

The Database Imperative in Computer Graphics Projects

Ronald J. Glotzbach
Purdue University

Abstract

Applications of ideas and projects in Computer Graphics Technology have developed more and more into requiring a database to drive some or all of it on the back-end. Is an imperative developing where universities are more in need of producing students with database development or administration capabilities? This paper focuses on engineering projects with database involvement, attempting to determine the level of knowledge students should have and in what areas. Web-development projects in particular typically require at least one database on the server. In addition, many multimedia and manufacturing projects require databases, anything from Product Data Management systems on a company intranet to collaborative multimedia in a classroom. What requisite knowledge does a student need to competently work with a database that comprises only a portion of a larger project? Which database management system(s) should a student learn and does one provide an advantage over the others?

Introduction

Applications of ideas and projects in Computer Graphics Technology have developed more into requiring a database to drive some or all of it on a server. The Web has grown past the days of static HTML pages and into an age of dynamics⁴. Designing the user interface was a typical job when web sites were static. Only being able to produce the user interface is no longer enough; now the typical web developer is tasked with that and the creation of dynamic content, which is typically driven by a database residing on the server. Other areas also have the same database imperative, such as multimedia and manufacturing applications. These areas have grown to see personal portfolios, interactive games, collaborative data projects and data management systems where it is not uncommon to have a database back-end.

With these ideas and projects in mind, a Computer Graphics Technology student's requisite knowledge of databases and database management systems should be strong in three primary areas: theory, design, and application. This paper will discuss each of these as they apply to Computer Graphics projects.

Theory

When focusing on database-related subjects, students should have a strong background in theory, design, and application. A lacking in one of these areas would be a detriment to the student's education. To some extent, the three build upon one another. It is necessary to know the theory of database architecture before starting to design a database³. Likewise, proper design concepts need to be taught and then applied so that students are prepared for real-world applications.

Database theory is taught in many computer courses now; having a computer curriculum simply requires this knowledge, as can be seen in both technology and computer science programs at Purdue University. This is because most client-server systems integrate database management systems, thus requiring knowledge of those systems as well. Typical 3-tier architectures for web development have a user interface component, a business logic component, and a database management system⁶. The web development curriculum in Computer Graphics Technology (CGT) structures itself around this type of architecture.

Database theory involves the evolution from file systems to current database systems. There are entire languages that must be learned, such as Data Manipulation Language and Data Definition Language, which both use Structured Query Language (SQL) for formulation of queries. These areas should be taught to the students seeking to get involved with multimedia, manufacturing, or literally any other type of web project. For students using Product Data Management (PDM) systems, while they may not have to create a system, having an understanding of how the system was designed and built will further enhance their competency in this area. Check-in and check-out procedures are a typical operation dealt with, along with other topics such as versioning control.

In the web development area, whether the focus is personal, small business, commercial, industrial, or any other organization with the potential for web-related projects, it is imperative that students know how to select information, update existing information, insert new data, and be able to delete data that is no longer required. All of these fall under the umbrella of Data Manipulation Language¹. Without this knowledge, it would be extremely difficult to teach students how to generate dynamic content on the Web. Examples of this type of interaction can range from querying a database of engineering components for display on an intranet to tracking user information and performance in an online course, to full-blown e-commerce solutions similar to the popular Amazon.com and bn.com (Barnes and Noble) sites.

Design

Design principles are crucial to any database project. The design of a database, how tables are designed, relationships formed, data validated, and normalized to prevent data redundancy is central to the success of the project. This applies more to the multimedia area, and the web development arena in particular. When a student procures a project in the real-world, it is not likely that the database will already exist for them to utilize. It is more likely that the student will be tasked with designing the database and creating it for the specific purpose for which that project was intended.

