

OPC Software Guide & Exercises

Version 1

| | | | |
|-------------|----|-------|----------|
| Created By: | BH | Date: | Jan 2016 |
|-------------|----|-------|----------|

EIT ENGINEERING
INSTITUTE OF
TECHNOLOGY



Using EIT's Remote Lab Instructions

Various softwares used throughout the following practical exercises are installed on EIT's **Remote Lab 1, 4, 9 and 12.**

(Refer to instructions: How to Connect to Remote Labs_Electromet_HTML5.pdf)

Note: On the remote labs – you do not need to buy any add-ons.

Practical Exercise 1: Basic Client/Server interaction

1.1 Objectives

Using Remote Lab 1 and Lab 4

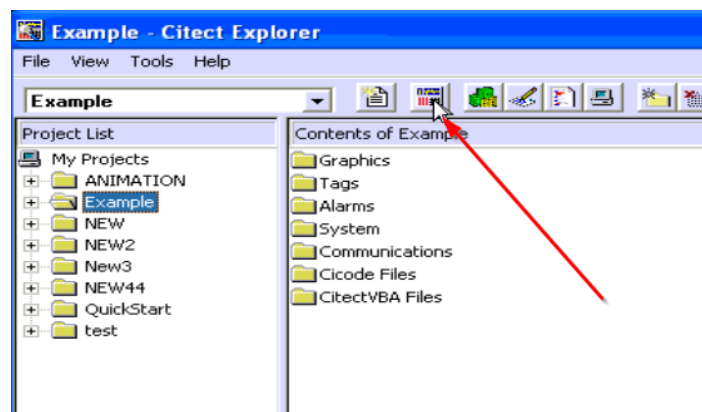
To demonstrate the ease with which data can be accessed from an OPC-enabled SCADA system. In this case we will use Citect version 5.4. This version of Citect has not been implemented with OPC, but rather with custom software drivers. An OPC compliant Data Access server version 1 (DA1.0) was then added as a 'back end' to the system. Also take note that the CatID was (accidentally) omitted from the Citect registry entries, which will explain some of the Clients' behavior. The Citect SCADA and the OPC client are installed on the Server in the Office and are accessed through [Electromeed](#) as usual using demopass to get access to the remote labs after you have logged in:

1.2 Procedures

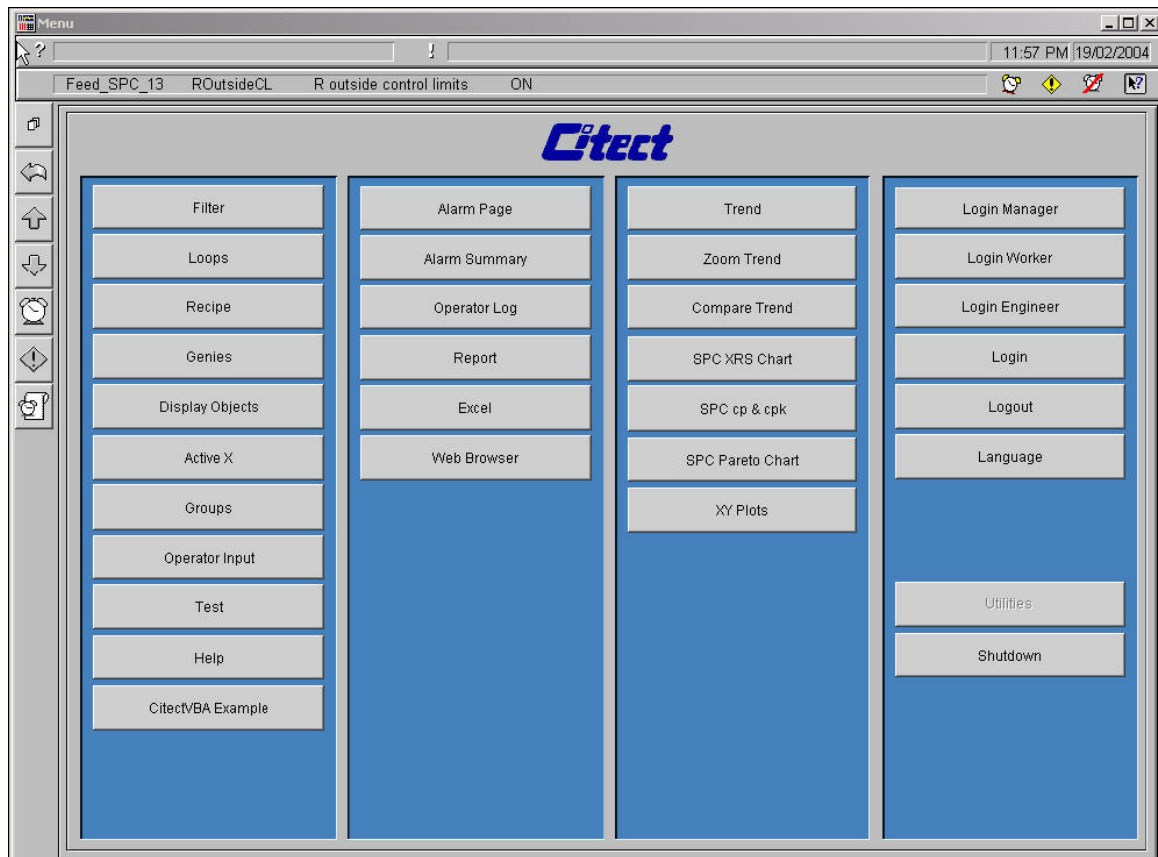
Run Citect by clicking on the "Citect Explorer" icon on the desktop.



Close any other unwanted windows that may open up (there may be several). Select 'Example' under 'My Projects' and then depress the run button (second one from the left) as indicated in the following figure:



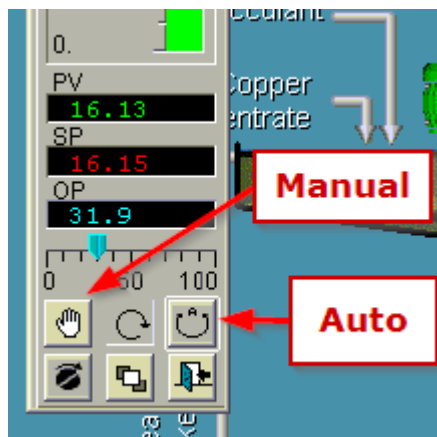
Since you have no software key, it is necessary to run in demo mode when prompted. The example project will now open up.



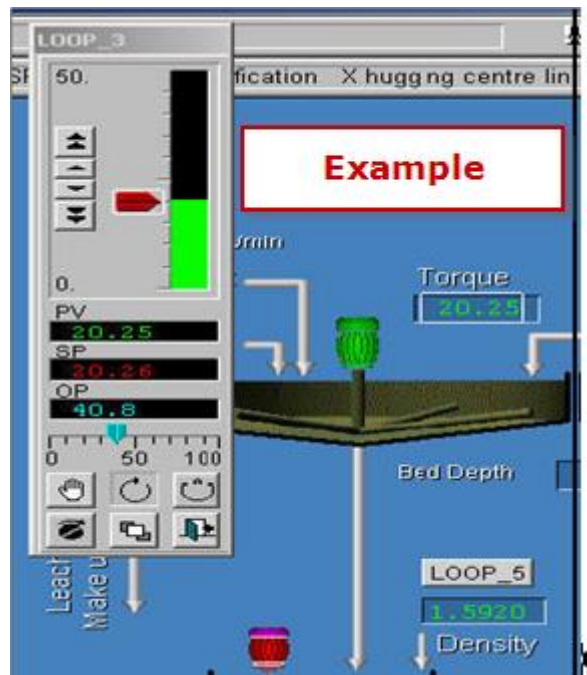
Spend a few minutes and examine all the functions available on the left hand side of the screen (Filter, Loops, Groups, etc.).

Go to the Filter page. Select the process variable 'Torque' (tag ID = loop_3PV). Place the cursor on the numerical display for 'Torque' to see the associated tag ID. Double-click on the Torque display to invoke the PID controller faceplate display for this variable. **Note** the red set-point value display (loop_3SP) and the green process variable display (loop_3PV).

Drag the red arrow up and down and see how the values change. If its not automatically controlling it may be in manual – have a play



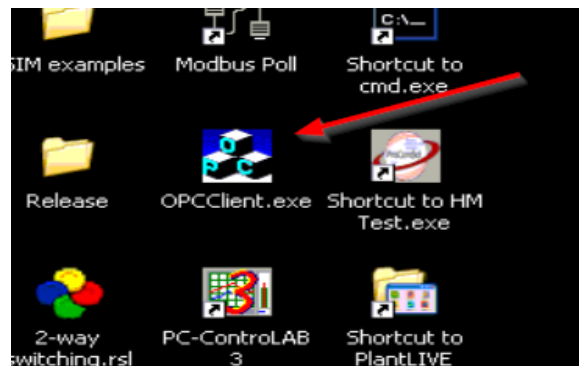
Task 1: Drag it as high as you can and take a screen shot and paste this in your assessment document. Record separately in the assessmet document the Max value that you read with the slider all the way up.



