



GridWorX64 Viewer – Displaying Hyper Historian Data



APPLICATION NOTE

September 2013

Description: Displaying Hyper Historian Data in GridWorX64 Viewer.

OS Requirement: Vista x64/ Windows 7 x64/ Windows 8 x64/ Windows Server 2008 x64/ Windows Server 2008 R2 x64/ Windows Server 2012.

General Requirement: GENESIS64 and Hyper Historian v10.7x (or higher) installation, basic knowledge of Hyper Historian and GENESIS64.

Introduction

GridWorX64 Viewer was developed in order to visualize database data or OPC DA data through a data grid control. GridWorX64 can also read historical data stored in Hyper Historian database using the Hyper Historian OLE DB provider and SQL Query Engine.

This application note provides an example of how to read Hyper Historian data in a GridWorX64 Viewer. This application note also assumes that you have followed though the example in the Application Note titled *GENESIS64 – GridWorX64 Server*, *GENESIS64 – GridWorX64 Viewer*.

Before You Start

1. Before you start to read the data from Hyper Historian database make sure that Hyper Historian is properly setup and you are logging the data into database. If you have never configured Hyper Historian, you can start with the application note entitled, *Hyper Historian - Quick Start*.
2. Double check that Hyper Historian Core Engine is started (via traffic light or service manager). Please notice that the Hyper Historian engine has to be running to read the data from Hyper Historian database using the SQL Query Engine. For more details how to use SQL Query Engine see the related application note, *Hyper Historian - SQL Query Engine Quick Start*.
3. You should load the query from the stored procedure below into SQL Server Management Studio and test it to make sure that is accurate for your configuration. You can test it in SQL Management Studio. For more details how to work with stored procedures in SQL Server see your SQL Server documentation or related application notes.

The stored procedure below needs to be created in your SQL Server (in the Hyper Historian configuration database). We will call this stored procedure from GridWorX64 Server to get our data.

NOTE: This stored procedure can actually be added into any database on the same server as the Hyper Historian Linked Server, not just the Hyper Historian configuration database, but the Hyper Historian configuration database is usually the best choice because it will be easier to find later if you need to make changes.

```
/****** Object: StoredProcedure
[dbo].[sp_GetRawData_perTag_perTimePeriod] Script
Date: 10/26/2010 *****/
/***** Copyright ICONICS Inc. *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE PROCEDURE
[dbo].[sp_GetRawData_perTag_perTimePeriod]
    @StartTime varchar(50),
    @EndTime varchar(50),
    @LoggingGroup varchar(100),
    @CollectingGroup varchar(100),
    @TagName varchar(1000)
AS
Declare @OpnQ varchar(1000)
Declare @SQL nvarchar(3000)

Select @OpnQ = 'SELECT [TAGNAME]
                , [TIMESTAMP]
                , [QUALITY]
                , [VALUE]
FROM
[HH2].[ '+@LoggingGroup+' ].[ '+@CollectingGroup+' ].[RAW
DATA]
                WHERE
TAGNAME=''''+@TagName+'''' AND TimeStamp >=
'''+@StartTime+'''' AND TimeStamp <='''+
+@EndTime+''''

Select @SQL = 'SELECT * FROM OPENQUERY (HH2, ''' +
@OpnQ+''')'

exec sp_executesql @SQL

GO
```

Connecting to the Hyper Historian Database

1. Open up Workbench by going to **Start → All Programs → ICONICS → Workbench**.
2. In the Project Explorer, choose the **GridWorX64 Server** provider.
3. Under the GridWorX64 configuration database, right click on **SQL Server Connections** and click on **New Database Connection**.
4. Provide a name for the connection such as "Hyper Historian" and click on **Change Connection**.
5. Double click on SQL Server Connections and click on the  button to open up Connection String dialog box. Connect to



GridWorX64 Viewer – Displaying Hyper Historian Data



APPLICATION NOTE

September 2013

your Hyper Historian configuration database. Click OK when finished.

- Back in the Connection Strings dialog, select the connection you just created and click OK.
- Click “Test connection” to check if you can connect to database, then click Apply.

Configuring GridWorX64 Data Source

- Right-click on the database connection you created and choose **New Data Source**.
- Name this data source **GetRawData**.
- In SELECT command tab, click on the pencil icon and choose **Stored Procedure Call**.
- Browse for stored procedure which you created in the “Before You Start” section. It should be named “sp_GetRawData_perTag_perTimePeriod”.

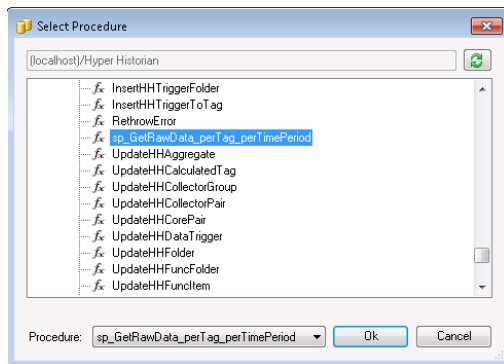


Figure 1 - Selecting the stored procedure

- After you have selected the stored procedure the “Schema Metadata Auto-Discovery” dialog will appear, asking if you’d like to execute the command now to get the schema columns. Click “Yes, Execute the Command now...” When prompted, fill in parameter values appropriate for your Hyper Historian configuration. The parameters may look like Figure 2.

