

The Forest and the Trees: Using Oracle and SQL Server Together to Teach ANSI-Standard SQL

Gary B. Randolph

Purdue University
319 Cottage Ave.
Anderson, IN USA
01.765.641.4563

gbrandolph@purdue.anderson.edu

ABSTRACT

Students in a sophomore-level database fundamentals course were taught SQL and database concepts using both Oracle and SQL Server. Previous offerings of the class had used one or the other database. Classroom experiences suggest that students were able to handle learning SQL in the dual environment, and, in fact, benefited from this approach by better understanding ANSI-standard versus database-specific SQL and implementation differences in the two database systems.

Categories and Subject Descriptors

H.2 [Database Management]: Query languages

H.2.3 [SQL]

H.2.4 [Microsoft SQL Server]

H.2.4 [Oracle]

General Terms

Languages

Keywords

SQL, database, teaching in IT, training vs. education

1. INTRODUCTION

A problem arises in many technology classes. The instructor wishes to teach principles and concepts of a technology. To give students hands-on experience putting those theories to work, a specific product that implements that technology is selected for a lab component. Suddenly the students are learning more about the specific product than they are about the technology concepts. They may or may not realize what is specific to that product and what is general to the technology. Students may even start referring to the course as a VB course, a PHP course, or an Oracle course when what you wanted to teach was programming, web scripting, or database principles.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CITC'03, October 16–18, 2003, Lafayette, Indiana, USA.

Copyright 2003 ACM 1-58113-770-2/03/0010...\$5.00.

This paper presents the experiences from a database fundamentals course that used both Oracle and SQL Server so that students would better understand ANSI-standard SQL. Though each database is ANSI SQL compliant, there are definite differences in implementation (Gorman, 2001; Gulutzan, 2002). By learning each implementation and how each departs from ANSI-standard SQL, students can be better prepared to work with any database and better understand general concepts of databases and SQL. The paper discusses the observed results from this approach and how well the approach met learning objectives.

2. COURSE CONTEXT AND LEARNING OBJECTIVES

CPT 272, Database Fundamentals, is a sophomore-level database programming and design class taught primarily to computer technology majors in their fourth semester. Students will have previously taken a freshman-level course that introduces them to databases as a tool for learning general information system development terms and concepts. The freshman-level course uses Microsoft Access because it is easy to use for quickly developing a small personal information system. That course also introduces both SQL and Query By Example methods for querying a database as well basic database design concepts, which are applied for simple data models.

Students then move into two programming courses, the second of which uses single-table SQL statements for providing data to (formerly) Visual Basic or (currently) web-programming applications. So by the time the students take the Database Fundamentals course they have a concept of what a database is and how it is used as the back-end for programming. The Database Fundamentals course is the course where students learn SQL in depth, database concepts, and basic database design. It does not teach stored procedure programming, triggers, or enterprise or distributed database design, which are covered in more advanced courses.

The learning objectives for the Database Fundamentals course are:

- To understand the fundamentals of a relational database.
- To understand the fundamentals of client-server and multi-tiered applications.
- To understand the principles and characteristics of good relational database design.

- To design entity relationship models for a business problem domain verified by the rules of normalization (through third normalized form).
- To build simple to moderately complex data models.
- To write simple to moderately complex SQL to query a multiple-table database.
- To write data manipulation language (DML) SQL to insert rows, delete rows, and update rows.
- To understand the concept of database transactions and demonstrate the proper use of commits and rollbacks.
- To write data definition language (DDL) SQL to create and drop tables, indexes, and constraints.
- To understand and be able to implement the fundamentals of security and permissions in a database.
- To explain the benefits of using views and write SQL statements to create views.
- To create and use SQL scripts and use SQL to build scripts.
- To gain a working knowledge of query optimization, performance tuning, and database administration.
- To apply team skills to build a client-server database application.

3. CONSIDERATIONS FOR CROSS-ENGINE SQL EDUCATION

CPT 272, Database Fundamentals, is taught in a multi-campus university. It was initially taught on the main campus using Oracle. When the course was rolled out to the regional campuses, SQL Server was first used because of administration considerations involved with Oracle. Now WAN connections have been established that allow the use of either database engine or both.

During the spring 2003 semester one regional campus experimented with the use of both databases. The reasons for doing this were:

- To accomplish the course learning objectives in SQL necessitates going beyond ANSI-standard SQL into database-specific functions, sub-queries, and other aspects of SQL that are implemented differently in different databases. If the students learn only Oracle or SQL Server (or any other database) they are likely to confuse ANSI-standard SQL with the database-specific implementation, which can hinder them when they enter the job market. By using both databases, it was hoped that students would learn and understand the differences among ANSI-standard SQL, Oracle SQL, SQL Server T-SQL.
- Some design considerations, such as Identities/Sequences and datatypes, are implemented differently in different databases. Again, students will enter the job market with a stronger understanding if they understand the difference between the concept and how it is implemented.
- Neither Oracle nor SQL Server commands a majority of market share. However, the two together make up about fifty

percent of the current market share, positioning students well for the job market (Wong, 2002).

- Studying two databases together opens the door for discussing the pros and cons of these and other databases, including DB2, MySQL, and Sybase.
- Finally, students often want to install a database engine on their personal computer and work on lab assignments at home. Both Oracle and SQL Server have licensing and hardware requirement issues that on any given computer may preclude one or the other. Using both allowed most students to do at least some of their work at home.

