

2006-1304: INSTANT DATA GATHERING, PROCESSING, AND DISPLAY FOR INDUSTRY SPONSOR USING INTERFACED SOFTWARE PACKAGES

Tracey Windley, University of Idaho

Tracey Windley grew up in Moscow Idaho and started his education at the University of Idaho in 2002 for a BS Electrical Engineering. He is currently a senior emphasizing his education in microelectronics and semiconductor physics. He is currently at the MRCI at the Univeristy of Idaho doing research in the area of microelectronics.

Jasper Nance, University of Idaho

Kevin Buck, University of Idaho

Herbert Hess, University of Idaho

Herb Hess received the PhD degree from the University of Wisconsin-Madison in 1993. He served on the faculty of the United States Military Academy from 1983-1988. In 1993, he joined the University of Idaho, where he is Associate Professor of Electrical Engineering. He received the Best Paper Overall Award for the 1999 ASEE Annual Conference. His interests are in device and circuit aspects of power electronic energy converters.

Instant Data Gathering, Processing, and Display for Industry Sponsor Using Interfaced Software Packages

For several decades, a fundamental skill of the engineer has been the ability to program a computer. As software tools have advanced, the requirements of this skill have changed from code writing to interfacing several pieces of software at once. This paper describes a project utilizing several pieces of software for rapid data acquisition, storage, and presentation to industry sponsors, while addressing problems encountered in data acquisition.

Problems Encountered in Data Acquisition

Data acquisition has two main parts; the actual collection of raw data and the organization and storage of collected data. A problem that occurs often with data collection is lack of precision introduced in measurement. People are not perfect, but the nature of data acquisition requires precision. Often the same test must be performed hundreds of times. The data gathered must be repeatable to have value and meaning. For novice researchers, this is a situation where human error could easily prosper.

In addition to lack of precision in measurement is the problem of data collection speed. Data collection speed is a serious problem with novice researchers for two main reasons. One reason is simply due to inexperience with measurement devices and technique. The other reason is due to an overly cautious approach on the side of the researchers to make no measurement errors. This cautious approach, while helping to improve precision, is time costly. Often taking the extra time for measurement is not possible or cost effective. Measurement time is also an issue when large amounts of data need to be collected and there is limited man power

The other part of the data acquisition is the organization and storage of the data. Data stored “well” is data stored in a uniform format including all pertinent data surrounding the project, such as the date the data was recorded, who recorded it, what instruments were used to record it, and specific notes about what the data represents. This ensures that data is credible and easily understandable for future use.

However, with inexperienced researchers, often the only information recorded is the raw data excluding the pertinent information mentioned above. Therefore, the only record of this other information is the memory of the researcher. Often there is a significant amount of time between actual data acquisition, data processing, and presentation, especially in a university setting. Memory fades and what may have made sense two months ago may have no meaning now. Also the person processing the data or presenting it may not be the same person who recorded it. Even worse, the person who recorded it may no longer be part of the project. These time lapses and researcher discontinuities often result in situations where data could become lost, non-credible, or incomprehensible to other researchers. In university research this is a real problem due the high turnover rate of researchers and inexperience in data acquisition.

In summary, the three main problems encountered during data acquisition with inexperienced researchers are the slow speed of data collection, lack of precision in

measurement, and poor data organization and storage. Thus the goal of this project is to select low cost software to help overcome these problems in the university setting.

Solutions to Data Acquisition Problems

A solution to the problems of slow data collection speed and lack of precision is to introduce the idea of test automation. A computer is both quick and precise, and can be programmed to perform the data collection task. An experiment designed to keep human intervention at a minimum frees the researcher to multitask, minimizes human error in measurement, increases precision, and greatly increases data collection speed. A data collection process that would take days by hand could take minutes if automated by a computer. Automated test systems also facilitate repeatability in measurement for cases when tests need to be performed numerous times.

A solution to the problem of data organization and storage is to use a database. Figure 1 shows a simple representation of a database.