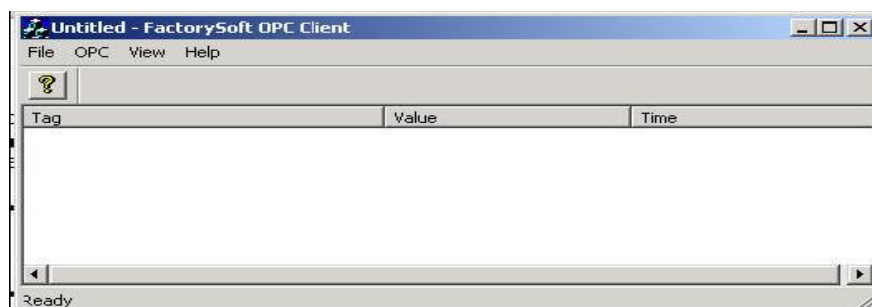
Leave the Citect session running

Now we will run a very simple diagnostic client (FactorySoft).

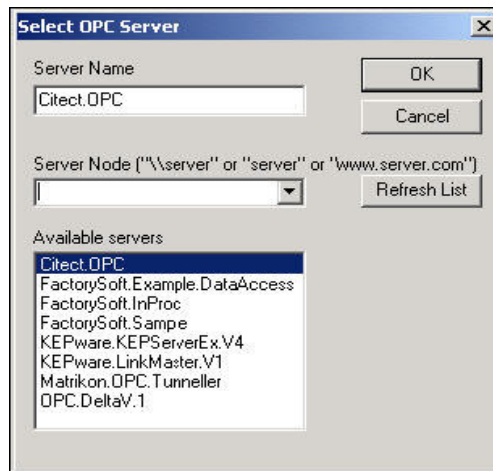
Locate the FactorySoft OPC Client icon on the desktop. It has OPC on a Blue Background.



Double-click to start the Client. The following will appear on your screen:

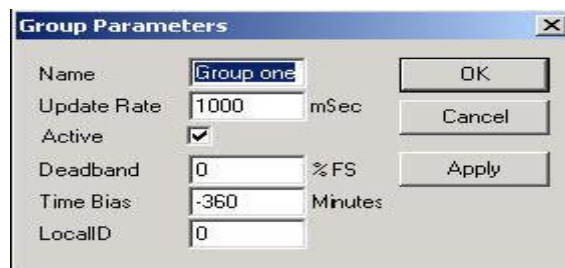


The next step is to establish a connection between the client and the Citect OPC Server. Click OPC -> Connect and wait for the following to appear:



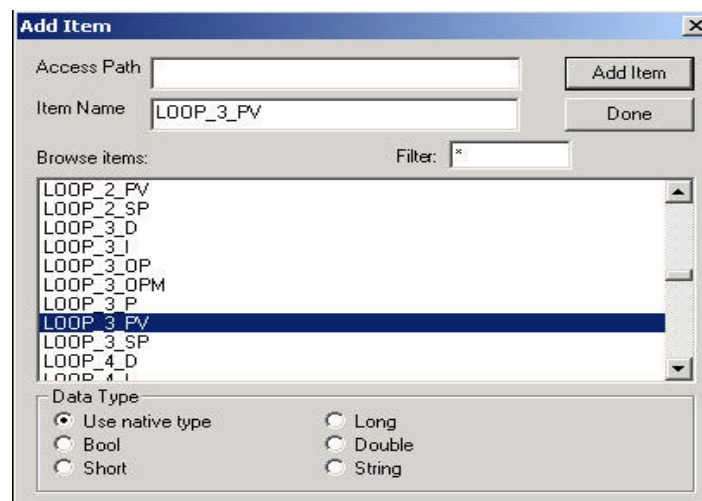
Locate the Citect OPC Server (ProgID = Citect.OPC) amongst the available servers. This will now appear as 'Server Name' in the upper window. You may ignore the server node entry as it will default to the local host. Now click OK.

The next step is to create a Group object as this is necessary in order to access the tags. Click OPC -> Group Parameters. The following window will appear:



In this particular case the group name is 'Group one', the update rate is once every second, the group is active, the dead-band is 0% of full scale, the time difference between the client and server (in terms of time zones) is -6 hours and the language used by the client (local ID) will default to English. Pretend that this is all fine and click OK.

Now it is time to add the items to the group. Click OPC -> Add Item. The following window will appear:

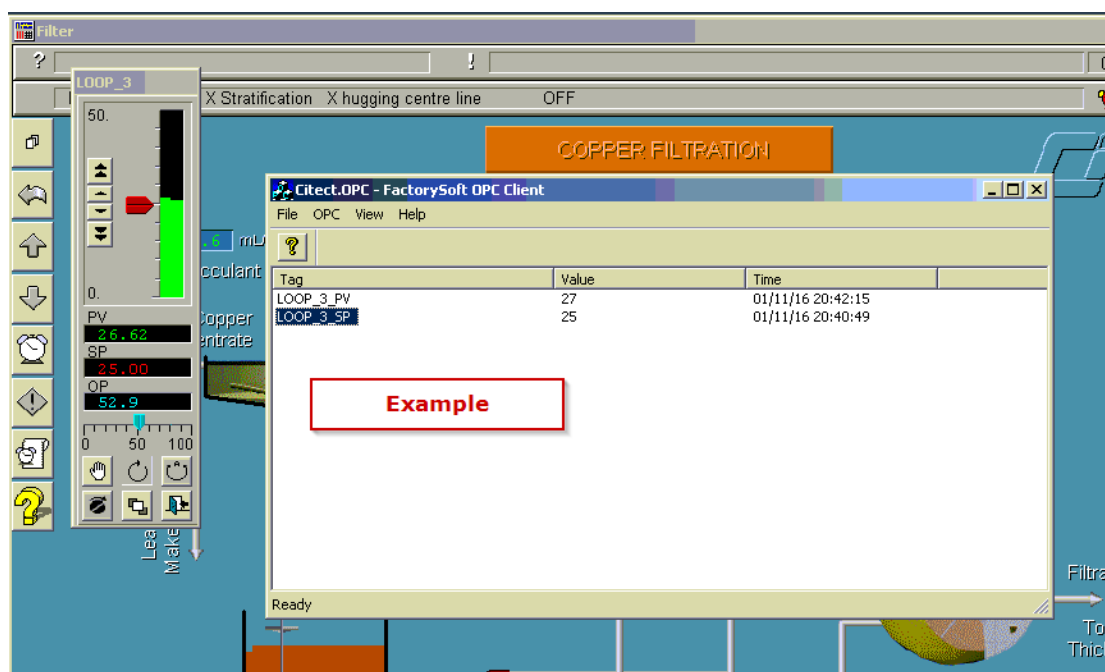


Select the items (as many as you wish) and enter them by clicking on the 'Add Item' button. For example, you may enter loop_3_PV (a read-only value) and loop_3_SP (a read/write value). When finished, click 'done'. The process variables will now appear on the client display.

Maximize the filter display (the client display will be hidden) then select the client again so that it is now superimposed on the plant display.

Use your cursor to vary the set-point on the PID display and observe how the client values change accordingly. Now right-click on loop_3_SP (loop 3 set-point value) in the client display and attempt to write back to the server. You may also attempt to this with a process variable and observe what happens.

Task 2 – Take a screen shot of this opc client to show what you have selected – with the current date and save it into your assessment document.



Remember to close all the programs on the remote lab before exiting the session.

Practical Exercise 2: OPC Server

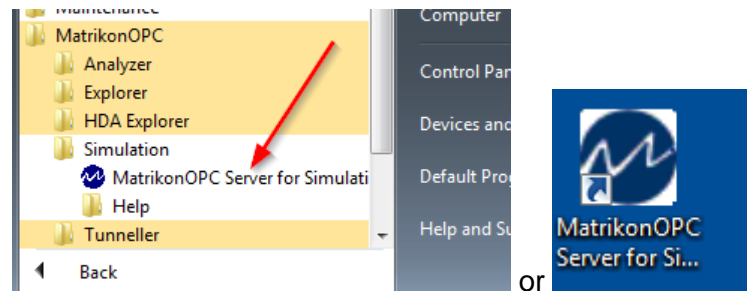
2.1 Objectives

The purpose of this exercise is to familiarize yourself with a fully-fledged Data Access server.

