



DataLogger Tips and Tricks

After you verify that your data source is supported by DataLogger (by visiting the [DataLogger Product page](#) at Kepware.com):

1. Logging Data Consumes Disk Space

The amount of data you are logging and the frequency at which it is logged can consume hard disk space at a very quick pace. For example, in testing we used a wide data format to log 1000 points of data every 10 milliseconds. It took about 3 days to consume 5GB of disc space.

If you need to log data at this rate then there is no way to avoid this. However, most applications will require much slower rates. Be aware of how your logging choices may impact your disk space consumption.

2. Plan your design

The best way to plan your DataLogger project is by determining what you want to do with the data you are logging. Knowing what you are going to do allows you to create your data tables in the most optimized format. You can then create your DSN(s) and your DataLogger project to populate these tables accordingly. For instance, if your goal is to log product data by shift in order to generate a weekly production report, you might create a table with records consisting of shift #, up time, down time, parts produced, parts rejected, and a date/time stamp.

Here are two database design websites we recommend:

Five Simple Database Design Tips:

<http://articles.techrepublic.com.com/5100-22-1045788.html>

Tips on Database Design and Use:

<http://www.techsoup.org/learningcenter/databases/page5093.cfm>

3. Optimizing Data Logging

It is possible to tailor the threading model of DataLogger to suit the user's needs.

The DataLogger interprets each DSN as a queue to create a new logging thread. For example, if LogGroup01 uses a DSN named 'MySQL_ONE' and LogGroup02 uses a DSN named 'MySQL_TWO' then the DataLogger will generate two distinct logging threads, even though both MySQL_ONE and MySQL_TWO may point to the same database.

If LogGroup01 and LogGroup02 both use MySQL_ONE, then they will take isochronous (i.e. equal opportunity) turns processing data.

This scalability allows the user to log tens of thousands of points simultaneously (we created two log groups, each of which was logging to distinct DSNs pointing to the same MySQL database and was logging approx 50,000 columns per second).

4. Logging to Microsoft Excel

Logging to existing tables in Excel files is supported by DataLogger via the Microsoft Access ODBC driver and the Link Tables feature in Access.

Note: DataLogger cannot be used to create new tables in Excel.

To log to an existing table in an Excel file:

1. Create a new Access DSN that points to the Access database where the Link Tables process will be performed for the Excel spreadsheet that you want to log to. (In one case, using a pre-existing Access DSN for the same database met with poor results.)
2. Open the associated Access database and select File|Get External Data|Link Tables. Browse and select the Excel file to link.

*Project Design Tip: Choose descriptive names for table headers in Excel. This will make it easier to map the OPC server items to the table headers (the mapping is done in the Map Item Fields dialog).

5. Data Source Handling of Timestamp and Date/Time

Each data source is going to handle the date and time a little differently and in some cases the formatting will be very different.

- Microsoft Access Timestamp

Access stores date/time internally as a double precision floating point number: xxxx.yyyy. Since the time value at midnight is “00:00” the 'yyyy' part is not displayed by the Access user interface.

- Timestamp and Date/Time differences in Microsoft SQL.

TIMESTAMP is interpreted by MS-SQL internally as a binary blob used to process image information. It has nothing to do with OPC timestamps, or dates, or time information. Users should always use SQL_DATE or SQL_DATETIME to log OPC timestamp information.

DataLogger will either log data to an existing database table that you have created, or you can choose to have DataLogger automatically create database tables. When DataLogger creates tables automatically, it “suggests” a set of column names and SQL data types that are compatible with the database you have selected. When DataLogger detects a request to create an SQL_TIMESTAMP column in an MS-SQL database, DataLogger posts an event log warning and overrides the column type, substituting SQL_DATETIME instead.

6. Error Handling – Database full

Each data source will have a slightly different error response when the database is full. Typically, KEPServerEX will post an event log message indicating that the RecordSet could not be queried. It is recommended that you monitor the “_DataLogger.<Logger Group Name>._Error” tag from your OPC client application to know an error has occurred. If the tag’s value changes from zero to one (1), this indicates an error state.