Data Mining – Persistent OPC Variables

APPLICATIONS NOTE

October 2013

Description: Guide to using the Data Mining OPC Server to store persistent OPC variables.
OS Requirement: Win 2000, XP Pro, Server 2003, Vista, Server 2008, Windows 7
General Requirement: A Relational Database (SQL server, MSDE, Access, Oracle, etc); GENESIS32 or BizViz.

Introduction

The Data Mining OPC Server is designed to query a database and expose the data there as OPC tags, available to any OPC client. Because it has the ability to both read and write to these tags, you can also use the Data Mining OPC Server to create OPC "variables" that will persist even after the machine shuts down. This is an advantage that the Data Mining OPC Server has over our Simulator OPC Server or DataWorX32.

To create such variables, you will need to use an existing database or create a new one, add a table or two to store your data, and then configure the Data Mining OPC Server to make this data available as an OPC tag. The example below uses Access for this database, but you can also use other databases, such as SQL Server or Oracle.

If you are very familiar with relational databases, you may want to create more complicated databases structures. This application note provides an example of the simplest way to implement persistent OPC variables.

Creating the Database Tables

The first part of this process is to create a database where our data will be stored. Since the data is in a database, it will not be lost, in the event that the OPC server or the machine shuts down.

1. Create a new Access database or open an existing one. For this example, we are creating a new database and naming it "Variables.mdb."

   NOTE: It is possible to use the Data Mining OPC Server's own configuration database for this, but we recommend you create a new database to prevent confusion later.

2. Create a new table in the database (also called "Variables" for our example) and add a few columns. These columns will represent the different variables you want to store. In this example, we'll create three columns like so:
   - MachineName, type Text
   - BatchNumber, type Number (Long Integer)
   - FlowTotal, type Number (Double)
3. Edit your table and put a value in the first row of each column. For this example, we will only be using one row, and our values will be:
   - MachineName: MyMachine
   - BatchNumber: 50487
   - FlowTotal: 10.99
4. Close your database. If you want to add more variables later, edit your database again and add more columns to your table. You can add more tables as well if you like.

Configuring the Data Mining OPC Server

1. Start the Data Mining Configurator by going to Start → Programs → ICONICS Tools → Data Mining Configurator.
2. Right-click on "Databases" in the tree-control and select New → Database Connection. Give it a name and configure it to point to your Variables.mdb database.

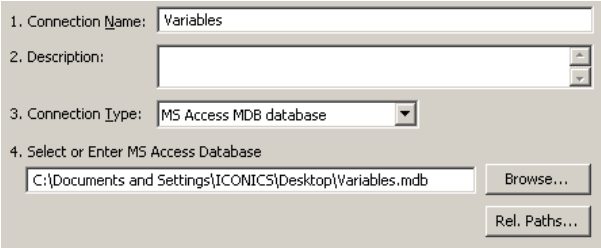

Figure 1 - Database Connection

3. Expand the new Database Connection, Right-click on the Data Sources folder and select New → Datasource. Give it a name, browse for your Variables table, and check the "Enable Writes" checkbox.

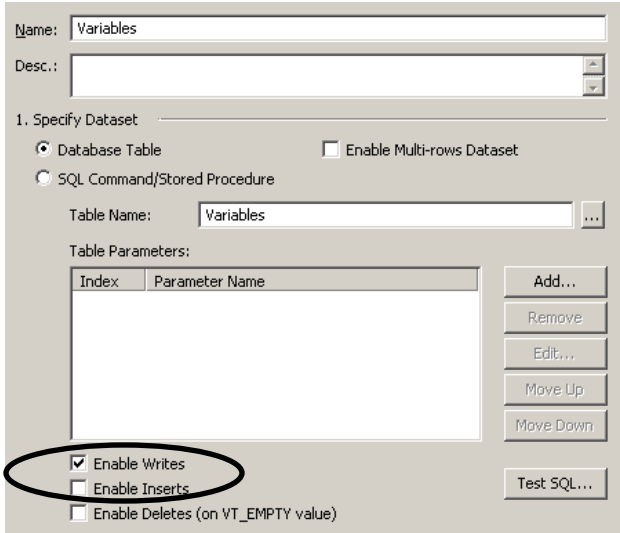

Figure 2 - Datasource

5. Right-click on the new data source and select New → Data Item. Give it a name (usually the same as the column), and pick one of your columns.
6. Repeat step 5 for each of the columns you created. These are now OPC tags ready for use.

