CHOOSING THE BEST INDUSTRIAL DATA COLLECTION METHOD IN A CROWDED MARKET PLACE

Introduction:

Most companies today are not thinking, *"Should I be collecting data?"* They are thinking *"What data should I collect and how do I want the data displayed for my team?"* After asking those questions many find themselves searching for solutions in an overly crowded marketplace of industrial data collection. They begin to ask *"What is the difference between historians and relational databases such as SQL?" "Should I go with a local or plant wide system?" "What are the types of reporting software?"*

I am going to cover these questions, and try to guide you to what best fits your company's project needs.

Historian - Relational Database:

Historian databases collect massive amounts of data in collection intervals of 500 ms or less. They are constantly speed reading a value and recording changes in that value. Many Historians use a proprietary collection and compression algorithms, with some adjustment available to the user. What do these fancy algorithms do? That depends on the Historian that was chosen. But for most, the algorithm decides what data changes will be logged to the Historian database file. For example, the user could have settings set to only record changes of the RPM of a motor based on .25 intervals, so if the motor fluctuates for 2 hours starting at 24.01 rpm to 24.23 rpm, the value for that time period would be logged 24.01. All of the data would be captured in real-time, but it would be discarded and compressed to that number to save space. This could be extremely useful if you're collecting massive amounts of data on a single point at short time intervals and don't want to fill up the historian hard drive. You are losing data when you do this, but the data has been defined as useless by the engineer; and therefore is discarded to save space.

Historians are not good for storing batch information, material tracking, and cannot be used to store recipe information for equipment download. Historians are good for tracking thousands of fast data changes in equipment and sensors over a short time span.

Relational databases organize data into tables set up by the user. In each table, there is a set of rows and columns. Most tables have a Primary Key that makes accessing that row easy for outside software to read from the database. In the industrial world, the Primary Key is usually a recipe ID, material ID or batch number. Relational databases are great for storing recipe information, keeping track of materials usage, and storing end-of-batch information. However, these databases do not keep track of tags in an OPC server like a historian would. The user/software must initiate the data update/store from an outside software package or relational database management tool. Most information technology departments have become highly proficient in maintaining relational databases and integrating multiple software packages into them. Relational databases are not conducive for constant data logging like historians. They do not have the capability to accept data entry at the pace of a historian and the database would become bloated with useless data without the ability of compression.

If you have the capital, you should leverage both systems. It would give you a wealth of information to source for cross-functional reports.

Local - Plant-wide Data Collection:

Local systems are placed in the area of the machine you are collecting data. It is software placed on a nearby PC or a module mounted in the PLC rack. They are usually maintained by the local users in the area. This gives the user greater control and flexibility on what information they want to capture/use. This works great for monitoring 1- 3 pieces of equipment in a localized area. Local systems also take away any cost that you might have for the IT Department to maintain the system. A great local historian option on the market today is Kepware local historian plugin for Kepserver. This plugin stores the historian file on the local C drive. The information can be accessed using software that utilizes OPCHDA protocol. Another local option is Kepware ODBC data logger plugin, which has the ability to save information to any ODBC database - including MS Access. Rockwell

Automation also makes a local machine edition historian, which is a card that is placed in an empty PLC rack slot.

Plant-wide systems are great if you want to collect data from multiple lines and compare performance. Plant-wide systems are normally deployed on a server that is connected to the plant's network. There are many historian packages that provided the plant-wide option. For instance, Rockwell Automation offers a total package in which the reporting software is bundled with their historian. OSI Soft, GE, and Canary Labs also offer great plant-wide systems.

If you want to monitor data on a single device, then choosing the Local option is the best way to go. If you are looking at expanding beyond one device, I would suggest the Plant-wide option because it will give you scalability.

Reporting Software:

There are many types of data reporting software on the market today. I categorize them into three different groups which are: trending, reporting, and mathematical analysis. For example, Rockwell's Vantage Point Software can do all three. If using a relational database, some may choose to develop their own custom system that fits their specific data reporting needs. When selecting a reporting software, make sure that you are comfortable on how it displays the data and its ease of use. It's always a good idea to try software on a trial basis so that you can make sure it fits your reporting needs.

Canary Labs, Rockwell Automation, GE, Dream Report, XLReporter, and MATLAB are good options to look at.

Conclusion: As you can see, there are many options available when collecting plant floor data. If you have any questions, or would like to discuss designing a system, please email Apex Controls Specialists Industrial Data Management Department.

Written By: Matt West (Information Systems Integrator)

mwest@apex-engineering.com