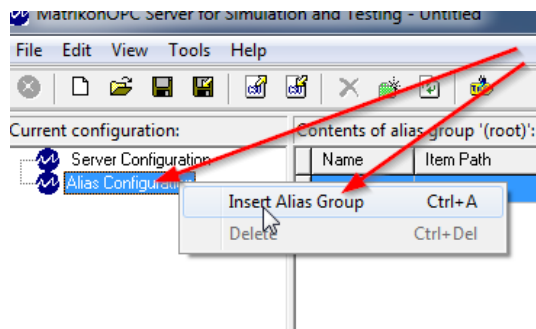
The Matrikon, Cogent and Kassl Software is installed on **Remote Labs 9 and 12.**

2.2 Procedure

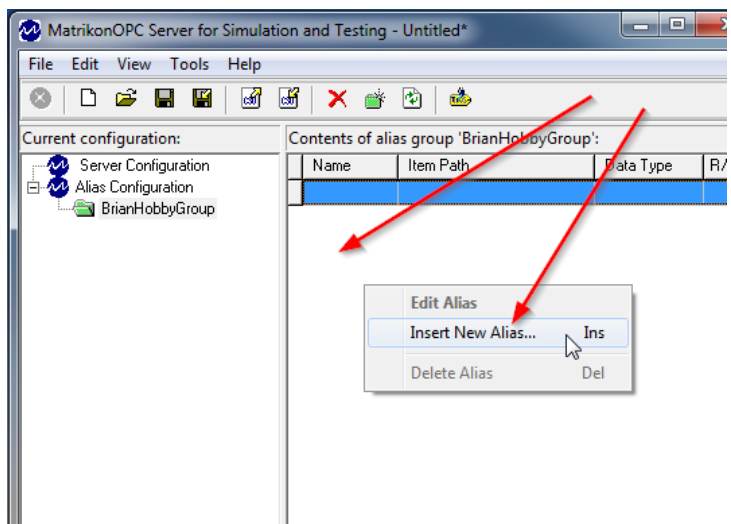
First we set up our own defined alias on the server so that we can see that it's your work. Start up the server configuration application on the Lab Computer



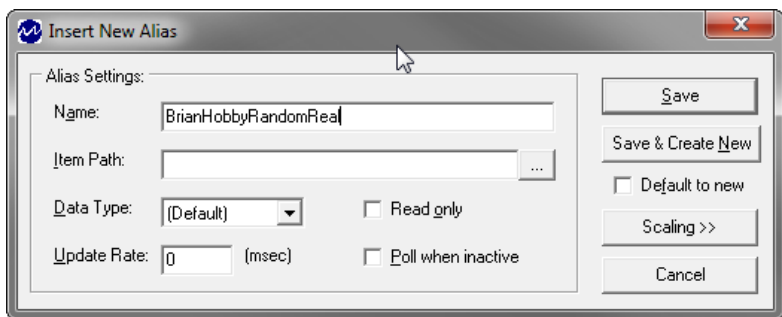
Right click on the Alias Configuration, this allows you to generate custom groups and names for the tags on the server that can be more “human” readable. Call it “YourNameGroup” eg. **JackSmithDIT07**.



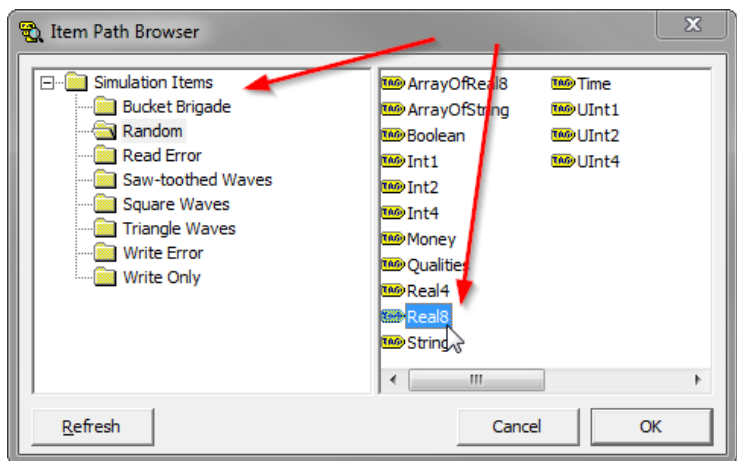
Now right click in the right hand pane and select Insert New Alias



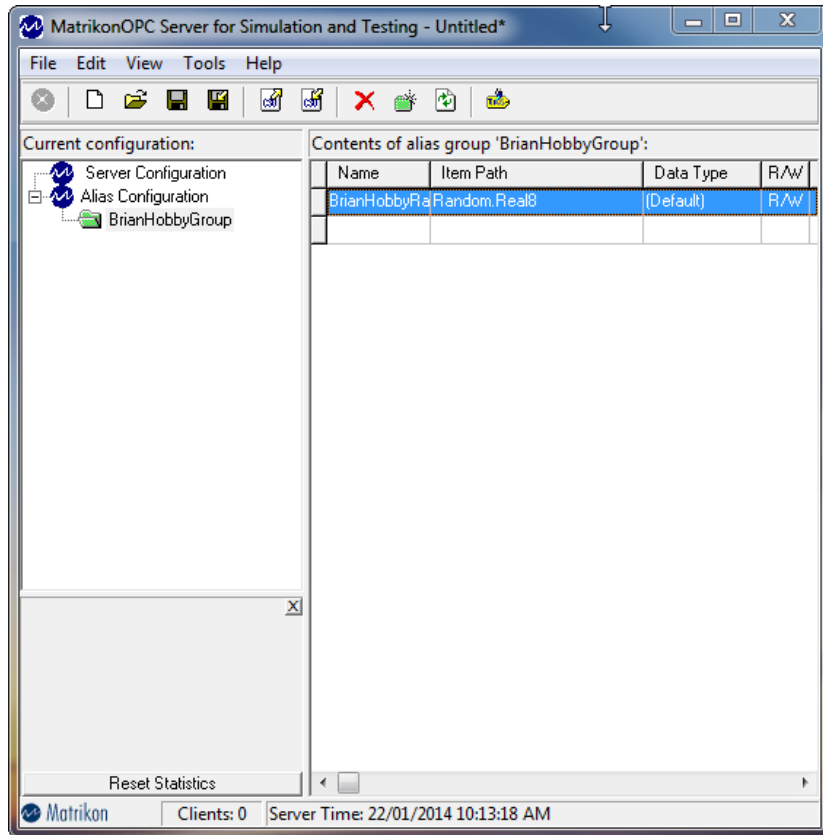
Call it something unique that allows it to be identified as yours



Then click on the button at the end of the Item Path entry field and navigate the tree structure and select a tag from the right hand pane – you are now setting up “items” in the server that can be requested from the client.



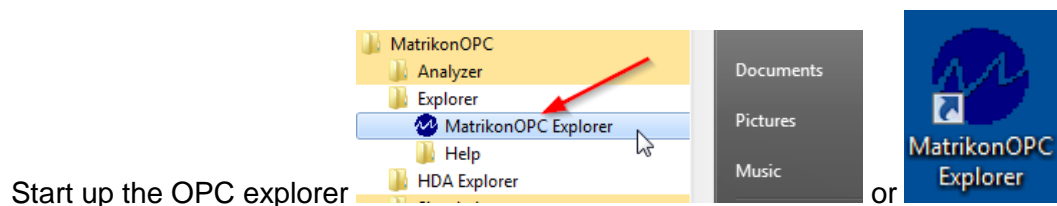
Click OK and then Save the Alias.



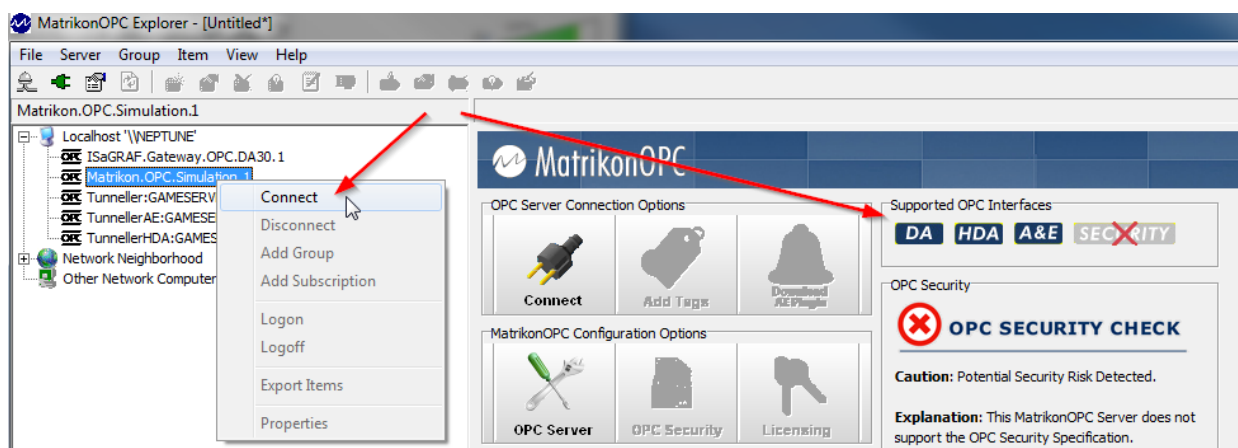
Task 3 - Copy a screen shot of your server window with your name in the group and the alias name and paste into your assessment document.

You can save the server configuration if you want to so you can easily come back and have it the same.

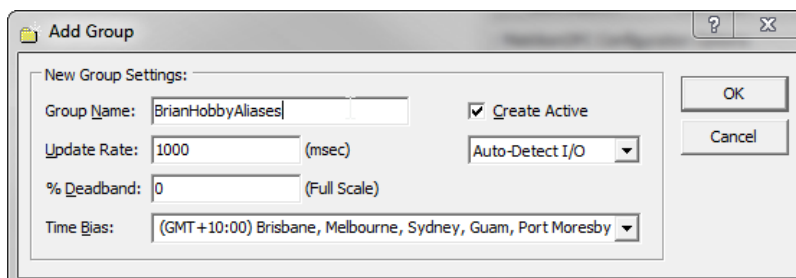
Next time to get the data across the COM interfaces to the client.



