ScriptWorX2010 – Features in the ScriptWorX2010

APPLICATIONS NOTE

April 2010

Description: Guide to the features in ScriptWorX2010
OS Requirement: Win 2000, XP Pro, Server 2003, Vista, Server 2008, Windows 7
General Requirement: General knowledge of GENESIS32 and ScriptWorX32

Introduction

ScriptWorX2010 is the next generation of ScriptWorX32. We have added major improvements and features to ScriptWorX2010. We will describe in these in this document. You can still use ScriptWorx32 if you wish to, but you may decide to use ScriptWorX2010 instead after learning the many advantages to this new application.

Compatibility with ScriptWorX32

ScriptWorX2010 will not import or work with ScriptWorX32 *.swx files so it is not directly compatible. Since the threading model, triggering mechanisms, variables and thread options are so vastly different between the two versions we could not make ScriptWorX32 configurations directly upgradeable. Since both versions are still based on VBA, you can certainly take code snippets from ScriptWorX32 and use them in ScriptWorX2010 but many internal ScriptWorX32 functions will not be the same in ScriptWorX2010.

New Threading Model

In ScriptWorX32, each time a script was triggered it would be executed in a new thread. The maximum number of threads allowed was specified in the settings, and if there were no threads available then scripts would queue up and wait for a thread to become free. This could sometimes cause scripts to execute in an unpredictable order.

In ScriptWorX2010, scripts are organized into VBA designers, and each designer runs in its own thread. The number of designers in your project specifies the number of threads available. If multiple scripts are triggered within a thread, the scripts will queue up within the thread. The limit of queuing scripts is specified on a per-designer basis, as is the behavior when this limit is exceeded. This new threading model gives the user much more control over the threads used by ScriptWorX2010. This is a major advantage over ScriptWorX32.

New Configurator Layout

We have complete reorganized the configurator in ScriptWorX2010 to better accommodate the new threading model. The new hierarchy is Project → Designer/Thread → Script.

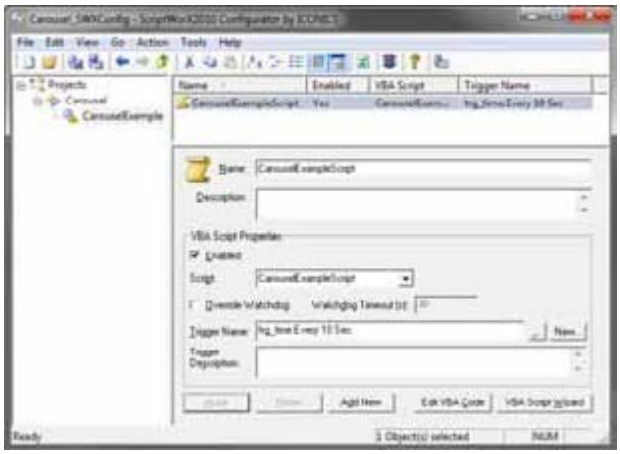

Figure 1 - ScriptWorX2010 Configurator

Projects are associated with .vba files that contain all of the code for the project.

Within a project are **Designer/Threads**. The .vba file for a project is split into multiple designers, each of which is associated with their own thread in runtime. Any functions, subroutines, and global variables within a designer are available to all scripts in that designer.

Scripts link a particular subroutine within the designer to a trigger created with the Unified Data Manager. Whenever the trigger happens for a Script, the associated subroutine will enter the queue for the Designer/Thread.

ScriptWorX2010 Global Variables

are available to all scripts within a Designer/Thread. The global variables can be used to share, watch, and change their values by each script. Any script can change the value of a global variable that is used in another script. You can use a global variable to get the total count of scripts executed within a Designer/Thread.

are available to OPC clients as OPC data through GenBroker Plug-ins (see application note *GENESIS32 - GenBroker Plug-ins Technology*)

support the OnVarChange events, which can catch any change in any global variable value. The event is activated by scanning two parameters: the global variable name and the thread ID of the script. The information is used to deduce if changes of the global variable value were executed from another thread.