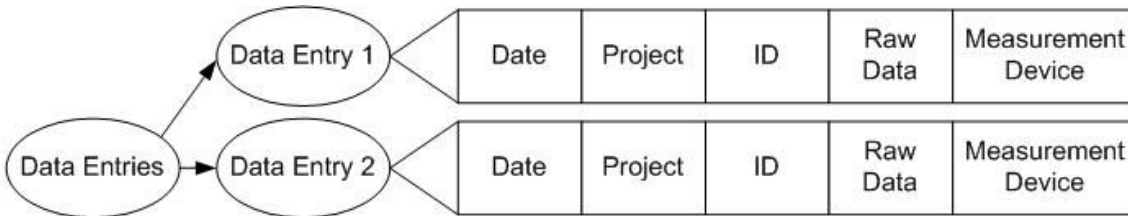


Table Containing Data Entries

Figure 1. Database Representation

A database is the equivalent of a virtual data spreadsheet. Each row in the data entry table represents a data entity, and all the fields in that row contain the pertinent data recorded during each formal test. Aside from the actual raw data, this information can include the researcher’s name, the date the data was recorded, the instruments used to record the data, or specific notes the researcher wanted to include about the test. The database provides a system where all this information can be stored in a single entity that can be searched and recalled for future use.

There are several software packages that can help to implement the chosen solutions to the problems. However, not many combine both the power of automation with database communication. Fortunately, there exists a software package that combines instrument automation with database communication. This software package is LabVIEW®.

LabVIEW® is a powerful and flexible instrumentation and analysis software system. It uses the graphical programming language G to create programs called virtual instruments (VIs) in a flowchart-like form called a block diagram. Figure 2 shows an example of simple code from the LabVIEW® environment used to write data to a database. The code is comprised of four functions developed by National Instruments for database connectivity. The procession of data through the code is similar to a system block diagram. In the case of Figure 2 the first block opens a connection to the database called “Database” and passes the connection reference to the next block represented by the green wire. The next block requests column names from the table

called “Table” in the database. That information is passed to next block responsible for inserting data into the table. The large pink wire flowing into the top of the block represents the data to be inserted in the table. This data can be the result of some measurement or project information. The last function block is responsible for closing the database connection.

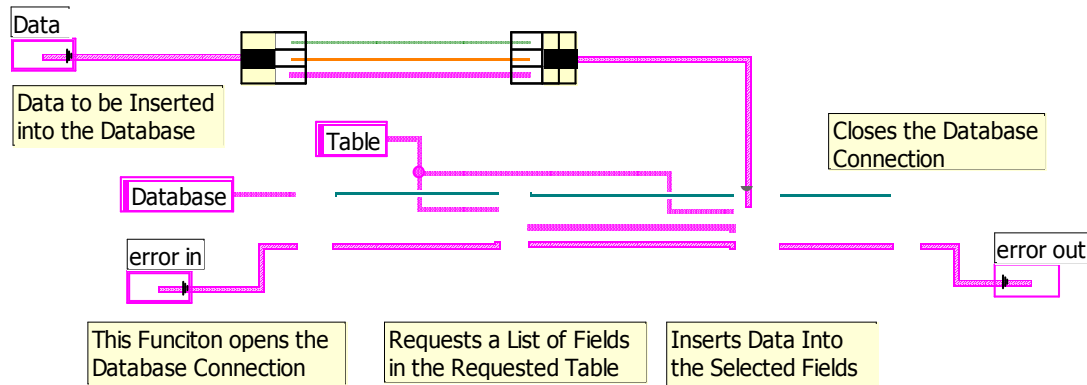


Figure 2. Example of LabVIEW® code writing data to a database

The LabVIEW® software package combines the power of instrument automation and data storage into a package that is visually appealing, easy to understand, and easy to use. The visual coding medium lends itself well to inexperienced coders and students. The modular nature of the code language has led to the development of a large library of function blocks and instrument drivers developed by National Instruments and third party sources suitable for almost any computer-oriented task or measurement instrument. National Instruments also has a large selection of programmable controllers capable of controlling most instruments with a GPIB or RS232 interface.