To implement a cross-engine approach, an SQL text would be needed that taught both databases. A special textbook was created by two of the instructors, a draft of which was used in CD format. In addition to covering SQL essentials in Oracle and SQL Server, it also covered Microsoft Access and MySQL in hopes that it might also be used in the freshman course and be a good reference for real world web programming. The text has since been picked up by a publisher.

4. COURSE DESIGN AND ASSIGNMENTS

CPT 272, Database Fundamentals, consists of a both a lecture and a lab component. The lectures cover fundamental database concepts, including SQL concepts, query optimization, and database design and normalization. The lab component focuses on mastery of SQL. Table 1 shows the labs that were assigned and the database used for each.

Table 1. Course Lab Schedule

Lab	Lab DBMS
Single Table Select	Oracle
Aggregates & Sub Queries	SQL Server
Joining Tables	Both Oracle & SQL Server
DBMS Specific Functions	Both Oracle & SQL Server
Advanced Queries	Oracle
Data Manipulation	Student Choice
Database Definition	Student Choice
Privileges	Student Choice

The first three labs concentrated as much as possible on ANSI-standard SQL. Of course, implementation of sub queries and joins is in some cases different between Oracle and SQL Server. These differences were taught and discussed. However, those labs avoided all DBMS specific functionality, such as concatenation, date manipulation, and datatype conversion. These things were covered in the DBMS Specific Functions lab. This approach limited some of what could be done in the first labs but provided a solid distinction between what was ANSI-standard SQL and what was database-specific. Later labs used these database-specific functions in various lab exercises so that these functions, which are crucial in the real world, were mastered.

In addition to the use of both databases in labs, lecture material constantly referred to how various design, administration, and optimization concepts would be applied in both databases and in other databases, such as MySQL. In addition, exam questions asked students to compare the capabilities of each database. Other

assignments led students into an exploration of the pros and cons of various database engines. Two of these were for students to write short papers on the following:

1. Research one of the following databases: DB2, Sybase, Informix, MySQL, PostgreSQL, or SQLWindows Solo. Write a 2-3 page paper comparing it to Oracle and SQL Server. Include your recommendations regarding the circumstances in which this database should be used.
2. Research what people on the Internet say comparing SQL Server and Oracle. Based on their perspectives and your own experience with these two databases, write 1-2 pages comparing them.

5. DISCUSSION

With all the course objectives listed above, the Database Fundamentals course makes for a full semester. Using two databases instead of one definitely adds to the challenge of fitting in all the material. This first attempt led to the realization of several “kinks” that would need to be worked out before it could be attempted again. These are listed below.

- In most cases any given SQL lecture had barely enough time for covering the SQL concepts and implementation in both databases, forcing the instructor to scrimp on in-class examples. This meant that students had a more shaky foundation going into the lab assignment. However, it should be noted that students did about as well on lab assignments as prior classes. One possible solution would be to focus each SQL lecture on only one database, but to revisit that concept in the following lecture when the other database is used.
- In past semesters using only one database, it was possible to include a discussion of datatypes along with the DDL lecture. Using two databases the DDL lecture had to be expanded, which did not leave enough time for thorough coverage of datatypes in both databases. A solution would be to move the datatype material to one of the design lectures, emphasizing datatyping as a design step.
- By switching between databases, database-specific syntax never clicked in students’ minds. They seemed less able than prior classes to apply concatenation and datatype conversions without looking up the syntax in reference material. While this is regrettable, it may be a worthwhile trade-off for gaining an understanding of what is ANSI-standard vs. database-specific SQL. It is likely that when students move to the real world and settle in with one database, they will quickly be able to internalize the syntax.

- Some students expressed that they would have preferred going through all labs with one database and then looking at differences with a second database. However, other students liked working with both databases side by side. This suggests that the alternating structure may need to be tweaked. But whatever one does, it will probably not work with every learning style.
- After using both databases, almost all students, when given a choice, used SQL Server. This was solely because of a perceived superiority in the user interface of Query Analyzer versus SQL Plus. In itself this is not a bad thing because one of the goals was to understand the differences in the two databases. But in future semesters the instructor may want to force a choice more often to insure that students will be exposed more equally to both.

These “kinks” aside, both instructor and students considered the experiment a success. Their papers indicated a mature appreciation of the differences between Oracle and SQL Server and how they compared to other databases. Their comments in class indicated that they understood what was ANSI-standard SQL and what was not. In the SQL lab exam these students performed as well as previous classes of students, indicating that the cross-engine approach did not hinder their learning. Students also indicated enthusiasm for being able to list experience in both databases on their resume. Compared to prior semesters, students left the course more comfortable and able to use either of these major database engines to accomplish the goals of a given information system.

6. REFERENCES

- [1] Gorman, Michael M. (2001). Is SQL a real standard anymore? The Data Administration Newsletter (TDAN.com) <http://www.tdan.com/i016hy01.htm> (17 Apr. 2003).
- [2] Gulutzan, Peter. (2002). Standard SQL. <http://www.dbazine.com/gulutzan3.html> (17 Apr. 2003).
- [3] Wong, Wylie. (2002). IBM passes Oracle in database market. <http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2864350,00.html> (19 June 2003).