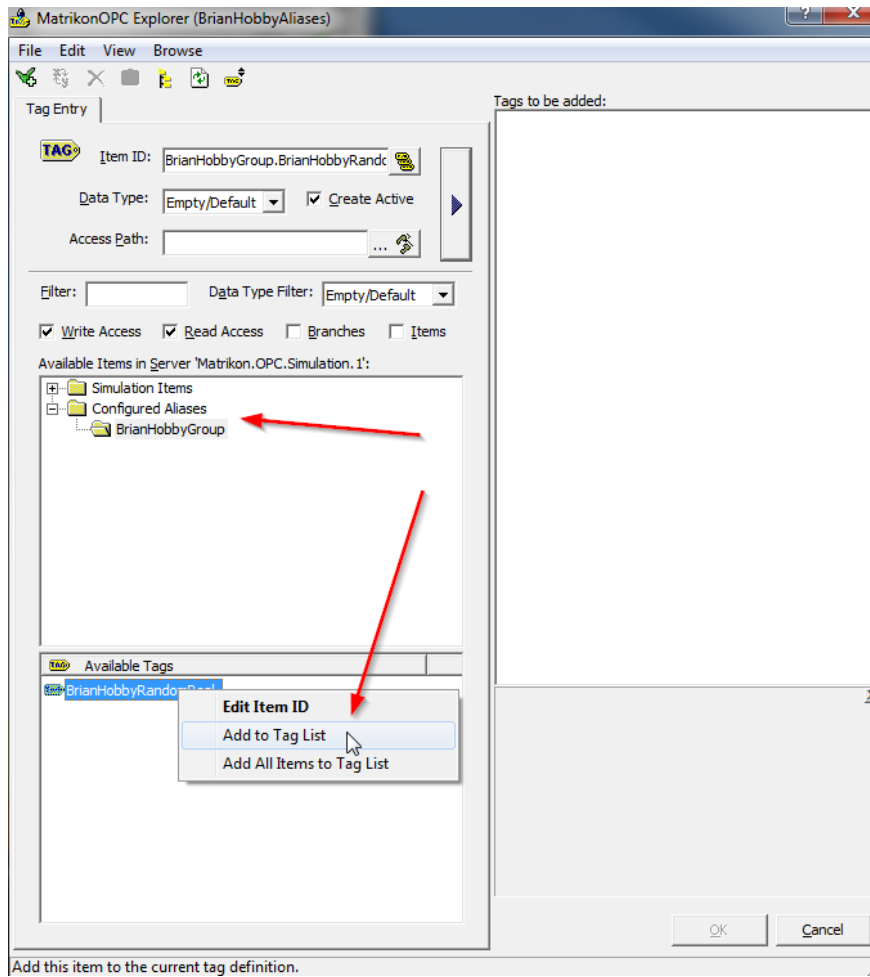
Right click on the Matrikon.OPC.Simulation.1 and connect. If you have several servers available as in this screen shot then as you click them the Explorer will talk to the ICommon interface and request information that is displayed to the right to give you information about the interfaces and categories available.



Right click and select add group – call it something that uniquely identifies it as yours, it can be different to the group you created on the server.

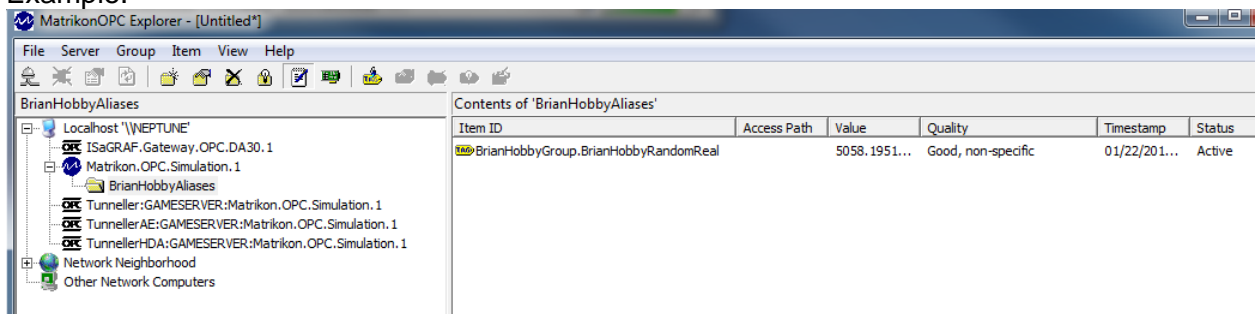


You will have a dialogue box pop up – navigate to the configured aliases group and add the tag you exposed earlier.



Task 4 – Provide a client screen shot into your assessment document – showing your Tag value, clearly showing your name together with the value (naturally there will only be one value in the screen shot).

Example:



Practical Exercise 3: Simple Alarm and Event (AE) Client/Server interaction

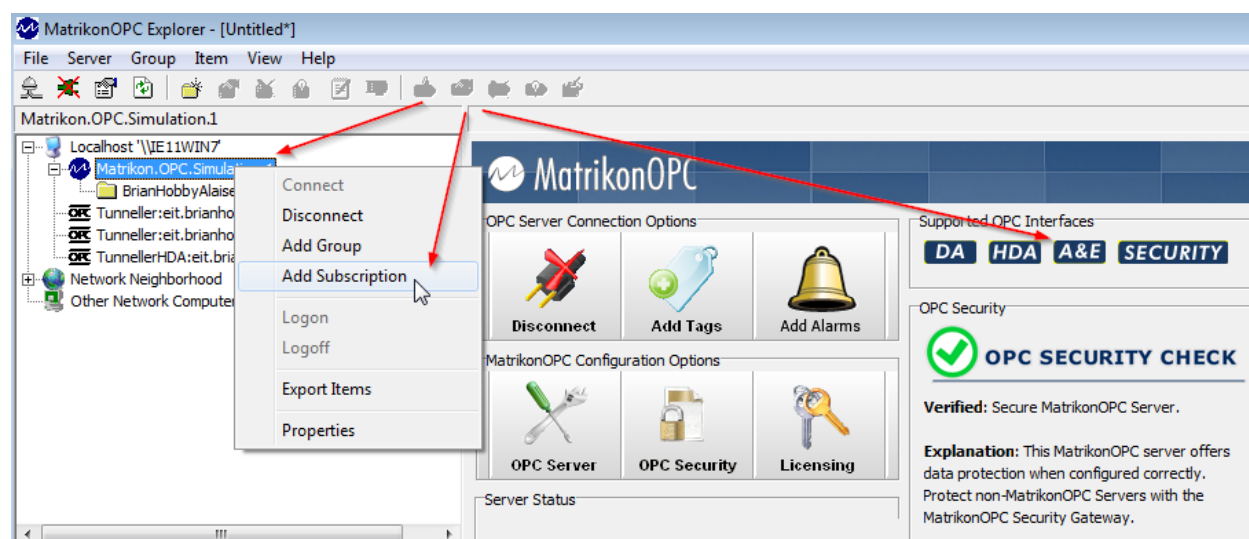
3.1 Objectives

The purpose of this exercise is to demonstrate the interaction between a simple AE (Alarms and Events) Client and an AE Server.

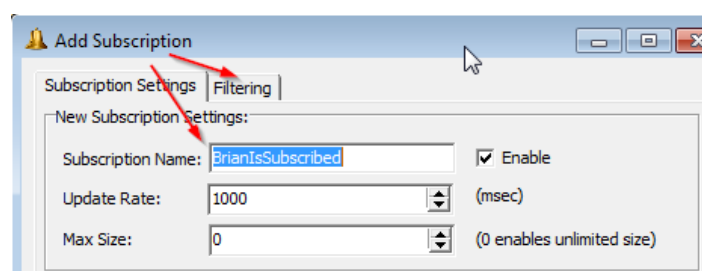
3.2 Procedure

Because of the way the server is configured it generates an event whenever a value is written to this particular string.

Next lets subscribe to the event. Change back to the Matrikon Explorer, right click on the Simulation and select add subscription – this is available because the A&E interface is implemented by this server.



Give it a unique name that identifies the subscription as yours and have a look at the filtering options – leave them as default for this.

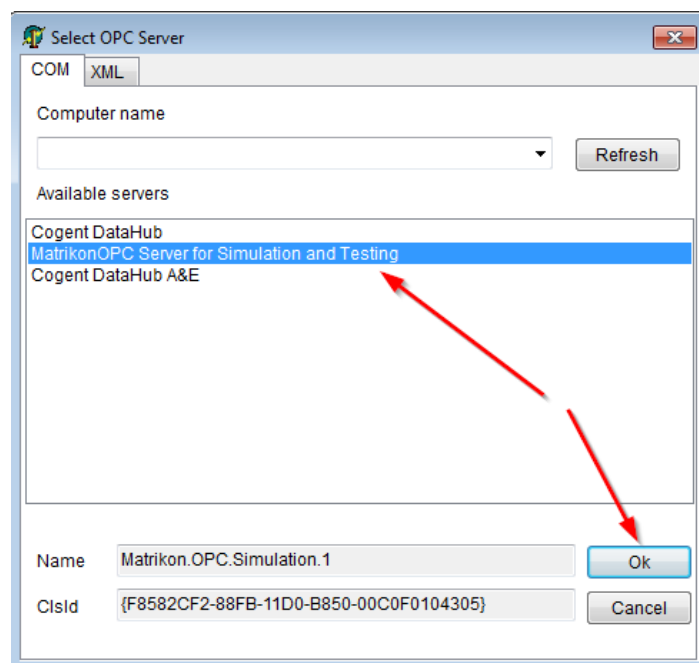
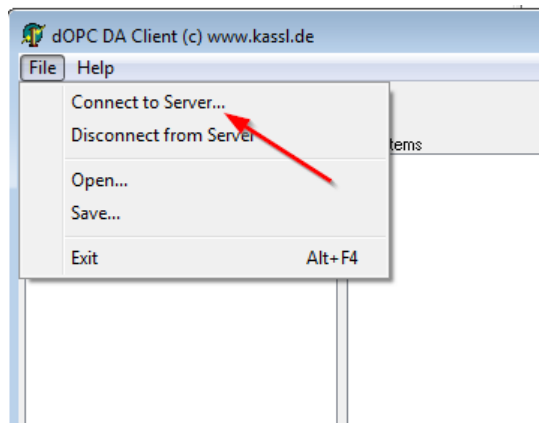


Next we will use the Kassl client to write values to the Bucket Brigade string on the OPC server proving they all play nicely.

