Selection and Use of MySQL in a Database Management Course

James W. Denton
A. Graham Peace
College of Business and Economics
West Virginia University
Morgantown, West Virginia 26506-6025
jim.denton@mail.wvu.edu   graham.peace@mail.wvu.edu

ABSTRACT

This paper presents a case study of the selection and use of a software package for an introductory Database Management course in a typical MIS program. Teachers of Database Management face the challenge of providing their students with meaningful experiences with actual database software. The software selected for use in a database course can generally be categorized as one of three types: commercial enterprise software, such as Oracle or IBM's DB2; personal database software, such as Microsoft Access; or software available for no cost (including open source software), such as PostgreSQL or MySQL. The advantages and disadvantages of each of these types of software are discussed, as is the selection process utilized in this specific case. The teaching approach examined in detail is the use of MySQL on a Linux platform to allow students to create, modify, populate, and query databases. This approach is shown to have several advantages:

- The software is available at no cost to the students or the institution.
- It is configurable and manageable by the course instructor without the need to consult specialized database professionals.
- It provides an enterprise database experience using Structured Query Language (SQL).
- Using the Internet, it is available to students from remote computers.

Keywords: DBMS course software, Teaching Structured Query Language, Database servers

1. INTRODUCTION

A problem that is often encountered in an introductory Database Management course is how best to provide a meaningful hands-on experience for students. Ideally, software should be available for students to perform homework exercises, experiment with database concepts, and bring their projects to life. Such software should simulate as closely as possible an enterprise database experience. However, given the complexity and cost of the most popular enterprise-wide systems and the limited free time of most instructors to learn, install, implement and teach with such systems, careful consideration is required to assure the selection of software that will offer the most valuable educational experience to students. It is important for the instructor to spend the necessary class time teaching the concepts of database management, while not being overly burdened with learning and teaching the details of a complex technical software product.

This paper details a case study of the selection of software for an introductory Database Management course in a Management Information Systems (MIS) curriculum at a large, Mid-Atlantic state university. The course emphasizes the role of databases in organizations, entity-relationship modeling, logical database design, physical database design, and Structured Query Language (SQL). After careful consideration, the instructors decided to use MySQL database management system software as a teaching tool in the course. The factors used to select the software are discussed; the experience of teaching with MySQL is described; and the lessons learned are presented. This information should prove useful to other instructors faced with the prospect of teaching an introductory Database Management course.

2. SOFTWARE CHOICES

Courses in database management at universities often utilize database management system (DBMS) software that represents commercial enterprise software systems such as Oracle, personal database software such as Microsoft Access, or some combination of both. This paper describes a third alternative: open-source (or freely

available) database software. Each of these options has advantages and disadvantages in terms of their cost to the institution, the quality of the experience provided to the students, and the ease of learning and use for both the students and the instructor. The following paragraphs describe in more detail each of these classes of database management software and the leading product packages available in each class.

According to a 2003 Gartner study, the leading relational database management systems (RDBMS) used in business today are IBM's DB2, Oracle, and Microsoft's SQL Server, with a combined revenue from new licenses in 2002 of $6.6 billion. Oracle software accounts for nearly 43% of RDBMS market share, with IBM holding a 24% share, and Microsoft a 23% share, based on 2002 new license sales (Graham and Strange, 2003). It is noted that, as these share values are based on licensing revenues only, no open source or free software packages such as MySQL are included in those figures. However, MySQL claims that their software products now represent over 20% of all RDBMS installations worldwide (Fegreus, 2002).

Since IBM, Oracle, and Microsoft have such dominant name recognition in the DBMS world, students often desire experience with those systems in order to enhance their job prospects. However, there can be problems with using these packages in an educational setting. These complexity and cost considerations will be discussed in more detail in the next section.

Personal database software includes those products that are designed to serve a single user or a small group on a PC platform. Microsoft Access is the leading product in this category. In contrast to complex and costly enterprise database software, Access is a more attractive option for many educational institutions. Most business school computer labs use Microsoft Office products, and installing Access and making it available to students is relatively simple. However, a personal database package does not provide the same quality of experience as a large enterprise system. Personal database packages often emphasize ease-of-use by hiding details that are important to the effective teaching of database concepts.

