GraphWorX64
Force Asynchronous Read

APPLICATION NOTE

September 2013

Description: Guide to using GraphWorX64 scripting to force an asynchronous read for one or more OPC-DA process points.
OS Requirement: Vista x64/ Windows 7 x64/ Windows 8 x64/ Windows Server 2008 x64/ Windows Server 2008 R2 x64/ Windows Server 2012
General Requirement: Basic knowledge of GraphWorX64 and JScript.NET scripting.

Introduction

Under most circumstances you do not need to worry about receiving updates from an OPC server in GraphWorX64. GraphWorX64 creates a subscription and the OPC server sends it data at the specified rate as long as the data has changed. For most users, this is all that's needed.

However, there are some times when you want to force an update of a process point before the next scheduled update. An example would be if you have a slow or an expensive connection so you have subscribed to your data points with a very slow rate, like every five minutes, but your operators want to be able to change something on the system and see the effect of those changes sooner than the slow update rate. If you keep the slow update rate your operators may have to wait an entire five minutes to see their changes, but if you speed up the update rate you may overload your data connection.

This application note will describe how you can use JScript.NET to force GraphWorX64 to do a single asynchronous read for a process point, updating that process point immediately, regardless of the update rate. This script will only work for Classic OPC-DA points.

This application note assumes that you are familiar with the basics of using JScript.NET in GraphWorX64. If you need more information on this topic, please see the application notes whose titles start with "GraphWorX64 Scripting".

Force Asynchronous Read

Follow these steps to create an example display that uses forced asynchronous read.

1. Create a new GraphWorX64 display. With nothing selected, go to the Properties panel and set the scan rate to a long period of time, like 30000 (thirty seconds).
2. Add a process point and connect it to an OPC tag that changes faster than the scan rate you chose in step 1, or a tag that you can change from outside of GraphWorX64.

   Note: The tag either needs to change on its own or you will need to change it from outside of GraphWorX64 to prove that the asynchronous read worked. It must be a Classic OPC-DA tag, not a simulator point or OPC-UA tag.

3. With the process point selected, go to the Properties panel and give the process point a name of "MyPPT".

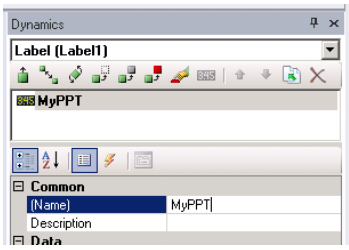

Figure 1 - Process point name

4. Add a button to the display. Choose "Run Script" as the action.
5. Click on the Event button (lightning bolt icon) to go to the Events section. Double-click in the blank field next to Run Script to open the Script panel and create a new function.

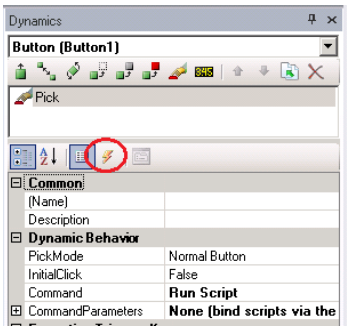

Figure 2 - Events button

6. Inside the function, add the code in the "Button Code" section at the bottom of this page.
7. Save the display and go into runtime. If you do not push the button then the process point will only update once every thirty seconds, but when you push the button, regardless of when the last update happened, you should get a fresh update.

Button Code

```
var dynamic: Ico.Gwx.GwxDynamic;
dynamic = ThisConfiguration.GetDynamicObjectByName("MyPPT");
dynamic.RefreshPrimaryDataSource();
```