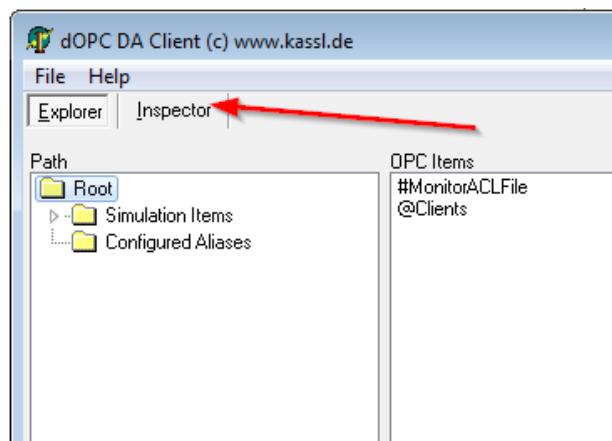
Open the Kassl DA client – on the desktop, make sure you get the **dOPCDAClient** as there are AE and HDA ones as well.



So we need to connect to the server with the Kassl client

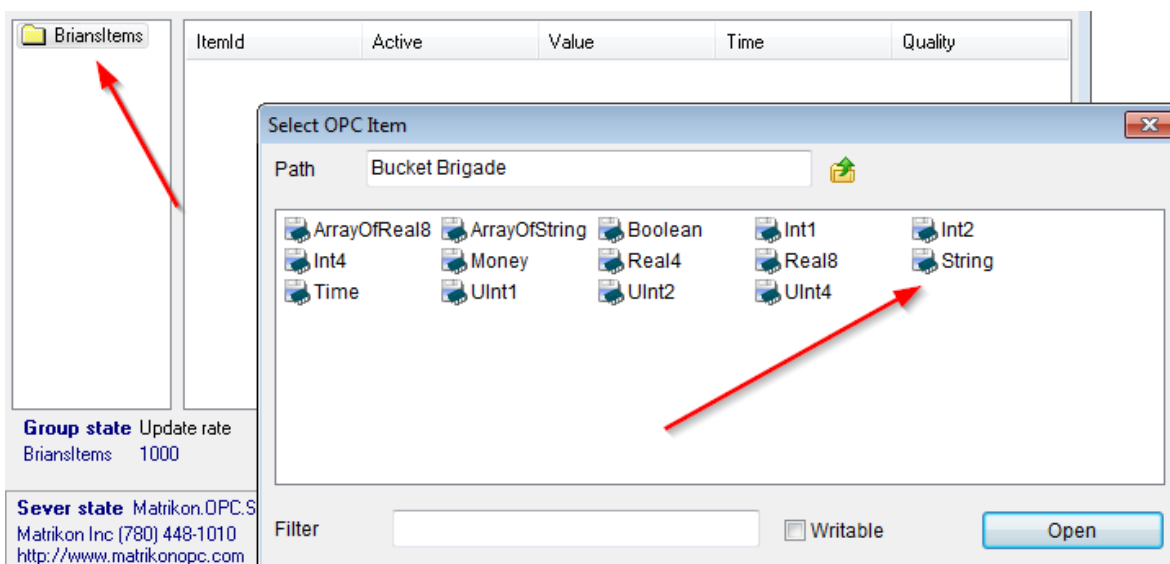


This will give you an explorer view with the tree shown; click the Inspector button to let us make a group – put your name in it.

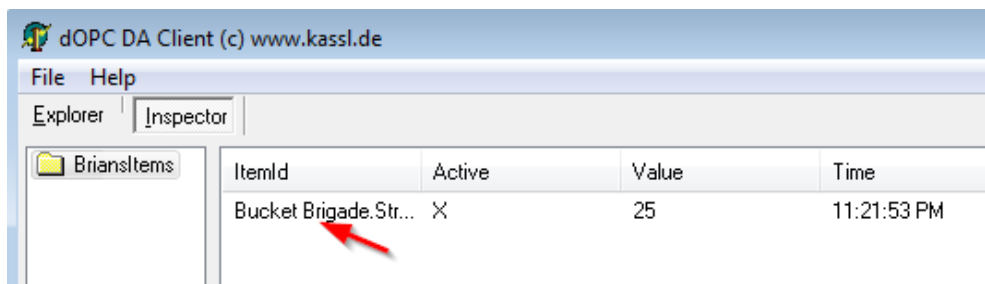




Drill down and add the bucket brigade string item



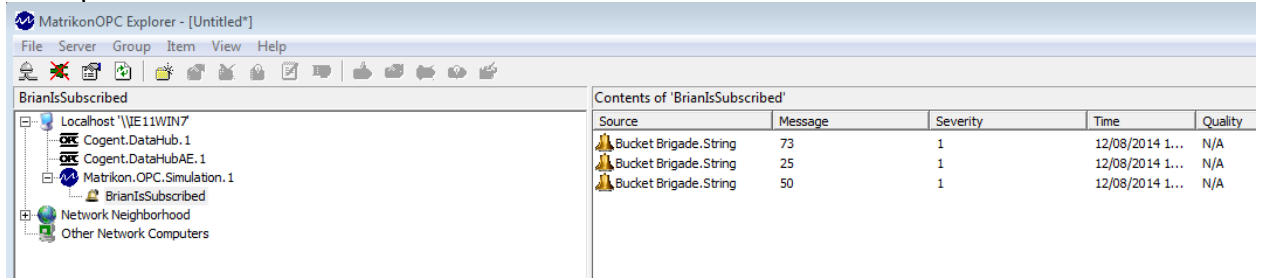
Next right click on the String Item and write a value to it. It is a string so numbers and letters are all permissible.



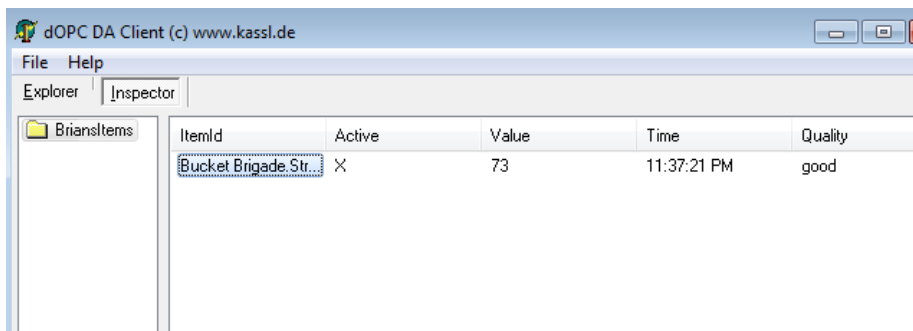
Your Subscription in the Matrikon Explorer should now have a value in it.

Task 5 - Obtain 2 screen shots : one showing a subscription event and another of the Kassi client showing the String Item with a value in it that matches the subscription event. Paste these into your assessment document. This can be one screen shot with both showing if you are able to arrange the windows easily.