While Microsoft Access is the clear leader in the personal database arena, there are other possibilities. Filemaker Pro is popular on Apple systems and has made minor inroads on the Windows operating system platform. Paradox and Lotus Approach are also available at reasonable prices, but due to the ubiquitous nature of the Windows environment and the Microsoft Office suite at most business schools, it is difficult to make a case for using any of these products as a substitute for Access.

Open source or free software presents an alternative to commercial enterprise and personal database software. Open source products are developed by networks of loosely linked individuals who work on software projects voluntarily. Their products, including the source code, are freely distributed and may be copied and further refined by others. Two of the remarkable success stories of the open source movement include the Apache web server and the Linux operating system.

Gurley (1999) recognized that open source software's distributed development results in faster product development than proprietary software, which is developed in a controlled environment by a relative few. In addition, the quality and reliability of open source software meets or exceeds that of proprietary software. It has been estimated that 63% of the world's web sites are served by Apache systems, and Linux, regarded as more reliable than Microsoft Windows 2000, commands a 33% (and growing) share of the web server operating system market (King, 2002; Netcraft, 2003).

Open source database software choices include PostgreSQL and GNU SQL. PostgreSQL (http://www.pgsql.com/) started as an academic research project at the University of California, Berkeley. It is an extension of Ingres, one of the first relational database products, which was also developed at UC-Berkeley. After Ingres became a commercial product, development efforts continued on PostgreSQL in order to fulfill the basic need in the Internet community for an open source, SQL-based, multi-user database. PosgreSQL is based on the object-relational database model, and it is considered to be one of the most advanced open source RDBMSs available based on its compliance with the ANSI SQL92 standard.

GNU SQL server (http://www.ispras.ru/~kml/gss/) is a free, portable, multi-user relational database management system currently being developed as an open source product. It features full ANSI SQL89 compliance and some SQL92 extensions. It does not, however, offer much in the way of third-party educational materials for novice users, and unsophisticated users will likely have difficulty implementing the system. For example, a search at the online book retailer Amazon.com revealed no hits for "GNU SQL", eleven hits for "PostgreSQL", and 55 hits for "MySQL".

MySQL (http://www.mysql.org/), the most popular "free" RDBMS in the world, is not an open source product, but it may be used without charge for non-commercial use on non-Windows platforms under the terms of the GNU Public License. It was developed by Sweden-based MySQL AB and is undergoing continuous revision. Over 4 million installations of MySQL exist, including production systems at companies such as Yahoo!, Cisco, Google, the US Census Bureau, NASA, Motorola, Texas Instruments, Silicon Graphics and others.

While some users may think that free software must be inferior to commercial products, MySQL has been recognized as a true enterprise RDBMS. MySQL 4.0 was recently evaluated in a review in PC Magazine as "one of the top five databases" along with IBM's DB2 Universal

Database 7.2, Microsoft's SQL Server 2000 SP2, Sybase's Adaptive Server Enterprise 12.5, and Oracle's 9i Database (Dyck, 2002). Although MySQL does not have as many features as its competitors, its cost (free) certainly compares favorably.

The philosophy of the MySQL developers is to emphasize reliability and performance rather than fancy features. This is consistent with educational objectives since fancy features sometimes obscure the teaching of basic concepts. MySQL is known for its clear and uncomplicated administration, which is also valuable in the academic environment. It can be administered using command-line text entries and accessed remotely via a simple telnet connection through the Internet. MySQL is also less demanding on resources. Kientzle (2000) reports that a MySQL Internet application can run with an Apache web server on the same Pentium 200 machine utilizing only 64 MB of memory.

3. SELECTION CRITERIA

The factors considered in selecting a RDBMS for our course included cost, ease of use, security, functionality, job market appeal, and compatibility. Each of these factors will be discussed in the following subsections.

3.1. Cost
The most obvious (and usually the largest) cost associated with database software is the purchase price. However, other components of the total cost of ownership should also be considered when choosing software. These may include costs associated with obtaining and supporting the operating system and hardware platform required for the database, training costs for faculty or staff, maintenance costs, and the risk of obsolescence.