Database design is often an arduous task². Many things come into consideration such as number of fields in each relation, data types of each field, length, precision, and scale just to name a few. Beyond that, multi-relation databases have relationships defined that in turn begin to identify the business rules that a student will have to implement as part of the project. Knowing that, students should have developed knowledge of primary and foreign keys and how those relationships are going to mold the queries sent to the database in the later applications of that project.

Application

Some applications of databases in CGT have already been discussed, such as the PDM systems that are almost entirely reliant on databases. Many of the rest of the applications discussed will be web-based solutions. Nearly every company has a need for these solutions, both on intranets and the Internet. A common solution might entail the management of components on a John Deere planter unit. Design elements for the box, brushes, scrapers, packing wheels, and bolt sizes may be tracked via an intranet system that allows employees to upload versions, track progress, and manage the maintenance of existing or new planter units. This would be an agricultural engineering application of the database management system. The solution could use one of several different technologies including, but not limited to, Active Server Pages, JavaServer Pages, PHP, or ASP.NET. All of these are among the most popular technologies that interact with databases to produce dynamic solutions⁵. These technologies have also been used on various projects in CGT, as well as some being taught in the classroom.

Among the most popular database management systems are Oracle, SQL Server, and MySQL⁷, any of which would be a possible choice for a project such as this. All three are very robust systems, providing roughly the same capabilities. SQL Server is a large scale solution, but can be costly and is typically confined to the Microsoft platform. Oracle is also a large scale solution, but is quite expensive and while it can be used with any platform, it is typically used in conjunction with UNIX operating environments. The third, MySQL, is again a large scale solution, but unlike the previous two, it is free and platform-independent.

When deciding what database management system to use, it would be best to expose students to one of these major systems when possible, as they will likely be the system of choice in industry. It would be acceptable to do some development on Microsoft Access; however Access is generally limited to the Microsoft platform and is not a robust system in that it does not handle large amounts of traffic, simultaneous hits in particular⁷. Access would be best suited for personal development before jumping to one of the more robust database management systems. The primary importance in using any one of these systems is learning SQL, which, as discussed earlier, is key in both the theory and design of databases. The system used will largely be chosen based on availability of resources within the department and platform requirements.

Conclusion

There is a tendency toward more database oriented projects in CGT. Students should develop a knowledge base in database theory, design, and application. These areas will provide students in any focus with the skill set to understand most database projects. Using any of the most common database management system packages, students will be prepared for real-world applications. Ensuring that students follow a curriculum that covers these topics will meet the developing imperative of producing students with database development capabilities.

Bibliography

[1] Elmasri, R. & Navathe, S. (2000). *Fundamentals of database systems* (3rd ed.). Reading, MA: Addison-Wesley.

[2] Greenspan, J. (2003). *Your first database*. Retrieved January 13, 2003, from <http://hotwired.lycos.com/webmonkey/99/13/index1a.html>

[3] *MS Access*. (2002). Retrieved January 13, 2003, from <http://www.beta-microsolutions.co.uk/courses/access.htm>

[4] Online Training Solutions, Inc. (2001). *Microsoft FrontPage version 2002 step by step*. Microsoft Press.

[5] Schmidt, R. (2002). *A primer on server-side web technologies*. Retrieved January 13, 2003, from <http://www.finebrand.com/ideacenter/technical-topics/scripting-primer.cfm>

[6] *3- and n-tier architectures*. (1998). Retrieved January 13, 2003, from <http://www.corba.ch/e/3tier.html>

[7] *Web hosting database technologies*. (2003). Retrieved January 13, 2003, from http://www.w3schools.com/hosting/host_databases.asp

RONALD J. GLOTZBACH

Ronald is a Visiting Assistant Professor for the Interactive Multimedia Development area in the Department of Computer Graphics Technology at Purdue University's West Lafayette campus. He earned his B.S. in Computer Graphics Technology and holds a M.S. in Technology. Related interests include dynamic content delivery methods and integration of varying media into highly technological solutions. Contact him at: rjglotzbach@tech.purdue.edu