NOTE: Be sure to use the Name of your group and not the Display Name.

NOTE: If you have more than two levels of folders, the additional levels should be added to the TagName parameter. So if you have a tag named MyTag nested in Folders named Folder1 → Folder2 → Folder 3, then your @LoggingGroup parameter would be “Folder1”, @CollectingGroup would be “Folder2” and @TagName would be “Folder3/MyTag”. See the help files or application notes for more details on the correct values for the SQL Query Engine.

Parameter Values		
Please assign values for input parameters:		
Parameter Name	Data Type	Value
@ RETURN_VALUE	DBTYPE_I4	N/A
@ @StartTime	DBTYPE_STR	2013-01-01 12:00:00 AM
@ @EndTime	DBTYPE_STR	2013-02-01 12:00:00 AM
@ @LoggingGroup	DBTYPE_STR	LogGrp
@ @CollectingGroup	DBTYPE_STR	OPCUA
@ @TagName	DBTYPE_STR	Sine

Figure 2 - Example parameters

- Once you are done you can run test query by clicking on the “Test SELECT Command” button. You can use the same parameters that were used for the schema auto-discovery. A successful query should return output that looks like Figure 3.

Query Results			
Table: 1000+ record(s) returned			
TAGNAME	TIMESTAMP	QUALITY	VALUE
Sine	9/6/2013	2161573888	
Sine	9/6/2013 2:34 PM	0	31.412811793591523
Sine	9/6/2013 2:34 PM	0	28.183778031757541
Sine	9/6/2013 2:34 PM	0	25.058064025642693
Sine	9/6/2013 2:34 PM	0	22.062352306990807
Sine	9/6/2013 2:34 PM	0	18.895034368011899
Sine	9/6/2013 2:34 PM	0	16.423798123996548
Sine	9/6/2013 2:34 PM	0	12.757393264211643
Sine	9/6/2013 2:34 PM	0	10.015767079136381
Sine	9/6/2013 2:34 PM	0	7.9970265215211498
Sine	9/6/2013 2:34 PM	0	6.1846660007051071

Figure 3 - Test results

- Remember to click Apply when done.

Reading Data in a GridWorX64 Viewer

- Open **GraphWorX64**.
- Select Controls ribbon and add a GridWorX64 Viewer to your display.
- Double click on the GridWorX64 Viewer control to bring up the configuration dialog box.
- Click on **Grid** from the tree on the left.
- Under **Source** tab, click on  to create a new source. Choose **Data Set Tag Subscription**.
- The tag browser will appear. Browse for **Databases → SQL Server → Hyper Historian → Data Sources → GetRawData**. Then click OK.



GridWorX64 Viewer – Displaying Hyper Historian Data



APPLICATION NOTE

September 2013

7. In the Select Data Tag field, fill in the parameter values after each equal sign. Use double quotation marks around each parameter value. A final Data Tag string should look something like this:

```
db:Hyper Historian.GetRawData<@StartTime="2013-01-01 12:00:00 AM",  
@EndTime="2013-01-01 12:00:00 AM", @LoggingGroup="LogGrp",  
@CollectingGroup="OPCUA", @TagName="Sine">
```

8. When fully configured, your Source tab should look similar to Figure 4.

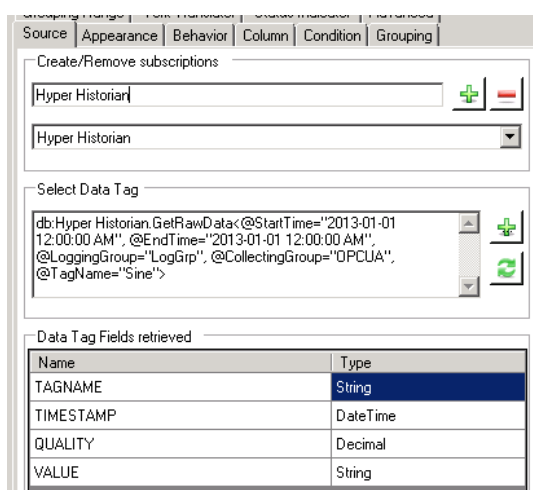


Figure 4 - GridWorX64 Viewer Source tab

9. Click Close to save.
10. Go to runtime and see the table with Hyper Historian data.

Using Global Aliasing for Dynamic Parameters

Rather than hard-coding the values into the GridWorX64 Viewer, global aliases can be specified as parameter values, allowing the user to dynamically change the tag or time range of the viewer in runtime.

1. Edit your previously-created GridWorX64 Viewer, select the Grid, go to the Source tab, and replace the tag name value with a global alias, as shown below. Make sure to leave the quotation marks around the alias.

```
db:Hyper Historian.GetRawData<@StartTime="2013-01-01 12:00:00 AM",  
@EndTime="2013-01-01 12:00:00 AM", @LoggingGroup="LogGrp",  
@CollectingGroup="OPCUA", @TagName="#TagName#">
```

2. Add a pair of buttons to your display. Both buttons should have Set Global Alias pick actions, and should set the value for the TagName alias. Each button should set the value with a different tag name from your logging group, like Sine and Ramp.

NOTE: You may add the TagName alias into the Global Aliasing configuration, but this step is optional.

3. Go into runtime and click the alias buttons. Your GridWorX64 Viewer should reload with data for the new tag.