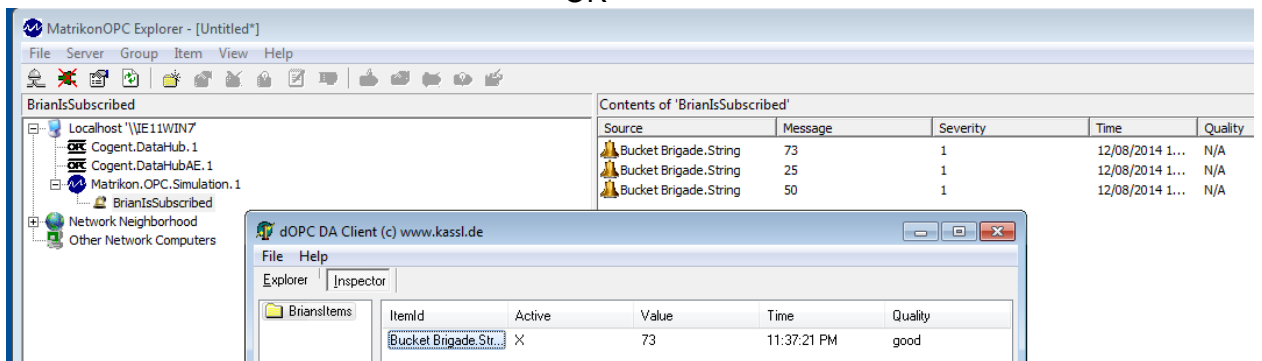
Example



And



OR



Practical Exercise 4: Simple HDA Client/Server interaction

4.1 Objectives

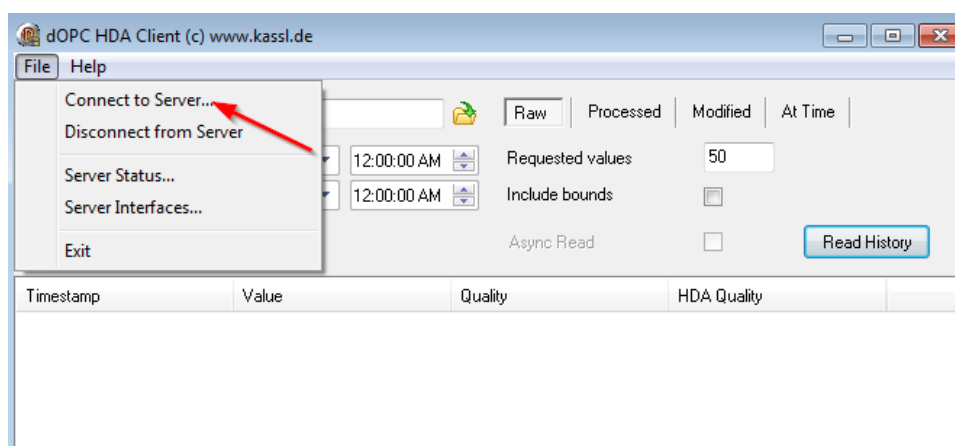
The purpose of this exercise is to demonstrate the interaction between a simple HDA Client and a server.

4.2 Procedure

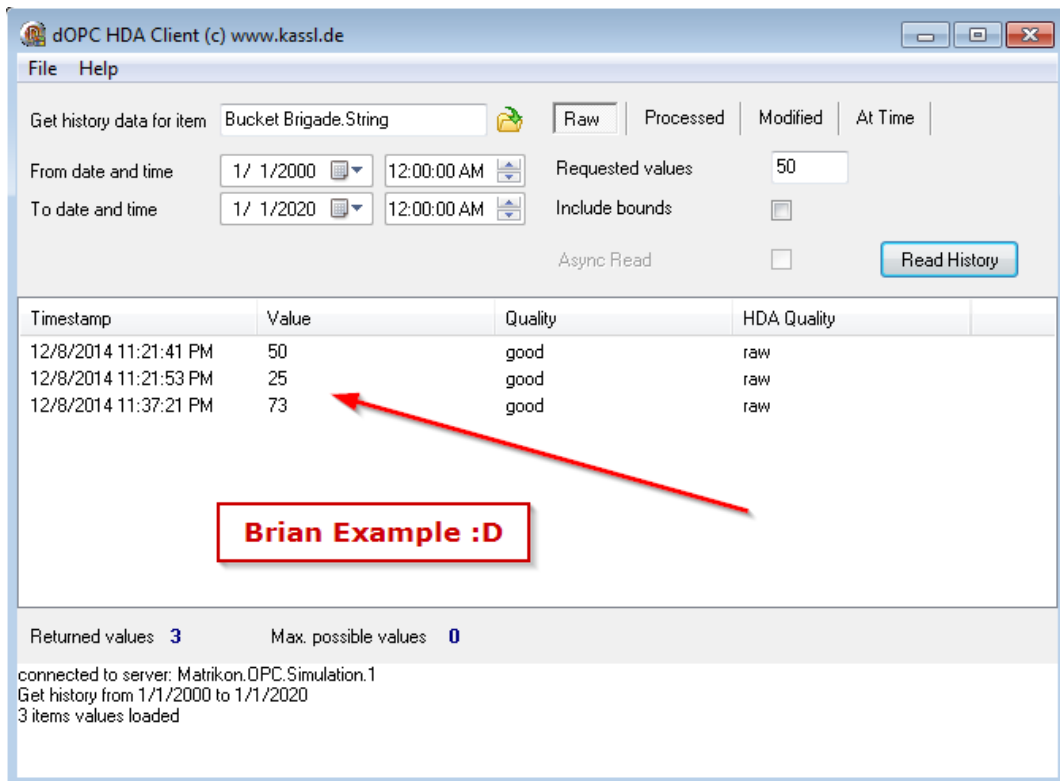
Open up the Kassl dOPCHDA Client, it's a very simple implementation good for checking values out.



As before connect to the server (Matrikon.OPC.Simulation.1) you should see a connected message in the bottom of the client window.



Drill down into the server until you get to the Bucket Brigade.String again, then press the read history button. You should get the values that you wrote into the DA server earlier that generated the A and E events.



Task 6 – Obtain a screenshot of any tag with valid time series data of some sort with a time stamp (Bucket Brigade.String would be a good choice as you should have been changing it in the previous questions).

Practical Exercise 5: Tunnelling

5.1 Objectives

The purpose of this exercise is to demonstrate connection between 2 separate computers using a Tunnelling solution for DA client server

5.2 Procedure

The software is installed on **Remote Labs 9 and 12**. These have been set up to be mirrors of each other with the configuration at each end set to expose the Matrikon Server through the Cogent tunnelling solution.

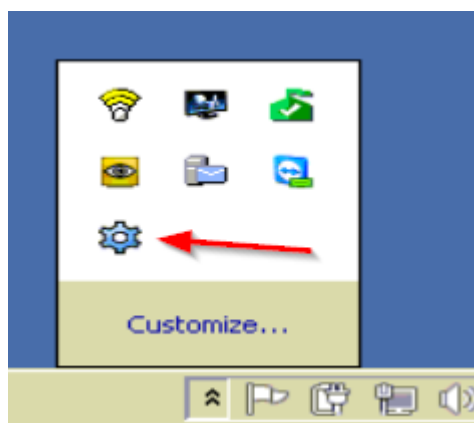
Brief Video on setting up the Cogent tunnel

http://www.cogentdatahub.com/DataHub_Videos.html

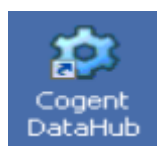
Cogent DataHub Videos

1. Connecting to an OPC Server. (5:00)
2. DataHub OPC Tunnelling in 3 easy steps (no DCOM problems). (5:45)
3. Logging data to a database / Reading data from a database.
 - a) Logging data creates new rows in the database table. (12:55)
 - b) Logging data modifies the same row in the database table. (9:05)
 - c) Reading data from a database. (8:24)
 - d) Using DataHub Scripting to read from a database. (10:50)

To access the Cogent software you double click the Icon in the tray



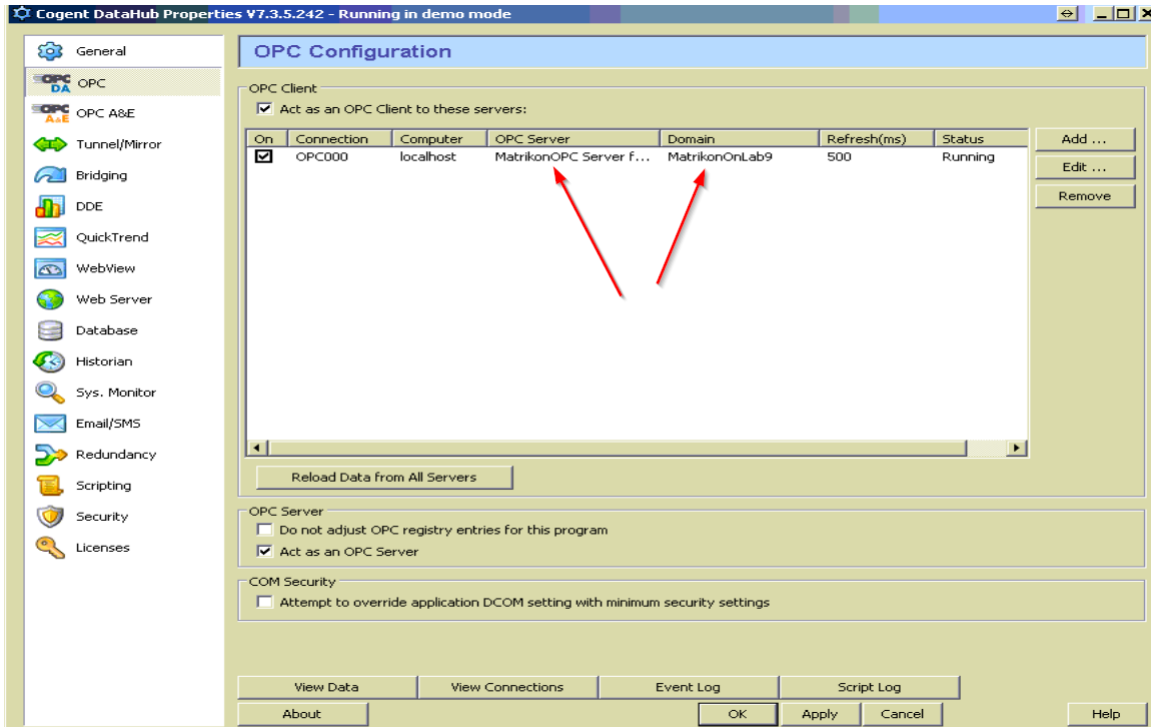
If this is not present you may need to start it up by double clicking the icon on the desktop. If the licence has not been loaded the Cogent software will only run for 1 hour



