ECE 480 DESIGN TEAM 6

Microsoft Access Database Management through LabVIEW and MySQL

Jacob H. Co

Dr. Virginia M. Ayres – Facilitator

Application Note

Friday, April 3rd, 2009

Executive Summary

In performing the quality inspection of products, database management is vital for proper statistical analysis. Through the LabVIEW platform, a MySQL interface can be implemented for Microsoft Access such that inspection results can be automatically stored as soon as they are gathered. This implementation facilitates inspection efficiency, as well as statistical organization in an inspection setting.

Keywords: database, automation, inspection, Microsoft Access, LabVIEW, MySQL

Table of Contents

1. Introduction

Quality inspection is an important part of product design, in verifying that it meets desired specifications, and validating that it fulfills its intended purpose. This verification and validation can be achieved through statistical analysis, ensuring that these specifications are continually met with each individual product build. With an automated system, this inspection can be streamlined for maximum efficiency. Using the LabVIEW platform, this application note will demonstrate an automated database management method for Microsoft Access using a MySQL interface.

2. Objective

The objective of this application note is to provide the reader with sufficient information on how to integrate database management in an automated setting. Specifically, this note will inform the reader how to add data to and retrieve tables from a Microsoft Access database, using a MySQL interface under the LabVIEW platform.

3. Resources

For the purposes of this application note, the reader should have the following resources available to them:

- Microsoft Access
- National Instruments LabVIEW
- LabVIEW Database Connectivity Toolkit

It is assumed that the reader has a basic knowledge of Microsoft Access operation, the LabVIEW platform, and MySQL syntax.

4. Implementation and Result

4.1. Adding a Recordset

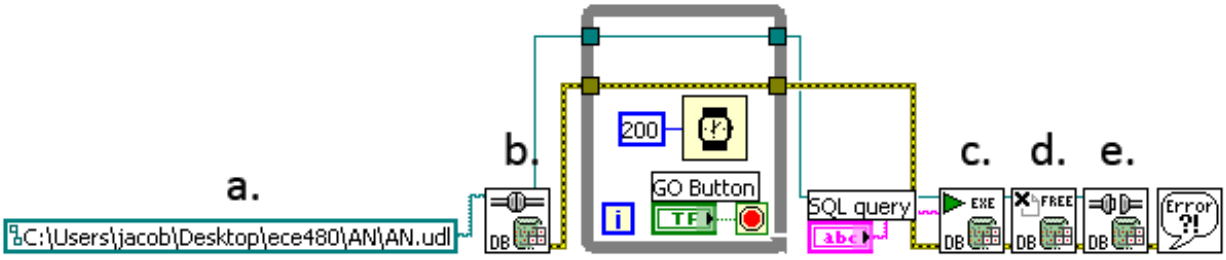

Figure 1. Add Recordset Example Block Diagram

Figure 1 shows an example block diagram for adding a recordset to a Microsoft Access database. The block diagram features the following key components:

