Hyper Historian
SQL Query Engine Quick Start

APPLICATION NOTE

August 2014

Description: Quick start document to use the SQL query engine in Hyper Historian.
General Requirement: Microsoft SQL Server installed.

Introduction

Hyper Historian logs data to a proprietary database. While you can use the TrendWorX64 and TrendWorX32 Viewers to see and edit your logged data in a graphical format, you may wish to create reports for the logged data or edit data in bulk. For this reason, Hyper Historian comes with a SQL Query Engine that allows you to use common SQL Data Manipulation Language (DML) queries to retrieve and edit data.

Hyper Historian Linked Server

First, open SQL Server Management Studio to see the providers, which are automatically installed and configured together with Hyper Historian.

1. Open SQL Server Management Studio by going to Start → Programs → Microsoft SQL Server 2005/2008/2012 → SQL Server Management Studio.
2. When asked, fill in credentials to connect to your local SQL server.
3. In the Object Explorer on the left, expand **Server Objects**.
4. Double-click on Linked Servers to expand it. You should now see a linked server called HH2, which stands for Hyper Historian.
5. Expand the **Providers** folder. You will discover a provider called ICONICS.HHOleDbProvider, which is necessary for your queries to work.

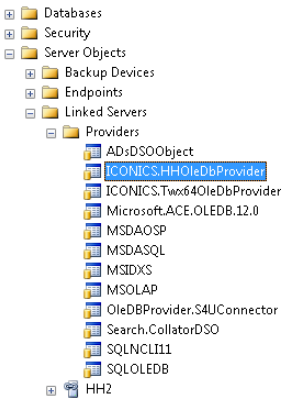


Figure 1 - Hyper Historian Provider

Building SQL Queries

In this example, you will use a simple SQL command that reads the list of all tags available in one Hyper Historian Logging Group. On top of that, you will create another simple query to retrieve logged data related to the selected tag.

You'll use the default samples Logging Group:

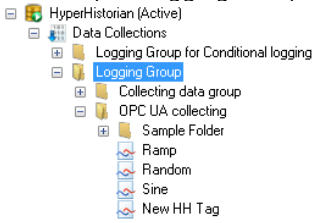
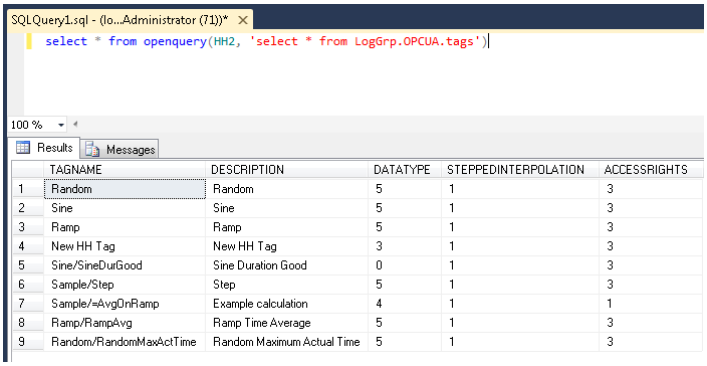


Figure 2 - Logging Group

1. Click on the New Query button, located in the standard Toolbar of Microsoft SQL Server Management Studio.
2. In the SQL Editor that opens, write the following query:

```
select * from openquery(HH2, 'select * from LogGrp.OPCUA.tags')
```

3. Click the Execute button in the SQL Editor Toolbar. You will see a result for the query similar to Figure 3.



	TAGNAME	DESCRIPTION	DATATYPE	STEPPEDINTERPOLATION	ACCESSRIGHTS
1	Random	Random	5	1	3
2	Sine	Sine	5	1	3
3	Ramp	Ramp	5	1	3
4	New HH Tag	New HH Tag	3	1	3
5	Sine/SineDurGood	Sine Duration Good	0	1	3
6	Sample/Step	Step	5	1	3
7	Sample/=AvgOnRamp	Example calculation	4	1	1
8	Ramp/RampAvg	Ramp Time Average	5	1	3
9	Random/RandomMaxActTime	Random Maximum Actual Time	5	1	3

Figure 3 - Getting Tag Names

NOTE: If you get no results for the query, you probably have no logged data. Open Hyper Historian (with the default configuration) and go into runtime mode. Then repeat step 3.

4. Now select one of the tag names to query. On the next line of the query editor add the following query, which will list all values for the specified tag:

```
select * from openquery(HH2, 'select * from LogGrp.OPCUA.rawdata where tagname = ''Ramp''')
```

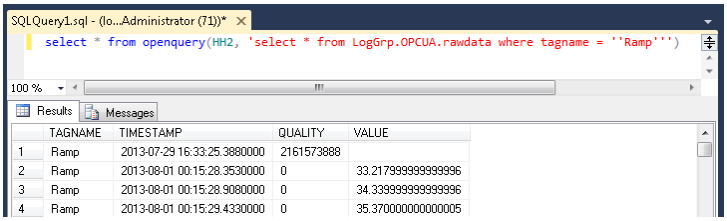


Figure 4 - Getting Logged Data

SQL Clients – MS Excel

The Hyper Historian SQL Query Engine can be used in other containers.

1. Open MS Excel by going to Start → Programs → Microsoft Office → Microsoft Office Excel 2010.
2. In the Data tab, click the From Other Sources button.
3. Select the From Data Connection Wizard option.