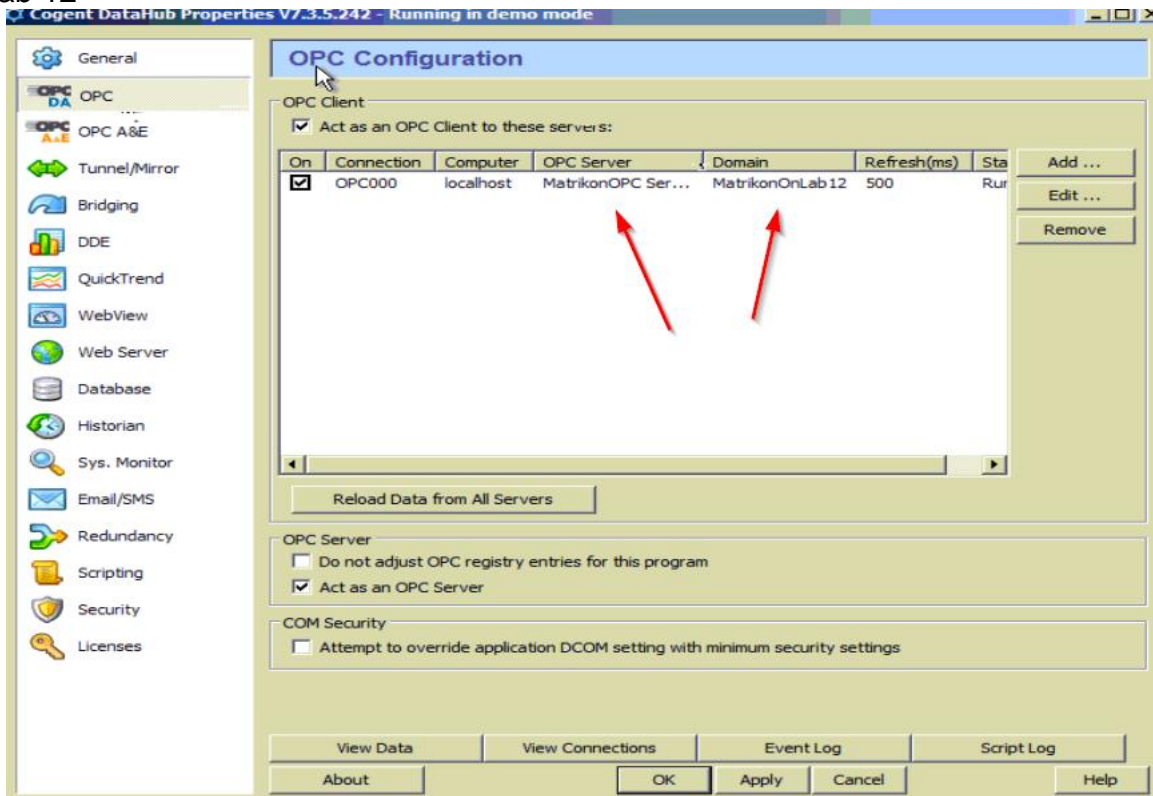
Note: You could conceivably do this question by logging into Lab 9 (or 12) starting the Cogent Software and then completing the rest of the question on Lab 12 (or 9) having started the Cogent software on the second Lab. The labs by default should have the software running on both 9 and 12, if you have trouble then check the other lab to ensure the tunnelling software is running there as well; a tunnel requires 2 ends to work correctly.

On each of the lab computers the the Cogent data hub has been setup as a client to the Matrikon server with a data domain of “MatrikonOnLab9 (or 12)”. Make sure you leave this alone as this is what the student at the other end of the “tunnel” will be connecting to.

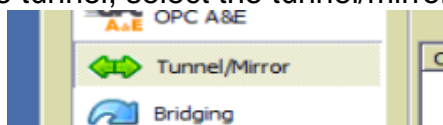
Lab 9



Lab 12



First you need to set up the tunnel, select the tunnel/mirror option in the control panel



Info you will need, IP address'

- Lab 12 is 192.168.1.55
- Lab 9 is 192.168.1.23
- Remote data domain "MatrikonOnLab9 (or 12)"

Click the add master button on the right to get the setup screen.

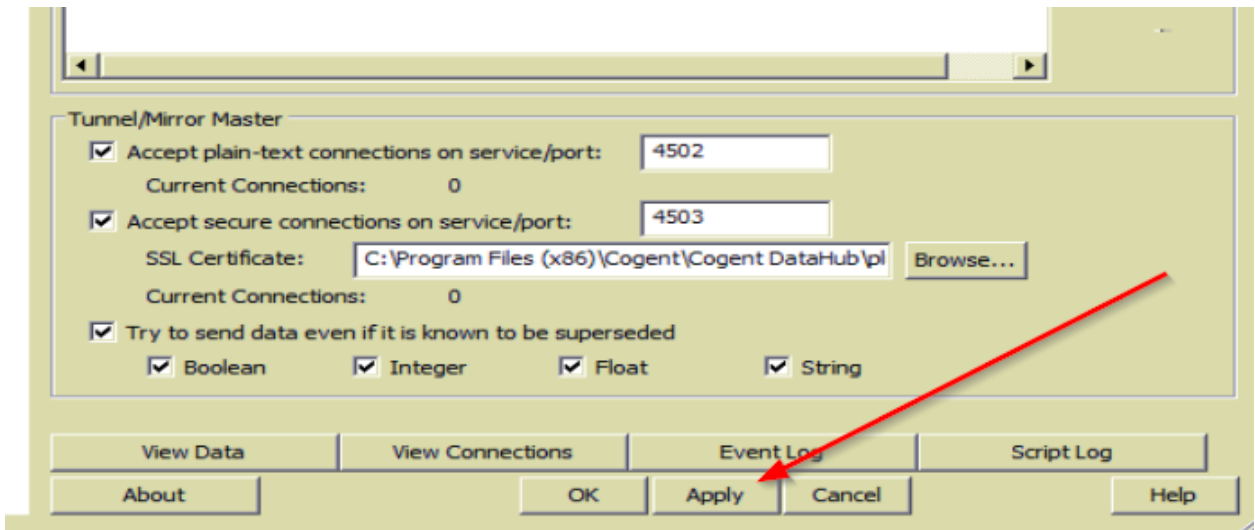
This is set up on Lab 12 so the IP and Remote data domain has been used for Lab 9. Call the local data domain something that uniquely identifies it as your assignment.

 A screenshot of the 'Tunnel/Mirror Master Configuration' dialog box. The dialog has several sections:

- Primary Host:** 192.168.1.23, Port: 4502, Secure (SSL) checkbox is unchecked.
- Secondary Host:** (empty), Port: 4502, WebSocket checkbox is unchecked.
- Local data domain:** BriansOPCTunnel
- Remote data domain:** MatrikonOnLab9
- Remote user name:** (empty)
- Remote password:** (empty)
- Data Flow Direction:**
 - Read-write: Send and receive data to and from the Master
 - Read-only: Receive data from the Master, but do not
 - Write-only: Send data to the Master, but do not receive
- When the connection is initiated:**
 - Get all values from the Master
 - Override the Master's values with my values
 - Synchronize based on time stamp
- When the connection is lost:**
 - Mark data quality here as "Not Connected"
 - Mark data quality on the Master as "Not Connected"
 - Do not modify the data quality here or on the Master
- Connection Properties:**
 - Replace incoming time stamp with the local current time
 - Transmit point changes in binary (faster, x86 CPU only)
 - Target is a Cogent Embedded Toolkit server
- Heartbeat (ms):** 1000, **Retry Delay (ms):** 5000
- Timeout (ms):** 5000

