Cogent Real-Time Systems Inc

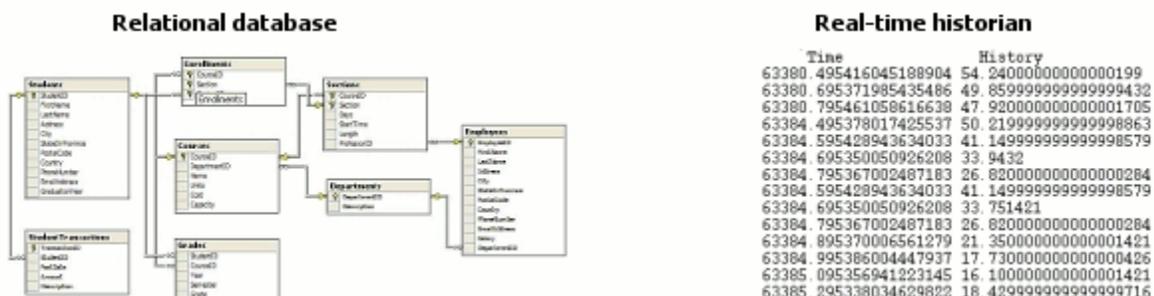# Relational Database or Real-Time Historian for Logging Process Data?

by Robert McIlvride, Cogent Real-Time Systems Inc.

Quite a few years ago, while living in a part of the world where personal computers were a relatively new phenomenon, I happened to be in an office watching a secretary busily typing away at her new PC. She was thrilled to have such powerful tool to use. "Look!" she said excitedly. "Now I can write and correct my work so easily!" I looked at the screen, and had to smile. She was composing an entire business letter within a single cell of an Excel spreadsheet.

What determines the right tool for the job? For that secretary, the right tool was the one that was available and that she knew how to use. What's the right tool for logging data from a process control application? Sometimes a CSV file is all that is needed. Sometimes Excel will do. More often, though, engineers and system integrators will use either a relational database or a real-time historian to store permanent records of process data.

Relational databases have the advantage of availability and familiarity. SQL Server, MySQL, Oracle, or any other relational database, including Microsoft Access, are usually already installed at the company. They offer a common interface, ODBC, and the IT department is familiar with them. It's no surprise that relational databases are being used for logging process data, particularly when the requests for data come from management personnel familiar with this kind of database.

But a relational database may not be the ideal choice for all process control applications. Designed for the needs of corporations, businesses, and banks to store transactional data, relational databases are optimized for analyzing complex relationships between data. These databases can afford to focus processing power on these relationships because the data itself gets updated relatively infrequently, usually in terms of seconds, minutes, or hours. While analytical power is good for business applications, process control applications typically don't need it. What they do need is speed.

A real-time historian, on the other hand, is like a flight-recorder for process data. Rather than relational, it is a temporal database, storing its records in a flat file, consisting of simply the name, value, quality, and timestamp for a data point. The historian is designed for speed of

storage and retrieval of data, and can typically process millions of transactions per second. This kind of performance is valuable for processes with variables that may change many times per second, and where capturing every event over the course of each eight-hour shift is vital.

Despite the performance advantages of a real-time historian, some companies opt for using relational databases for logging process data. This is completely understandable, since those are the tools that company IT staff and upper management are most familiar with. But there are three important reasons why this approach may not be sufficient.

1. A real-time historian logs every data change for a point, even when the values change rapidly. Using highly efficient storage algorithms, the complete data set can be stored for long time periods. A relational database, in contrast, may need to drop some or most of the data as it is being logged, because it is not optimized for storing data at very high transaction rates. Unfortunately, the data is dropped regardless of its importance. So you might end up logging routine changes and throwing away the unusual events that could lead to alarm conditions. In addition to detecting every change, big or small, the high volume capabilities of a real-time historian are useful for detecting subtle trends that may only appear over months or years.

2. A strong advantage of a real-time historian is its native ability to process time-based queries. For example, you might need the standard deviation of a point that changes on average 25 times per second, in 10-second windows for the last two minutes. A good historian will provide an easy way to submit such a query, and will return the results quickly, with a minimum drain on system resources. Built-in query functions typically allow you to select any time period, from a few seconds to weeks or more, and retrieve averages, percentages of good and bad quality, time correlations, regressions, standard deviations, and so on. All of this may be possible through SQL queries on a relational database, but through much more programming effort and greater system resource use.

3. The two above advantages of a real-time historian may perhaps best be appreciated when working with live trend displays. Calculating and displaying a moving line that updates several times per second requires not just the ability to log all the data points in real time, but also to re-emit them quickly and repeatedly for the moving display. And if a user wants to scroll backwards and forwards through the data set as it is being logged, the historian has to be able to manage rapid, continuous queries to the data set. This kind of task is difficult for an off-the-shelf relational database, unless the screen refresh rate is annoyingly slow.

Even with these points in mind, there are many applications for which logging process data in a relational database works just fine. In fact, sometimes logging to a CSV file is sufficient. To be fair, these are really not the same level of technology mis-match as writing a complete business letter in one cell of a spreadsheet. The well-informed system integrator or engineer will understand the values of each approach, look at the needs of the project and resources available, and employ the right tool for the job.

Cogent Real-Time Systems Inc

□   □   □

Founded in 1995, Cogent Real-Time Systems is the leader in real-time data integration between Windows, Linux and QNX systems. Customers include the Bank of Canada, Cadbury Chocolate and the European Space Agency. Cogent leverages its experience in real-time data communications to provide the next generation of OPC products. For more information, please contact Cogent at info@cogent.ca or visit our web site at www.opcdatahub.com. You can also call us at +1 (905) 702 7851.