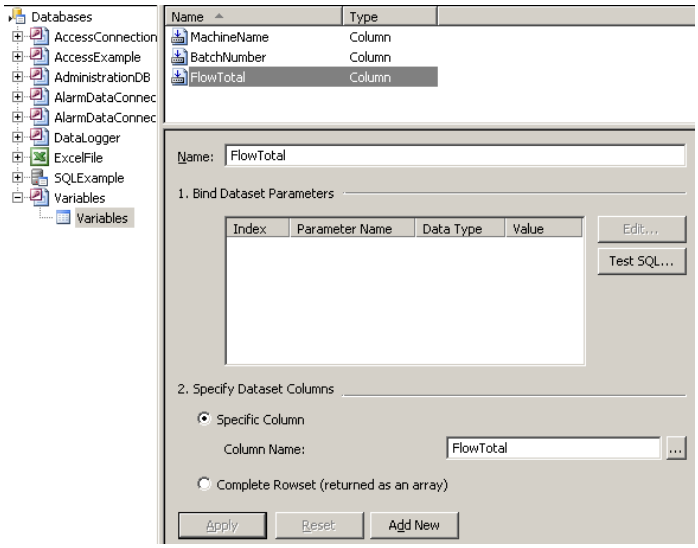

Figure 3 - Data Items

Using your Variables

1. Open OPC DataSpy by going to Start → Programs → ICONICS Tools → OPC DataSpy.
2. Hit the "Monitor Item" button in the toolbar.
3. Browse for your new OPC tags under the ICONICS.DatabaseOPCServer.3 server.

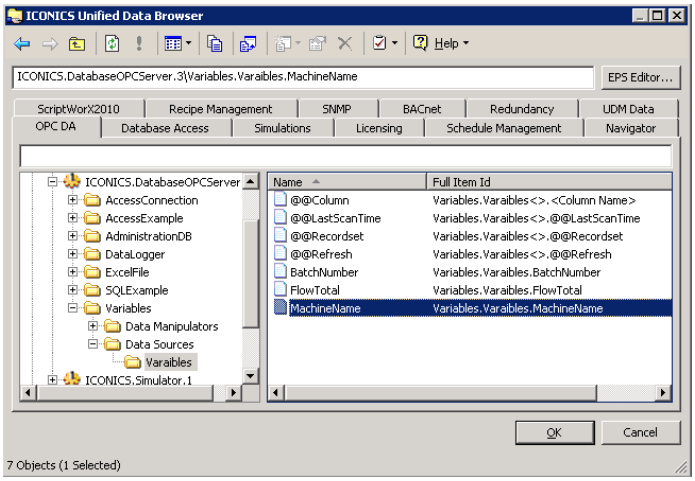

Figure 4 - Browsing for Variables

4. Select the Data Monitor item in the tree (do not expand it). You should see your values in the right-hand pane.

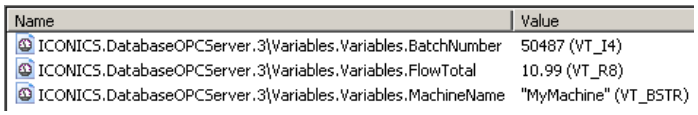

Figure 5 - Viewing the Data

5. Right-click on one of the tags, select Write and enter a new value. The new value will be reflected in DataSpy, and if you open your Variables.mdb file you will see the new value in there as well. These values will remain, even after the machine has been shut down.