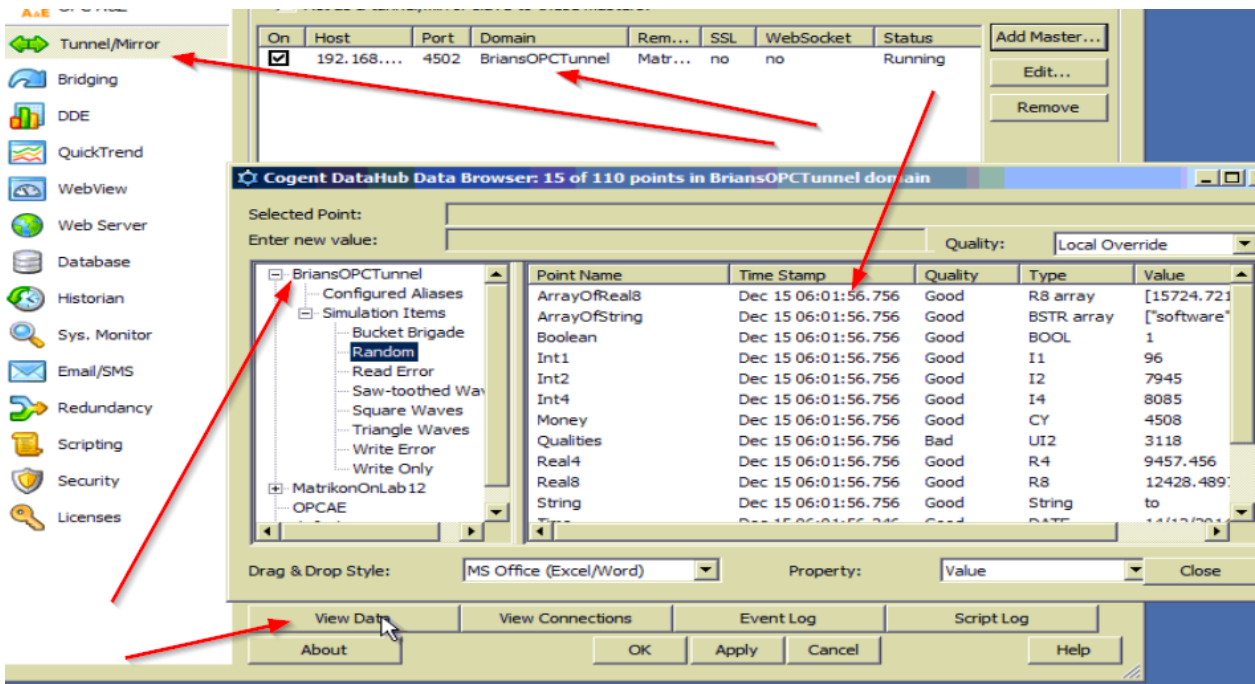
 Red arrows point from the 'Local data domain' and 'Remote data domain' fields to a red-bordered box containing the text: **Call the local domain something that identifies it as yours**.

You must click the Apply button after you ok the configuration for it to save.



Now click the view data button and navigate to the local data domain name you have just configured.

Task 7 - Obtain a screen capture similar to this one showing the data flowing across the tunnel from the other computer.

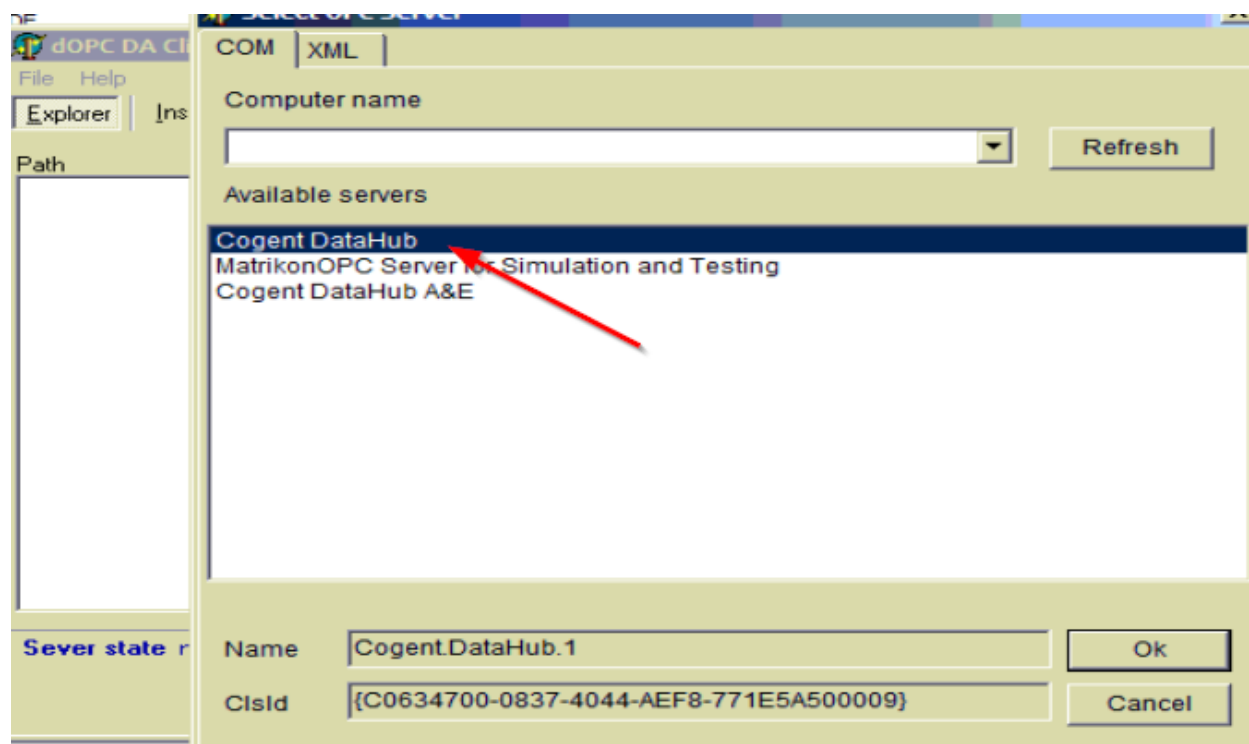


Next let's prove the interoperability of it all.

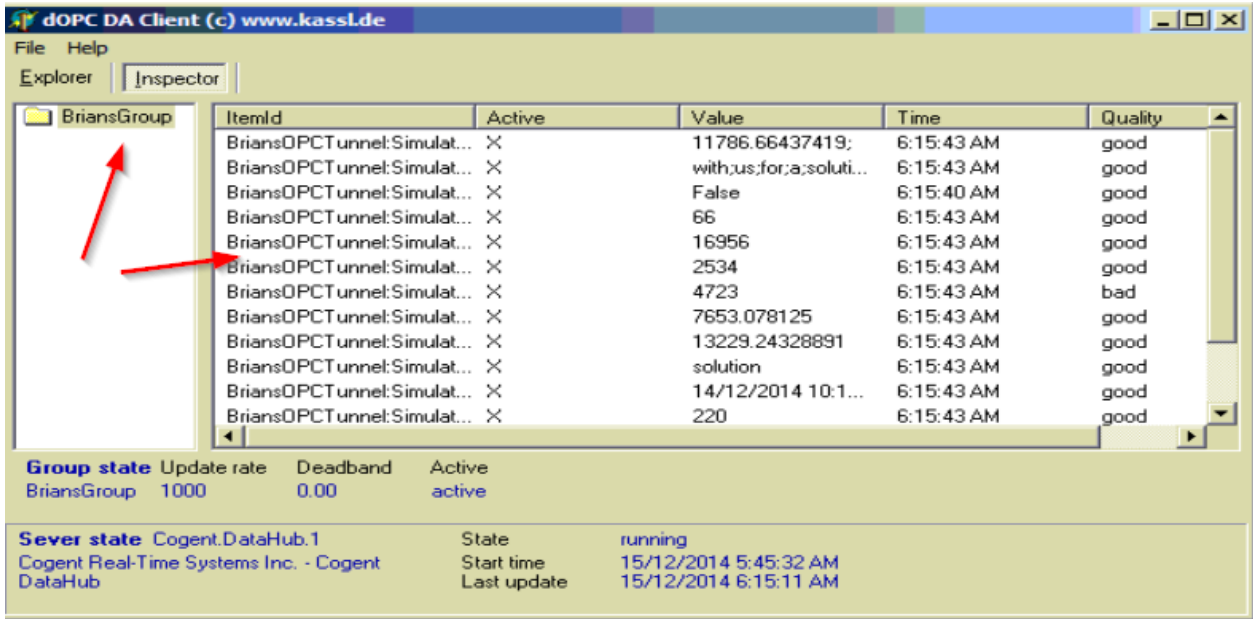
Open the Kassl DA client – on the desktop, make sure you get the **dOPCDAClient** as there are AE and HDA ones as well.



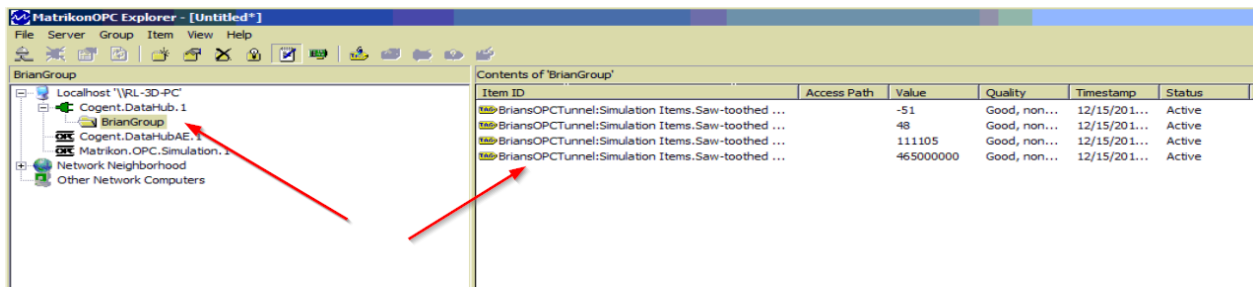
Connect as before to a server but make sure you choose the Cogent DataHub



As previously add a group that identifies it as your own and add some items from the tunnel you have created with your data domain.



And for the final part of the exercise - repeat the above using the MatrikonOPC Explorer
 Here is an example.



Task 8 - Obtain a screen captures of both clients (Matrikon and Kassl) showing showing the data flowing across the tunnel from the other computer.

As a final tidy up please go into the Cogent DataHub and **Delete** your tunnel so the next user can have a clear screen.

Remember you must “Apply” to make the changes confirmed.