a. Universal Data Link

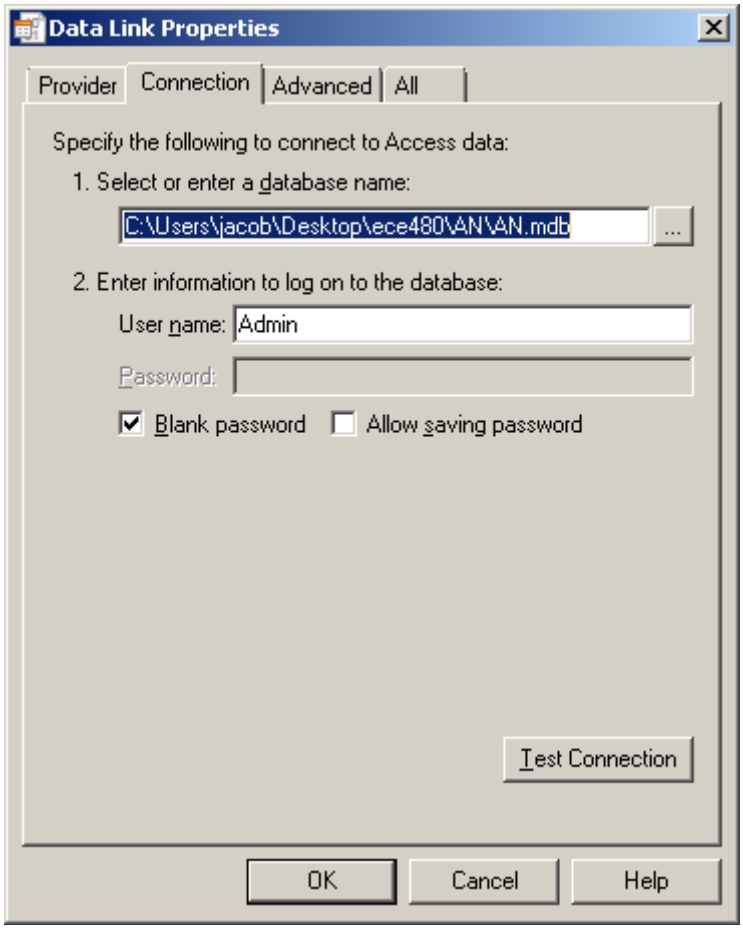

Figure 2. Universal Data Link Properties

The Universal Data Link (UDL) file specifies the type of connection to be established between the database and accessing application, as well as the location of the database itself. Various connection providers are available, based upon the type of data that is being accessed. For the purposes of this application note, the **Microsoft Jet 4.0 OLE DB Provider** is used. In Figure 2, the absolute path of the database to be accessed is specified; in this example, the database is located at **C:\Users\jacob\Desktop\ece480\AN\AN.mdb**.

b. 

DB Tools Open Connection.vi

The DB Tools Open Connection VI is part of the LabVIEW Database Connectivity Toolkit, which is a National Instruments-supported add-on for database connectivity in LabVIEW. This VI reads in the UDL file for database connection information, and passes out a connection reference to the database for use in other VIs.

c. 

DB Tools Execute Query.vi

The DB Tools Execute Query VI is also part of the LabVIEW Database Connectivity Toolkit. This VI reads in the connection reference, as well as a MySQL query. It executes the query, and passes out a recordset reference to the information in question.

d. 

DB Tools Free Object.vi

The DB Tools Free Object VI is also part of the LabVIEW Database Connectivity Toolkit. This VI destroys the recordset reference, as its purpose has been completed, and now allows later parts of the program to access it.

e. **DB Tools Close Connection.vi**

The DB Tools Close Connection VI is also part of the LabVIEW Database Connectivity Toolkit. This VI closes the connection between the database and LabVIEW, as its purpose has been completed, and now allows other applications to access it.

roster

lastname	firstname	pid	Add New Field
doe	jane	A12345678	
smith	john	A87654321	
*			

Figure 3. Example Database Table

Figure 3 shows an example database table in AN.mdb, for the purposes of this application note. The table is a class roster, with fields for last name, first name, and personal identification number (PID).

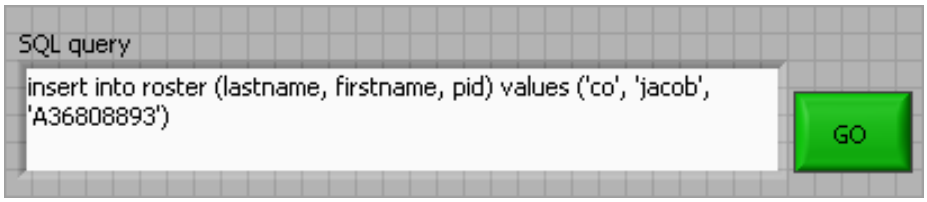

Figure 4. Add Recordset Example Front Panel

Figure 4 shows the corresponding front panel for the example block diagram in Figure 2 for adding a recordset. A MySQL query is provided in the text box, and the 'GO' button executes it. In this example the query **insert into roster (lastname, firstname, pid) values ('co', 'jacob', 'A36808893')** is provided. This will insert a recordset into the roster table, with values of 'co', 'jacob', and 'A36808893' for the fields lastname, firstname, and pid, respectively.

roster

lastname	firstname	pid	Add New Field
co	jacob	A36808893	
doe	jane	A12345678	
smith	john	A87654321	
*			

Figure 5. Example Database Table After Add Recordset

Figure 5 shows the results of the Add Recordset Example VI. A new recordset with the desired field values has been added to the table.