These qualifications make the LabVIEW® software package a good choice for the data gathering system. Figure 3 shows the proposed data gathering system for the LabVIEW® part of the project.

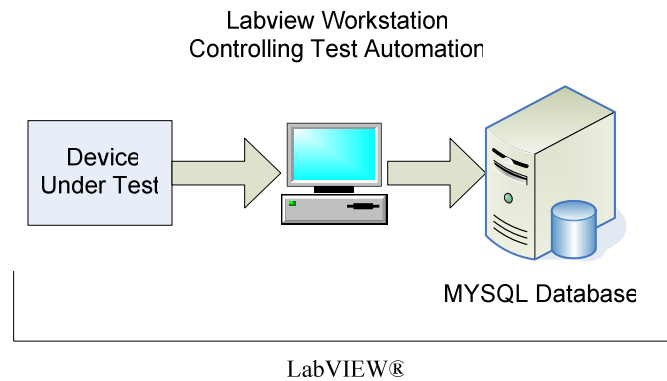


Figure 3. Data Gathering System Diagram

In Figure 3 the arrows represent data flow. The LabVIEW® workstation handles the test automation and data collection. Upon completion of the test, the LabVIEW® workstation uploads the data to the MySQL Database. LabVIEW® can work with most database drivers, and is especially good at working with Microsoft databases. Reasons for the selection of the MySQL database driver are explained in the next section.

Moving Information from the Database to the User

Once the data is in the database it is necessary to make that information available to the project sponsor. LabVIEW® is capable of providing an environment suitable for data extraction from the database. However, LabVIEW® is not as widely used as other software packages for this application. Therefore, not many project sponsors will be able to use LabVIEW® to view data in the database, especially if the database is located a great distance away.

A webpage could be used to interface to the database. This way project sponsors can view data using a simple web browser instead of the LabVIEW® interface. Also the data can be viewed from anywhere in the world with an Internet connection.

However, due to the complex nature of most databases, a simple webpage may not be enough to view the data efficiently. It would require an inordinate number of static HTML pages to view the data organized in all possible ways. This requires too much development time and manpower to be effective.

PHP is an answer to this problem. PHP script can be understood as a sort of “smart” HTML. PHP can split out HTML code based on input parameters. HTML code forms a sort of skeleton outline for the main page and fills in the blanks with information selected by the user from the database. This way any number of HTML pages can be generated from just one skeleton page. An example of this could be a personnel page designed to give information about the researchers. This information could include contact info, personal information, and a picture. Instead of making HTML pages for each researcher, one skeleton page with PHP script can be made. This page can query the database based on input commands from the website and can fill in the variable spaces in the HTML skeleton code with the correct researcher information. This is the approach used to design the website interface to the database.

PHP to MySQL Integration and Website Interface

The other half of the project lies in the PHP to MySQL (Structured Query Language) integration and web site navigational interface. When a project is undertaken in a university setting it is important to realize the high turnover rate of students. Projects using difficult or archaic methods and coding languages should be avoided due to the inability of an inexperienced student to simply pick up where the previous student has left off. This problem has led to use and incorporation of Open-source tools for this part of the project. Open-source tools and computer programming languages rarely stop evolving and documentation and support are free. PHP and MySQL are both popular Open-source interfaces, and are great solutions for implementation in an educational environment.

Figure 4 shows the chosen overall system integration. The system will have LabVIEW® controlling test automation and data insertion into the MySQL database, while the PHP website will control the data presentation and security to the industry sponsor.

