Databases: The Perfect Complement to PLCs

STEVE HECHTMAN, President, Inductive Automation

By STEVE HECHTMAN, President, Inductive Automation

PLCs? OK, you've tackled PLCs and now you can program 'em with one hand behind your back. So what's next? What's the next logical challenge? Think SQL and relational databases. Why? You'd be amazed the similarity. It's the next logical progression.

You might ask how it is they're even related. For one thing, relational databases can sort of be an extension of PLC memory. Live values can be mirrored there bi-directionally. Historical values and events can be recorded there as well. But operators and managers can interact with them too. It's been over twenty years of working, living, breathing and thinking PLCs, but over the last six years I've delved heavily into SQL and learned a lot about relational databases. I've discovered that working with SQL is remarkably similar to working with PLCs and ladder logic.

SQL has four basic commands and about a hundred different modifiers that can be applied to each. These can be applied in various ways to achieve all types of results. Here's an example. Imagine effluent from a wastewater plant with its flow, PH and other things being monitored and logged. That's what you typically see.

But now let's associate other things with these, such as, discrete lab results, the name of the persons who did the lab work, the lab equipment IDs and calibration expiration dates, who was on shift at the time and the shift just prior, what their certification levels were, what chemicals where added and when, who the chemical suppliers were, how long the chemicals sat before use, and so forth ad infinitum. All of this becomes relational data, meaning that if it's arranged properly in tables you can run SQL queries to obtain all types of interesting results. You might get insight into the most likely conditions which could result in an improper discharge so it can be prevented in the future.

In my explorations of SQL, I found myself looking at the layout of my tables and evaluating the pros and cons of each layout. I massaged them, turned them on their side, upside-down, and finally ended up with the most appropriate arrangement for my application. And similar to PLC programming, I explored innumerable what-if scenarios. I was struck by the amazing similarity in my approach to developing

solutions for PLCs. This has been a lot of fun — in fact exhilarating — just like PLCs used to be. It's the next logical progression you know.

SQL is a high level language that isn't very hard to learn and you can be very clever with it. I prefer to think of it as a natural extension to my PLC programming skills. Now that you have the machinery running, what did it do? Furthermore, relational databases and SQL pull people and processes together. Machines don't run alone. They're merely part of a containing process and that process was devised by people. SQL and relational databases form the bridge to integrate processes, machinery and people together.

I don't believe a COTS (commercial-off-the-shelf) package can do it any more than you could offer a COTS palletizer program and have it be of any use. It just doesn't work that way. Every machine is different. And every business process is different. That's where the SQL comes in. It has to duplicate or augment existing process flows and these are intimately connected to the machinery. And that's why the PLC programmer is best suited to implement solutions involving PLCs and relational databases.

So where do you start? I would suggest picking up a book at the bookstore like one of those dummies books. Then download and install the open-source MySQL database server along with the MySQL Administrator and Query Browser. It only takes a few minutes to install and then start playing. You can read about a LEFT JOIN or INNER JOIN but typing one in and observing the results is worth about 1,000 words. At the end of an evening you'll probably be very excited with all of your new found knowledge and be thinking of endless ways to employ it in your own field of practice. Happy SQLing!

For more information, please visit www.inductiveautomation.com *[1].*

Source URL (retrieved on *07/22/2014 - 12:12pm*):
http://www.chem.info/blogs/2010/10/databases-perfect-complement-plcs

Links:
[1] http://www.inductiveautomation.com/