AC 2009-1030: STUDENTS' EXPERIENCES FROM IMPLEMENTING A SINGLE PROBLEM IN RELATIONAL, OBJECT-RELATIONAL, AND OBJECT-ORIENTED DATABASE SYSTEMS

Reza Sanati-Mehrizy, Utah Valley State College

Floyd Wilkes, Utah Valley University

Afsaneh Minaie, Utah Valley University

Student Experiences from Implementing a Single Problem in Relational, Object-Relational and Object-Oriented Database System

Abstract

In a previous paper[2], the authors presented an abbreviated example of implementing a common problem in the relational, object-relational and object-oriented database models. The hypothesis was that understanding these latter models would be easier for students if they saw the same database example presented in all these three models. As a test of this premise, one of the authors gave students in a database class the assignment of implementing a database problem in the three models. Students who participated were asked to report their experiences from completing the assignment. The results obtained from the students are evaluated and reported in this paper.

Keywords

Database Design, Relational databases, Object-Relational databases, Object-Oriented databases.

1. Introduction

In a previous paper[1], the authors discussed a perceived difficulty that students have learning the features, capabilities, and uses of the three major database models: relational, object-relational and object-oriented. It was hypothesized that in part, this difficulty comes from the fact that in most database textbooks, these database models are presented in such a way that it is difficult to compare and contrast the features and possible uses of each model with the other two. In another paper[2], the authors presented an example of the implementing a common problem in the three database models. One of the authors then used this concept in a database class, asking the students to select a problem and implement it the relational, object-relational and object-oriented database models. At the conclusion of the class the students were surveyed to gather information regarding their experiences in this process. In this paper, we present the reactions of the students who participated in this process.

2. Teams of Two

As part of a junior level computer science database theory class (CS 3520) students were organize into teams of two. Each team selected a project from a list of projects provided by the instructor (see Figure 1).

```
Team 1: Employee Database
Team 2. Bank Database
Team 3. University Database
Team 4. Patient Database
Team 5. Recipe Database
```

Figure 1. Projects Selected

For the first phase of the project, the members of each team were asked to individually implement their project using a relational DBMS. In this phase of the assignment, each team member was allowed to use any relational DBMS that they were familiar with.

Each team needs to Design, Implement, Present, and submit a database for an organization which the team has been assign to. The process is as follows:

1. Find an appropriate organization for your database to interview.

2. Explain the activity of the organization.

3. Collect information about this organization and analyze it.

4. Design an ER/EER model for this organization

5. Convert your ER/EER model to a set of schemas.

6. Convert your set of schemas to BCNF.

7. Use a relational DBMS to create a DB for your selected organization. Create a table for each schema. Include about 3-5 attribute for each schema.

8. Enter 10-15 tuples in each of your tables.

8. Print every table in a nice tabular form.

10. Run a set of queries on your DB.

11. Print each **query** and its **result** to show your DB works correctly.

12. Submit a few pages of report to indicate what you learned from this project and the problems that you had (if any), hard copy of every thing, plus

Figure 2. Instructions for project Phase 1

In the next phase of the project, each team working as a team was asked to implement the same project two more times, once using an object-relational DBMS and once using an object-oriented database.

Each team needs to design, implement, present, and submit a database for an organization which the team has been assign to using Object-Relational and Object-Oriented Database Management System and then answer the survey questions. The process is as follows:

A. Object Relational Implementation:

1. Use Oracle Express Edition to create the necessary type definitions and tables for your EER/UML model.

2. Use SQL to load the tables the same data that you entered into relational tables in project # 1.

3. Print every table in a nice tabular form.

4. Write SQL queries to retrieve the same information that you did in project #1.

5. Print the English statement indicating the information that you are trying to retrieve.

6. Print each **query** and its **result** to show your DB works correctly.

B. Object-Oriented Implementation:

1. Use DB4o to create the necessary classes for your database.

2. Use eclipse(optional) to create objects corresponding to the tuples that you entered in the Object-Relational model.

3. Print all the object of each extent in certain readable form.

4. Write queries to retrieve the same information that you did in project #1.

5. Print the English statement indicating the information that you are trying to retrieve.

Figure 3. Instructions for project Phase 2

For this part of the assignment, Oracle 10g, Express Edition[3] was used for the object-relational implementation and db4o[4] was used for the object-oriented implementation. The instructions for phase two of the project are shown in Figure 3.