Purchase costs for enterprise systems vary from about $4,000 for Sybase, $20,000 for DB2 and Microsoft SQL Server, to approximately $40,000 for Oracle. These prices are out of reach for most educational institutions. However, most major software vendors have implemented educational discount programs, so instructors should always check with the vendors and obtain up-to-date pricing information. For example, Oracle Corporation has established the Oracle Academic Initiative (http://oai.oracle.com /) to provide their software to educational institutions at very low cost. In addition to relational database software, Oracle also offers electronic commerce, enterprise resource planning, systems analysis, and systems development software. Therefore, it may be possible to design an entire curriculum around Oracle products. Consideration must be given, however, to availability and cost of the platform used to host the database system. Enterprise systems, such as SQL Server and Oracle will require more advanced platforms than the relatively small and cheap PCs required for Access and MySQL.

For personal database software, licenses for Microsoft Office, which includes Access, can be obtained for about $48 per PC with a minimum of 300 units. While this might be prohibitive if it were purchased solely for the database software, most schools can also utilize the other products in the software suite. Indeed, multi-user licensing agreements may already exist at many institutions for Windows and also for Microsoft Access, making that a very affordable option.

A clear advantage of MySQL is that it may be obtained free of charge for most Unix-based systems. Code can be easily downloaded from the MySQL web site, and supporting materials are freely available online. In addition, MySQL can be run on the open source Linux operating system, eliminating any related costs for system software (although this may increase learning time for the instructor).

3.2. Ease of Use
Ease use includes the ability to install the software without difficulty, maintain the software with minimal effort, access the software from remote locations, find support and documentation easily, and quickly learn how to administer and use the software from both faculty and student perspectives.

Enterprise packages such as Oracle, SQL Server and DB2 are extremely complex, and require a significant amount of time and effort to properly install, maintain, set up and learn. However, if the instructor is willing to invest the time and effort, Oracle and SQL Server can be installed and operated in the classroom environment. IBM DB2 is a less friendly option, as it is designed more for the mainframe environment than PC-based systems. Flatto (2000) has documented his experiences with Oracle software, and has collected the comments of other faculty with similar experiences through his "Oracle in Academia" mailing list (http://business.uindy.edu/oia.html). It is quite common for users to have problems installing the software, configuring the software properly, enabling remote access for students, and finding answers to problems in the software's documentation. A major source of difficulty is the lack of specific Oracle training for the faculty and staff who are attempting to administer the system. While the software itself may be inexpensive, the time needed to learn proper use and implementation of the software and the cost of training programs and materials can be significant. Academic personnel usually do not have the level of training and support that exists for the database administrators for Oracle products in the business world.

Given that almost everyone in the MIS field is familiar with the Microsoft Windows and Office environments, Microsoft Access has a clear advantage in this area. Access is easy to install and relatively easy to learn and use. Many excellent books and tutorials are available both in print and online, and maintenance issues are minimal. However, one apparent benefit of Microsoft Access may be in fact a detriment. While Access can be run on students' home PCs and in various computer labs around campus, it is unlikely that all student and lab PCs will be

identically configured, and compatibility and support issues may arise. The students, MIS staff, and instructors need to clearly define who is responsible for compatibility and support issues on these remote machines. Server-side packages, such as MySQL or Oracle, avoid these issues. The use of server-side packages also reduces the probability of a student losing his or her work due to the failure of a home PC.

MySQL is easy to install and relatively easy to use and maintain. Many support materials exist, both in print and online, and the command line interface is not difficult for students to understand. A concern with a MySQL installation, however, is that the operating system of choice will most likely be Linux, or some other Unix-based variant. Linux is not as easy to learn or use as the familiar Windows operating system, but supporting materials are readily available. In the MySQL environment, the instructor may be required to spend some time learning how to administer a Unix system.

3.3. Security
Security is an important factor in the software selection process, especially when the software may be accessed remotely via the Internet. Some risk can be avoided by using secure shell (SSH) programs rather than the older telnet protocol for remote access. One such product is OpenSSH (http://www.openssh.com/), which provides secure remote administration over the Internet in contrast to the telnet connection.
Of course viruses are also a security concern. The recent SQL Slammer virus impacted SQL Server installations, but had no impact on MySQL systems. Overall, if the instructor is diligent, MySQL on a Linux system has a security advantage over a Windows-based, SQL Server or Oracle installation, but good security options are available for each.