NOTE: for more details on ScriptWorX2010 Global Variables, please refer to the application note entitled *ScriptWorX2010 – Easing Integration with OPC enabled Global Variables*.

We have introduced the idea of **Thread identification** as an input parameter. Having thread IDs aids in identifying which

thread has executed a ScriptWorX2010 script. For example, you can set up a condition to decipher if the script is called from the same Designer/Thread or from another thread.

We have added new **Events**. You can run your script on a time period using OnTimer function. Also, when a global variable changes, OnVarChange is activated, from which you can call your script.

Trigger Data introduces the following advantages when a script is executed by firing a trigger:

- Passing the trigger data to the triggered script as an input parameter (i.e. "Public Sub NewScript(td As TriggerData)"
- All triggers are shared for all GENESIS32 and BizViz applications. You can now run a script and a ReportWorX report at the same trigger
- Triggers are managed by Unified Data Manager application

Designer/Thread Options

Every Designer/Thread has the following options in ScriptWorX2010:

Priority: Each Designer/Thread can be given one of 7 priorities (Realtime, High, Above Normal, Normal, Below Normal, Low, and Idle), allowing the user to better manage when the threads are given processor time.

Startup Squelch: ScriptWorX2006 will prevent the thread from executing within a given number of seconds after ScriptWorX2006 is started. A squelch of zero indicates the thread may start as soon as possible.

Timer: A timer can be created for each Designer/Thread. This timer behaves similar to the ICONICS Timer ActiveX. You can use it as a simple way to time events within a script without re-triggering the entire script.

WatchDog: The WatchDog prevents scripts from running out of control. If a WatchDog is enabled for a Designer/Thread, any scripts within the Designer/Thread that run longer than the specified number of seconds will be killed. Individual scripts can override the WatchDog if needed.

Processing Queue: If multiple scripts within a Designer/Thread are triggered to run at the same time, they will queue up. The queue can be configured to hold only a certain amount of scripts and to either ignore new scripts or drop the old ones when it reaches the limit.

ScriptWorX2010 and OPC-DA

ScriptWorX2010 has much more built-in functionality when working with OPC-DA tags.

- VBA global variables are exposed as OPC-DA tags, which are available to any OPC-DA client (such as GraphWorX32, AlarmWorX32, and TrendWorX32).
- The status of a thread is also exposed as OPC-DA tags.
- OPC clients can access the length of the queue, the remaining time in the WatchDog, how many times a script was executed, etc.
- ScriptWorX2010 has built-in functions to read OPC-DA tags.

NT Service Support

You can run ScriptWorX2010 as a service. You can configure this in MonitorWorX. Open Genesis32Tray (GenTray) application, right-click its triangle and select ScriptWorX2010 →NT Service.

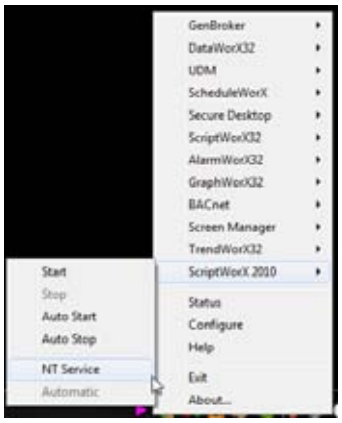

Figure 2 - Running ScriptWorX2006 as NT Service

Alternatively, you can register this application as NT service manually: go to your command line (Start → Run and type in cmd), change directory to your installation (by default "C:\Program Files\ICONICS\GENESIS32\Bin") and register the SWXRuntime.exe file as a service by typing in the command line below:

```
SWXRuntime.exe /service
```

To check the service properties, go to Start → Settings → Control Panel → Administrative Tools → Services. Right-click the ICONICS ScriptWorX2010 and select Properties.

For more information about services with respect to products of ICONICS, see the application note entitled *GENESIS32 – Running Applications as Services on Windows 2000*.