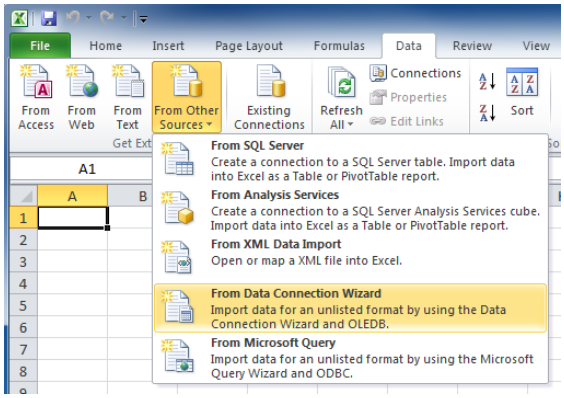


Figure 5 - Creating New Data Connection

4. Choose the Other/Advanced data source, and click Next.
5. Select ICONICS HyperHistorian SQL Query Engine, and click Next.

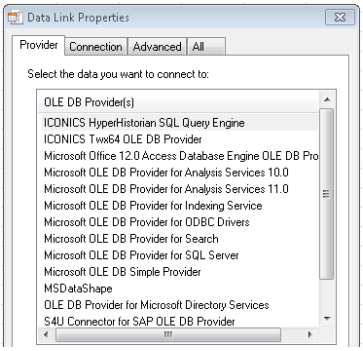


Figure 6 - Selecting Hyper Historian SQL Query Engine

6. On the Connection properties table, select the Catalog and Schema name, which correspond to the Hyper Historian logging group names. You should be able to leave the HyperHistorian URL box empty.

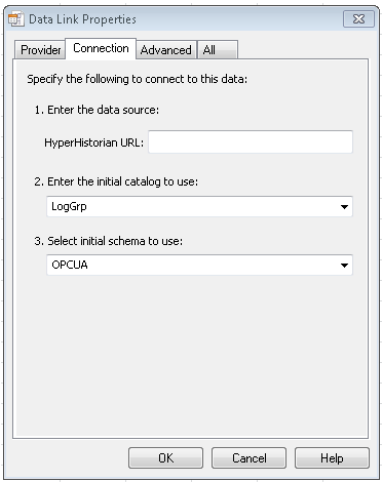


Figure 7 - Selecting Logging Group Name

NOTE: If you get an error message after clicking OK, check to see if your version of Microsoft Excel is 32-bit or 64-bit. There are known issues with Hyper Historian interacting with 64-bit Microsoft Office. Try installing a 32-bit version of Microsoft Excel and repeat step 6.

7. A list of tables appears as shown in Figure 8. You can select table TAGS (display all tags in HH configuration), RAWDATA (display logged data for selected logging group) or QUERYSETTINGS (display setting for the SQL query). Select one and click Next.

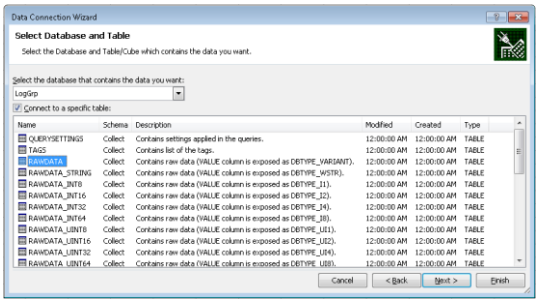


Figure 8 - Selecting Table

8. The Data Connection Wizard window appears. Configure the name and description for your data connection file. Click **Finish**.

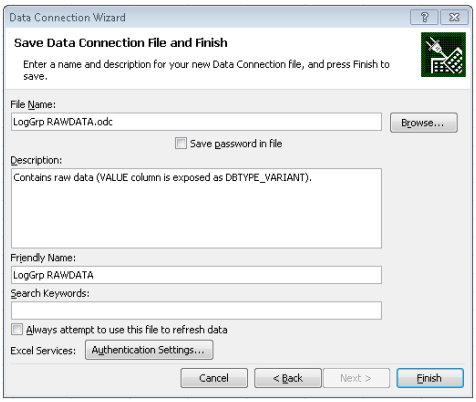


Figure 9 - Data Connection Wizard

9. The Import Data properties dialog appears. You can change properties to your liking or leave them as they are. Click **OK** when you are ready to continue.

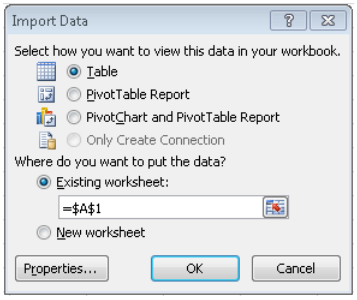


Figure 10 - Import Data Dialog

10. The query may take a few moments to complete. The result of the query will look like the following:

TAGNAME	TIMESTAMP	QUALITY	VALUE
Ramp	6/10/2010 14:33	0	77
Random	6/10/2010 14:33	0	1.498458815
Random/RandomMaxActTime	6/10/2010 14:33	2147484672	1.498458815
Sample/Step	6/10/2010 14:33	0	73.68421173
Sine	6/10/2010 14:33	0	99.60573507
Ramp	6/10/2010 14:33	0	78.124
Sine	6/10/2010 14:33	0	99.03987682
Ramp	6/10/2010 14:33	0	79.124
Sine	6/10/2010 14:33	0	98.33081016
Ramp	6/10/2010 14:33	0	80.25
Sine	6/10/2010 14:33	0	97.30426794
Ramp	6/10/2010 14:33	0	81.25
Sine	6/10/2010 14:33	0	96.19397663
Ramp	6/10/2010 14:33	0	82.374
Sine	6/10/2010 14:33	0	94.72863613

Figure 11 - Logged Data in MS Excel

NOTE: You can also use Hyper Historian SQL Query Engine in the BizViz ReportWorX application for creating scheduled reports. See the application note *ReportWorX - Reporting on Hyper Historian Data*.