3.4. Functionality
The software system chosen must provide the functionality needed to achieve the pedagogical goals of the course. In an introductory course, the major goal is to teach the basic concepts of database design and to provide some experience using SQL. The software package selected must allow the student to create the database and perform basic queries and updates. Enterprise database products have a rich array of capabilities that will more than satisfy the requirements of an introductory course, and all utilize SQL as the basis for interacting with the DBMS. While these systems are clearly superior with respect to overall functionality, it is very unlikely that the students will utilize more than a small fraction of the tools and features available.

Microsoft Access is the easiest of the packages to use, but it is most likely too easy to use. Such personal database packages have been designed to emphasize ease of use. Reducing complexity and eliminating detail-oriented tasks advance the marketing goal of maximizing the base of potential users. This is accomplished largely through the use of a sophisticated GUI to perform many database tasks, so that even unsophisticated users can create and use a database. The user interface simplifies the creation and manipulation of databases by hiding many important details from the user. While this is desirable for users who want to get their databases up and running quickly, it can defeat the purpose of a database course.

Since it is precisely these details that we want to teach, it is important to expose, rather than hide, these fine points to our users, the students. For example, Access allows users to create tables by simply entering data without specifying the table's structure. In addition, queries are normally generated in Access by using either an automated wizard procedure or a visual query-by-example (QBE) grid. Neither the wizard nor the QBE grid requires knowledge of SQL, which is a major teaching objective of most database classes. While it is possible to bypass the QBE grid and generate Access queries with SQL, the version of SQL provided is non-standard.

On the other hand, MySQL utilizes a large subset of the ANSI SQL99 standard and provides all of the functionality necessary for an introductory course. The student must define and create the necessary tables, field types, indexes, keys, etc., and they must use SQL code to manipulate and query the data in the database. MySQL does fall short in some areas: it is unable to perform cascading updates and deletes (MySQL does not enforce foreign key relationships); it lacks commit and rollback operations for transactions; and it does not support subqueries. (Support for subqueries and transactions is scheduled to be implemented in future MySQL releases.)

3.5. Job Market Appeal
The software selected for a database course can provide experience for students that is desirable to potential employers. Since Oracle, DB2 and SQL Server are the leading business RDBMS systems, it is expected that hiring organizations will find experience with these packages desirable. However, as previously noted, many organizations are also utilizing MySQL as an enterprise DBMS. MySQL recently announced an alliance with SAP, along with a $19.5 million venture-capital round of financing, giving MySQL the resources to upgrade its products and better compete in the marketplace. SAP will allow MySQL to develop the database component of their enterprise system software, giving MySQL much greater visibility and a powerful ally in the marketplace.

As MySQL is best suited to a Linux platform, the students may gain some benefit by being introduced to Linux, while learning MySQL. 41% of managers at Fortune 500 companies responding to a recent survey identified Linux as an important skill for business professionals in 2005, up from 29% in 2002 (Zhao, 2002). While the nature of this course precludes an in-depth coverage of Linux, some exposure to this operating system can only benefit the students (and, perhaps, the instructor). It should also be noted that Linux is gaining a foothold in the business

community, and is run on over 33% of the world's web servers (King, 2002). However, SCO's $1 billion lawsuit against IBM may hamper market share growth in the near term (Port, 2003).

Finally, given that good database design and SQL knowledge are really independent of the software platform and that the students will most likely fail to gain more than an elementary experience on the platform chosen, the popularity of the specific product chosen is not as important as other selection factors.

3.6. Compatibility

Ideally, the software chosen for the introductory database course should fit well with both the technology platform available at the university as well as the rest of the MIS curriculum, thus reducing overall program costs and enhancing synergy. Almost all academic institutions have standardized on a Microsoft Windows environment, making the Unix-based software choices less desirable. In our specific case, we utilized a small Linux platform independent of the institution's Windows environment. In addition, the choice of Linux and MySQL fit well with our follow-on courses. These included a web development course, using the Apache web server and PHP, which built on the Linux and MySQL knowledge acquired by the students in the database management course. All software used in both courses is available at little or no cost and effectively presents the concepts necessary for students to perform well in the business world.