4.2. Retrieving a Table

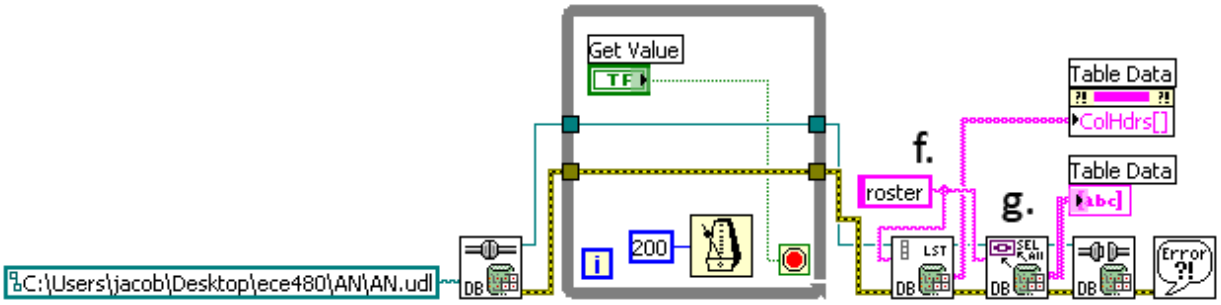

Figure 6. Retrieve Table Example Block Diagram

Figure 6 shows an example block diagram for retrieving a table from a Microsoft Access database. The block diagram features the following additional key components:

f. 

DB Tools List Columns.vi

The DB Tools List Columns VI is also part of the LabVIEW Database Connectivity Toolkit. This VI reads in the connection reference and the table name in question, and passes out the table field names.

g. DB Tools Select All Data.vi

The DB Tools Select All Data VI is also part of the LabVIEW Database Connectivity Toolkit. This VI reads in the connection reference and the table name in question, and passes out all the table data.

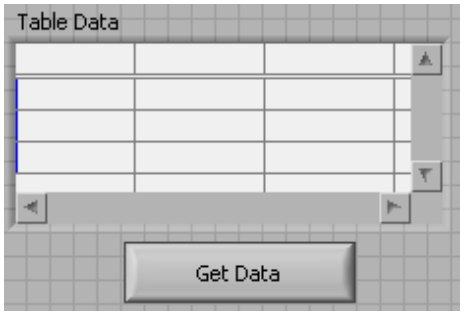

Figure 7. Retrieve Table Example Front Panel

Figure 7 shows the corresponding front panel for the example block diagram in Figure 6 for retrieving a table. Data is retrieved when the 'Get Data' button is pressed.

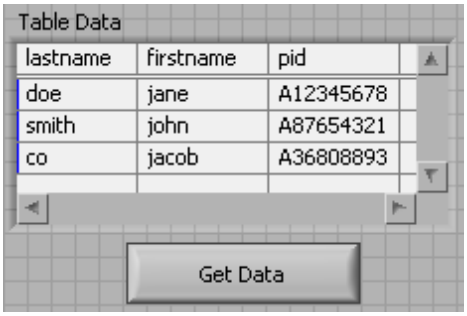

Figure 8. Retrieve Table Example Front Panel Execution

Figure 8 shows the table retrieval front panel after execution. The roster table has been retrieved, complete with field names and recordset data.

5. Conclusion

By being able to add recordsets and retrieve tables from Microsoft Access databases in LabVIEW, an automated system can be designed for use in verification and validation settings. ECE 480 Design Team 6 is utilizing such a system for their Automated Actuator Inspection Device (AAID), which inspects BorgWarner, Inc.'s fan clutch actuators for proper electrical operation. The device stores inspection results in a database, which can be looked at later to identify failure trends to improve future actuator builds.

6. References

[1] LabVIEW Database Connectivity Toolset User Manual.

http://www.ni.com/pdf/manuals/321525c.pdf

[2] MySQL 6.0 Reference Manual. http://dev.mysql.com/doc/refman/6.0/en/index.html