At the conclusion of the class, each student was asked to complete a questionnaire in which they were asked to report on their experiences with the different models (see Figure 4).

Every team member must answer these survey questions.
1. Circle the model that you found to be the most difficult to implement?
   Relational   Object-Relational   Object-Oriented
2. Why did you find it to be the most difficult (be specific)?
3. Circle the model that you found to be the easiest to implement?
   Relational   Object-Relational   Object-Oriented
4. Why did you find it to be the easiest (be specific)?
5. Did implementing this problem in three different models give you a greater understanding of databases in general? (circle one)   Yes   No
      Why?
6. Having completed the projects for this class, do you think in the future you will be more competent to select a database model? (circle one)   Yes   No
   If you answered yes, on what basis would you select a database model for future projects?

Figure 4. The survey questions

3. The Findings

Eight students answered the survey questions. The result is as follows:
Question #1: Relational 1; Object-Relational 5; Object-Oriented 2.

Question #2: Justify your selection.

> The hard part about object-relational was that Oracle Express had limited support for the object-relational features.

> Object-relational was the most difficult because very few resources were available online for object-relational.

> Object-relational was the most difficult because a lot of extra things needed to be done.

> Object-relational was the most difficult because it took the most planning.

> Object-relational was the most difficult because it was so different from the traditional relational.

Question #3: Relational 6; Object-Relational 1; Object-Oriented 1.

Question #4: Justify your selection.

> Relational was the easiest because the student was the most familiar with it.

> The relational DBMS was flexible and the schema was not difficult.

> OO-Database was the easiest one because it seemed a very natural database model.

> Relational was the easiest because there was a lot of support for it and the student had a lot of experience with it.

Relational was the easiest because the student was very familiar with it.

Relational was the easiest because it was the simplest one.

Relational was straight forward and the student had experience with it at work.

Question #5: Yes 8; No 0.

Justify your answer.

It provided the student the opportunity to analyze a study from three different perspectives and learn the strengths and weaknesses of each model.

Although the OODBMS was difficult, the student thought the concept was interesting.

The last two models were new to the student. By doing these projects, he was able to get a better grasp of how to model the same database idea in three different models. The student was able to explore the complexities and simplicities of each model.

After working with the three different models, one can truly appreciate the similarities and differences between them. The student learned how to properly structure and model the data to be stored and varying methods to do so. The models taught how to design a database efficiently.

Question #6: Yes 6, No 2.

4. Discussion

While data from this study are not large enough to be statistically significant, the findings do yield some interesting information. We were not surprised that the majority of students found the relational model to be the easiest to use and the reason given that it was the model with which they had the most familiarity. The reasons given for object-relational being the most difficult were more interesting since they had more to do with the software itself and the supporting documentation.

The reason given by the student who found the object-oriented model to be the easiest also raises a question about how the student population may affect the outcome of the study. The reason this student gave was that it was the most natural. Since the students studied were all computer science students, it raises the question whether the results may be different with a population of information systems students. This will have to be a question for future research.

5. Conclusion

This paper reports the results of a study done comparing the implementation of a single problem in three different database models. The results indicate that students acquire a greater understanding of the different models when using this approach.

6. Refrences

1. Floyd Wilkes and Reza Sanati Mehrizy , "An International Study of the Coverage of Relational, Object-Relational and Object-Oriented Database Concepts in Database Textbooks and a Database Example Represented in These Database Models", American Society for Engineering Education (ASEE) Annual Conference, June 2007.

2. Reza Sanati Mehrizy and Floyd Wilkes, "A Comparison of Implementing a Single Problem in Relational, Object-Relational and Object-Oriented Database Systems", American Society for Engineering Education (ASEE) Annual Conference, June 2008.

3. http://www.oracle.com/technology/software/products/database/xe/index.html

4. http://www.softlist.net/program/db4o-software.html