3.7. Our Selection

MySQL on a Linux platform was determined to be the best available option for our course. All software utilized is either open source or free, requiring no upfront cost. Training and support materials are also freely available on the web, and many good books can be found for under $50. From the students' perspective, once they understood how to connect via the Internet and learned some rudimentary Linux commands, operating the DBMS and manipulating the data via SQL with the command line editor was relatively simple. This allowed them to focus on proper database design and SQL coding for the bulk of the course. Finally, the skills learned in the database course reduced the learning curve in subsequent courses.

The biggest problem with the selection of MySQL is the need for the instructor to learn enough Linux to be proficient at installing the software and maintaining the Linux system. However, this is not an insurmountable problem, and may even be an enjoyable and interesting experience for an MIS instructor. Another drawback is that students prefer to acquire database experience with a higher-profile name such as Oracle, DB2, or SQL Server. However, the skills learned in the course are easily transferable to these systems, and the tradeoff was deemed to be worthwhile.

4. IMPLEMENTING AND USING MYSQL IN THE COURSE

In our specific situation, a Pentium 200 machine with 64 MB of RAM and two 2 GB hard drives running Red Hat Linux and MySQL, was the chosen operating environment for the Database Management course. The use of Red Hat's Linux distribution added some cost, but it provided additional support, which was deemed to be worthwhile for novice Linux users. The server was set up with a user account for each student in the class. Students logged into the server with their User ID and password, via the Internet using telnet. Since a telnet client is usually available on Windows PCs, students could log in from any campus computer lab or from their own computer and Internet connection. (For future classes, we plan to utilize a safer secure shell connection.) A total of 75 user accounts were established for two sections of the course, with no performance problems noted.

In addition to a user account on the Linux server, each student must also have permission to access one or more MySQL databases. Full permissions were granted for students on their own databases, which allowed them to create, modify, and delete tables within that database. The students were initially asked to create relatively simple databases containing three or four tables, each with five or six records. This allowed the students to practice a wide variety of SQL commands without the needless repetition involved in typing many records. Permissions to perform record selections, but not modifications or deletions, were granted to all students on a much larger sample database. This permitted the students to practice more complex SQL queries without allowing them to modify the data in any way. Shell scripts were also provided to the students to create databases, both to illustrate how database tasks may be automated, and to provide larger data sets for manipulation without repetitive typing. Useful instructions for administering MySQL are available in third-party books such as the volume by DuBois (2000).

As the system administrator, the professor must provide each student with a MySQL user account by logging into MySQL as the "root user". The root user has all administrative privileges and access to all user databases, so the password for the root user should be closely guarded. Databases for each student are created, typically naming that database with the student's initials and last name. Then, using the MySQL "grant" statement, privileges are granted to the students' Linux user IDs to allow them to create, drop, and alter the structure of tables; to insert, delete, and modify records; and to select records and columns for queries. In MySQL, a database is equivalent to a Linux directory, and a table is equivalent to a file, so security can be assured through the operating system's built-in file permissions, which determine which users may read, write, and execute files.

A useful side effect of using Linux-based MySQL is that students are exposed to the Unix command-line

environment. For many students, this represented their first excursion outside of the Microsoft Windows world. Brief training was provided for simple Linux commands to navigate through the file system; copy, move and delete files and directories; set permissions; change passwords; and use a simple text editor.

In all, the course was very successful. Only minor problems were encountered in the installation of both the Red Hat Linux and MySQL products by a professor with no previous experience in using either product. These problems were quickly solved after consulting readily available references. The students were able to learn to use the software with minimal training, allowing the instructor to focus on more relevant database topics. The majority of students were able to complete their assignments on time, and almost all student problems involved a lack of understanding of course material, as opposed to issues involving the actual use of the MySQL system. Finally, it should be noted that the students were very well prepared for the following term's course on web development, which included the use of a Linux server running Apache, MySQL and PHP.

5. LESSONS LEARNED

There were several lessons learned, during this process. First, planning is imperative, especially if a software installation will be required. In our case, both the installation of the DBMS and the operating system were required. Depending on the capabilities of the installation personnel, some time for learning the details of installation may be required. Our system was installed and brought up to speed in less than two weeks by installers that had no previous Linux or MySQL experience.

