SD	N	F	Sig.
Entity	5.00	0.00	18	4.94	0.24	17	4.89	0.32	18	1.02	0.367
Identifier	4.81	0.29	18	4.55	0.42	17	4.82	0.51	18	2.32	0.109
Specialization	0.46	0.46	18	0.80	0.31	17	0.95	0.13	18	10.48	0.000
Relationship	3.08	1.38	18	4.04	1.12	17	4.44	1.67	18	4.46	0.017
Total Score	13.35	1.60	18	14.34	1.58	17	15.10	2.27	18	4.08	0.023
Schema Time ^a	28.00	11.00	18	15.00	5.00	17	21.00	5.00	18	11.52	0.000
Confidence	4.50	1.54	18	4.90	1.43	17	5.18	1.14	18	1.103	0.340

Facet	Statistics	Pair-wise Comparisons		
		No ERD vs.		Drew ERD vs.
		Drew ERD	Given ERD	Given ERD
Entity	Mean Difference	0.06	0.11	0.05
	Sig.	0.758	0.367	0.803
Identifier	Mean Difference	0.26	-0.01	-0.27
	Sig.	0.192	0.999	0.176
Specialization	Mean Difference	-0.34	-0.49	-0.15
	Sig.	0.013	0.000	0.414
Relationship	Mean Difference	-0.97	-1.37	-0.40
	Sig.	0.139	0.020	0.705
Total Score	Mean Difference	-0.99	-1.76	-0.77
	Sig.	0.294	0.023	0.478
Schema Time ^a	Mean Difference	13.0	6.0	-6.0
	Sig.	0.000	.052	0.076
Confidence	Mean Difference	-0.40	-0.68	-0.28
	Sig.	0.699	0.343	0.832

^a: in minutes