December 2001

Demonstrating the Client Environment Through Oracle Developer

Mary-Ann Robbert
Bentley College

Follow this and additional works at: http://aisel.aisnet.org/amcis2001

Recommended Citation
http://aisel.aisnet.org/amcis2001/430

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2001 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
DEMONSTRATING THE CLIENT ENVIRONMENT THROUGH ORACLE DEVELOPER

Mary Ann Robbert
Bentley College
mrobbert@bentley.edu

Abstract

This workshop was designed to address configuration and pedagogical issues with Oracle tools as discussed by database faculty at prior conferences. Having used Oracle and Oracle Developer as support tools for the past ten years, I designed this workshop to demonstrate simple configurations for off campus access to a database server, Oracle Developer basics and projects that have worked well with my courses. Workshop participants will create forms, reports and graphics using a sample project database.

Introduction

At a number of conferences database faculty have raised the issue that Oracle is difficult to administer and complex to learn. Though the Oracle Academic Initiative offers a great software package including most of the current Oracle development tools, technical support is limited, the products are sufficiently complex to require assistance and there is a steep learning curve for students. This workshop is designed to present solutions to these issues, as well as other issues that we have experienced in using Oracle as a foundation for the database course.

The introduction to the seminar will contain information on configuring Oracle Developer/2000 and Forms and Reports 6i. Selected team projects suitable for a semester course, hands-on demonstrations of form creation, and generation of graphic results will follow. It is assumed that attendees have some knowledge of Oracle, but not necessarily Oracle Developer and are teaching or planning to teach a database course.

Supporting the Database Course

Oracle Developer makes it possible for students to gain a better understanding of front-end tool capabilities through demonstrations. Students can then implement database concepts covered in the course to solve a problem through application development. In general, Oracle 8i can be configured to provide powerful support for database courses through its ability to present concepts beyond those provided by lower level database products such as Access. The environment can be constructed to allow student access to their own and shared tables on the server through the Internet. If Oracle is the database used to solve problems and demonstrate theoretical underpinnings of databases for a course, then the environment provided by Oracle Developer is a natural to demonstrate client capabilities and application development.

The primary concepts I cover through application development include data independence, security, integrity, data models, referential integrity enforcement and concurrency control. Other topics such as portability and performance can be included. Most importantly, students are taught to focus on the problem not the solution. Defining the problem first, then analyzing causes and possible solutions before designing an application is still crucial. Concepts should arise naturally as the problem is being examined and problem resolutions proposed.

Many of the fundamental concepts are exemplified easily through the Oracle tools. The client/server configuration with the tables remaining resident on the server begins to demonstrate data independence. Sharing forms and multiple users accessing data simultaneously show the concurrency controls available through Oracle and should lead to a discussion of locking, error messages returned and adding additional controls. Placement of integrity constraints can be tested and data entry through views can be demonstrated.
Configuring the Environment

Oracle Developer can be installed in a two or three-tier client/server architecture that is transparent to the students as they build their applications. The simplest solution is the two-tier system with Oracle8i on the server and Oracle Developer on the client. Actually, once the connection is established any ODBC compliant front-end tool or language can be used for application development.

I have been using Oracle Developer 6 on Windows 98, interfacing with Oracle 8i on the server through the Internet. The Oracle Academic Initiatives program had distributed Developer 6 for student use. Students install Developer, which contains the Oracle 8i client, on their own computers and configure their environment following an instruction set and inserting code I have posted on my web page (http://cis.bentley.edu/mrobbert/cs652x.htm). It is necessary to define the server in the tnsnames file and verify the sqlnet.ora file.

I have tested Developer/2000 on Windows 2000 and there were no obvious problems. The configurations for Developer 6 match those for Developer/2000. This provides a common installation format and a common interface for students using Windows 95, 98 and 2000. If students work from home on there own machines consistency among various versions of Windows is essential.

The new 12 CD tool set distributed by Oracle Academic Initiative this year includes Oracle Forms and Reports as part of the Internet Developer Suite. The tool suite has functionality similar to Developer, containing a set of integrated builders and wizards that help developers to construct highly interactive forms and charts with minimal effort. However, the extensible Java client offers developers vast potential to improve web applications with packaged samples. This enables publishing forms on the web and writing Java classes that are wrapped with PL/SQL code.

Student Projects

The problems selected for students’ solution must be complex enough to warrant developing front-end applications and simple enough to be completed in the semester. Developer’s application generator and report writer simplifies development of the applications. The same PL/SQL used in the creation and querying of the database ties together all the needed Developer objects. Built-in subprograms allow manipulation of objects in various components enabling students to create projects with more functionality.

Projects that have worked well with Oracle and have fit within the confines of a semester course include a hospital, a museum, and an auto dealership. Focusing on the hospital as a sample, a basic hospital is defined with patients, doctors and staff. A patient is admitted to a room and incurs charges for the room, doctor visits, treatments and extras. From this point the project gets customized each semester. A team can develop the admission system, creating means to determine empty beds, defining admission forms and generating bills and occupancy reports. Another team can develop a staffing system to assure twenty-four hours a day coverage of doctors and nurses and staffing for the labs at appropriate hours. I have even assigned hospital payroll project. If time permits I have the teams integrate the projects into a single working system sharing the data and accessing the forms and reports through a common Developer menu. For example, the hours an employee works in the staffing system should be fed to the payroll system. The difficulty with this last assignment is team completion of their segment. The team that finishes their segment first is always impatient waiting to synchronize with the next group. Sample assignments will be distributed and demonstrated at the workshop.

Using Oracle Developer

Let us now examine the development environment provided by Oracle Developer, i.e. the framework and the set of tools used to develop applications. The three major components of Oracle Developer are Forms, Reports and Graphics. This workshop will focus on the forms and examine graphics as time permits. The Forms component is interactive and includes the framework to develop menus and PL/SQL library modules. Once students can develop forms the reports follow in a similar manner. The only tools specific skills students need to be taught in class are how to configure the Oracle environment, the capabilities of the required modules and how to access the wizards.

The major components (not the complete set) of the Form Module are the triggers, blocks, canvas-views, windows and list of values (LOV). Triggers can be built-in or user-defined. There is a sufficiently extensive set of built-in triggers making it unnecessary to design new triggers often. The two kinds of blocks are self-explanatory, base-table and control blocks. Blocks
also provide query-by-example features. The blocks contain typed items. The LOV can be associated with an item. A relation is a special object used by the form to define a master detail structure. The canvas-view is the background on which the items are displayed and the canvas-view itself is displayed in a window. Workshop attendees will develop forms with constrained items and menu navigation.

The graphics component contains the display modules that are charts, which may be derived from the database or independent sources. SQL queries can be used to generate the data, though the exact format required for the graph must be provided. This can be done with the group by and sum command. At the workshop, sample code for generation of graphs will be distributed for modification and experimentation.

**Conclusion**

Oracle Developer glues together all the objects, within and without the Oracle database the students want to work with. The primary glue is the PL/SQL language. If Oracle SQLPLUS and PL/SQL are used for the database design then the use of Oracle Developer is a natural extension that allows students to see the full environment. This workshop gives an overview of Oracle Developer as a tool to allow students to implement database concepts learned in class and solve complex problems.

**Workshop Overview**

- Installing Oracle tools
- Sample Oracle project assignments
- Designing Applications with Oracle Developer
- Creating functional forms
- Setting up an Oracle Developer menu
- Using Oracle Developer graphics to display results
- Discussion