It should be noted that, in our experience, Linux is an excellent and easy-to-use system that is stable and predictable most of the time. However, if anything out of the norm occurs, it can be difficult to diagnose and fix problems without traversing a significant learning curve. It is important to spend some time working on the Linux system and making sure that it is properly running prior to the use of the system in a class setting. The installation of MySQL was simply a matter of downloading a file and executing an installation command.

Student response to the course was uniformly positive. Students found project assignments using MySQL to be both valuable and enjoyable. Comments on student course evaluations showed that they believed that they had learned significant and valuable skills. Many students who had previous experience with Microsoft Access found that they had acquired a more complete knowledge of database topics after using MySQL. One student commented that "Access did things that I didn't understand," and that it was "harder to make a stupid mistake with MySQL." However, there were some students that would have rather have used database software with higher name recognition, such as Oracle. Many students who had heard about

Linux, but had no experience with it, were pleased to get some hands-on experience with this "new" operating system, and the lack of a sophisticated GUI was not seen to be a problem. Finally, many students expressed an appreciation for the ability to connect to the system via the Internet from their home PCs. The use of MySQL provided the advantages of low cost, high connectivity, and relative ease-of-use, while providing the students with the feel of an enterprise database system.

6. REFERENCES

DuBois, Paul (2000), MySQL. New Riders Publishing, Indianapolis.

Dyck, Timothy (2002), "Clash of the Titans", PC Magazine, March 26, pp. 122-138.

Fegreus, Jack (2002), "Building a Global Ecosphere", Open, October 10, http://www.open-mag.com/features/Vol_41/mysql/mysql.htm.

Flatto, Jerry (2000), "Implementing Oracle Products in the Curriculum: The Good, the Bad, and the Ugly", Proceedings of the Americas Conference on Information Systems, August 10-13, Long Beach, CA, pp. 2153-2155.

Graham, Colleeen, and Strange, Kevin H. (2003), "Gartner Study Shows Worldwide RDBMS Market Declined in 2002", Gartner Research, http://www4.gartner.com/DisplayDocument?doc_cd=115036.

Gurley, J. William (1999), "Why Free Code Makes Sense", Fortune, vol. 140, no. 3, August 2, p. 228.

Kientzle, Tim (2000), "Database Engines: MySQL Versus Oracle", Dr. Dobb's Journal, July, pp. 98-104.

King, Rawlson (2002), "Linux Market Share Within Web Server Sector to Grow", TheWhir.com, January 7, http://thewhir.com/features/linux-market-share.cfm.

Momjian, Bruce (2000), PostgreSQL: Introduction and Concepts. Addison-Wesley, Boston.

Netcraft (2003), Netcraft June 2003 Server Survey, http://www.netcraft.com/survey/Reports/200306/byserver/index.html.

Port, Otis (2003), "Will This Feud Choke the Life Out of Linux?", BusinessWeek, July 7, p. 81.

Zhao, Jensen J. (2002), "Computer End-User Skills Important for Business Professionals Now and Toward 2005", Journal of Computer Information Systems, vol. 42, no. 3, pp. 38-43.

AUTHOR BIOGRAPHIES

James W. Denton is an Associate Professor in the College of Business and Economics at West Virginia University where he has taught courses, in basic information systems, systems design and development, and programming. Prior to receiving his Ph.D. from Kent State University, Dr. Denton worked in Engineering and Quality Assurance in several industrial settings. He has previously published research on neural networks and information systems education in the *Journal of Computer Information Systems*, the *European Journal of Operational Research*, *Accounting, Management, and Information Technologies*, and others.

Graham Peace is an Associate Professor of MIS at the College of Business and Economics, West Virginia University. He received his PhD in MIS from the University of Pittsburgh in 1995. His teaching interests include Systems Analysis and Design, eBusiness and Database Management Systems. Dr. Peace's research interests include the ethical issues of Software Piracy, Privacy, Censorship and Freedom of Speech. He has had articles published in several journals, including *Communications of the ACM*, the *Journal of MIS*, the *Journal of Computer Information Systems*, *Business and Society Review*, and the *Journal of Information Ethics*.

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.