Data Gathering System Diagram

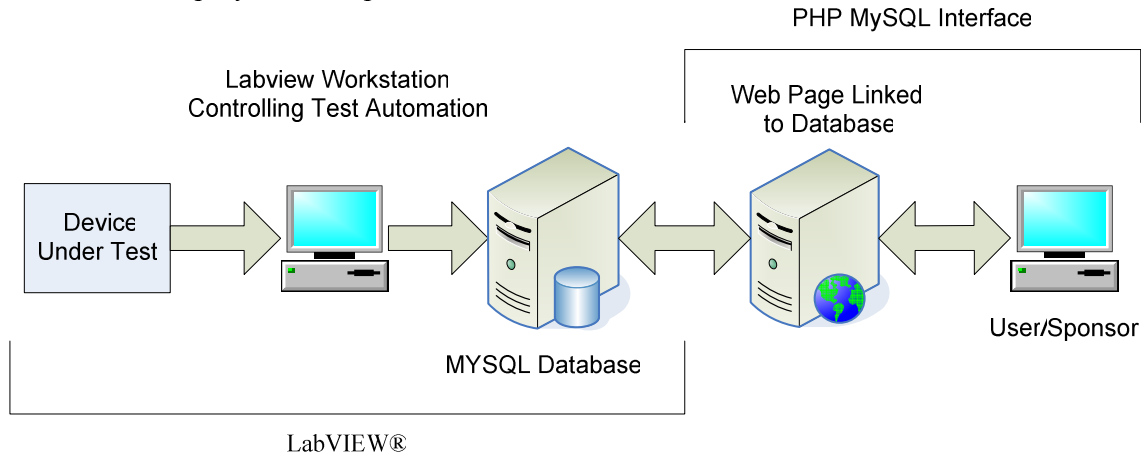


Figure 4. Data Gathering System using LabVIEW® and the PHP MySQL Interface

Using a PHP website, we can dynamically create and modify various presentations of data to a wide audience as explained in the previous section. Website security is enhanced by using PHP session variables, with the authentication information stored in an MD5 hashed MySQL table. This means that each person wishing to access the website must have valid authentication information on file in the database. Only the administrator account may add or remove users. All password hashing is done on the server-side for increased security. Individual projects may also have security permissions, giving lists of users who may have access to the data filed under these projects. For these projects, no anonymous access is granted.

MySQL is a secure and open-source SQL implementation that allows for very fast database searching and provides a backbone to the website from which all data is retrieved. Using a database like this means that often times the only parameters that need passed from one webpage to another is a row id number that references a specific database entry.

PHP Website Features

The overall structure of the website follows a typical blog-like layout, where a user can make journal-style entries that consist of a graph (or graphs) as well as text that is important to the graph. Entries can be viewed sorted by date, project, user, or search words assuming that the viewing user has permissions to the data in the entry. Users select the data to make graphs from a list of individual traces that can be imported from a CSV file, or entered directly from an instrument through the LabVIEW interface. The graphs are limited to a common X-axis and up to two Y-axes.

The entries and graphs are not simply stored as image files; rather, they contain a list of traces to include as well as formatting data. This data is then sent through an open-source PHP-based graph generation module called jgraph. A cached version of the graph is stored as an image file to relieve server load, however, a graph can be dynamically explored or recreated with different parameters at any time, for example, zooming in or out or the application of data point smoothing. The graph can also be exported as a document file if it needs to be used in a publication, for example.

The use of server-side scripting makes the creation of stylistic enhancements quick and easy. For example, user profiles and avatars (user icons that appear next to entries and comments) make a website look professional and allow end-users of the data, perhaps industry representative who wish to view the data from a project, to have an inside look at a group of researchers. Placing a face on a set of measurements can give perspective into a student based research group.

Conclusions

With all of these tools together in a single data acquisition system it becomes possible to rapidly and automatically collect data, upload the data into the database, and make that data easily available to select parties all over the world in a matter of minutes. This system is low cost and addresses the main problems encountered in data acquisition. The software packages chosen help solve these problems and make data acquisition easier for new students and novice researchers.

References:

1. Bishop, Robert, "Learning with LabVIEW 7 Express", Pearson Education 2004, Preface pg xiii-xix.
2. Bishop, Robert, "LabVIEW 6i Student Edition", Prentice Hall 2001, pg. 30.
3. MySQL Information. <http://dev.mysql.com>