THINKSCAPE LIMITED

USER GUIDE

BIZTALK SCRIPT ADAPTER

WWW.THINKSCAPE.COM

SUPPORT@THINKSCAPE.COM

TABLE OF CONTENTS

1 INTRODUCTION

BizTalk is a great tool. But there are times when you need to solve a simple problem quickly and find that doing it with BizTalk is just going to take too long.

Enter the power of scripting. Windows is great for rapid development. Microsoft provides a Windows script control that has been available for many years. This allows anyone to integrate scripting in to their products. It can also be integrated with .Net and therefore BizTalk. The Script Adapter takes full advantage of this; it combines the immediacy of scripting with the power of BizTalk.

With the Script Adapter you can create a 'one way' or 'solicit response' send port that runs your scripts. This means you can integrate scripting in to your BizTalk solutions simply and quickly.

The Script Adapter supports VBScript and JScript as standard. All the examples presented in this document are in VBScript.

1.1 HOW IT WORKS

When BizTalk sends a message to the Script Adapter send port, either via an orchestration or by a direct subscription, the Script Adapter runs your script passing in the message as a parameter. You can then interrogate the message, perform some calculations, call an external program, write to a file etc.

You can write a script in your favourite scripting language – as long as it is supported by the Windows script control. All you need to do is implement the OnMessage procedure in your script (more on this later), which the Script Adapter calls.

To implement a solicit response script, you must return a value from the OnMessage procedure (it must be a string) which gets converted in to a message and passed to BizTalk by the Script Adapter. This is the message that your orchestration will receive.

One way scripts do not return a value from the OnMessage procedure.

Once you have finished your script, create a one way or solicit response send port (depending upon the type of script you have written) with a transport type of “Script”, configure the port properties, then bind the port to your orchestration. Simple!

1.2 USAGE SCENARIOS

Here are some scenarios where you might find the Script Adapter useful:

- To get something done very quickly.
- To format the content of an email and send it without having to use an orchestration.
- To avoid having to restart BizTalk or redeploy an assembly when you make a change to your code.
- To implement a “new” send adapter rapidly. An example; Create a VBScript that uses the automation API that comes with WRQ's Reflection (the terminal emulator for Windows) to rapidly create a "screen scraping" adapter for integrating BizTalk with legacy terminal based systems.

- To implement "event listeners". Create a Script Adapter one way send port and bind it directly to a message type or a receive port. Any time a message is received that satisfies the binding your script will be run.

- To implement some business rules in a script which are likely to change frequently (an orchestration would call the script and get the results back via a solicit response port).

- Many other things. The power of scripting is enormous.

2 PREREQUISITES

The Script Adapter is for BizTalk 2006. You must install the Windows Script Control which the Script Adapter needs. You can get it here:

http://www.microsoft.com/downloads/details.aspx?FamilyId=D7E31492-2595-49E6-8C02-1426FEC693AC&displaylang=en

3 INSTALLATION

Run the installation wizard and install to a folder of your choosing. The default is C:\Program Files\Thinkscape Script Adapter. You can find binaries in the [INSTALLDIR]\Bin folder.

Once the Script Adapter has been installed you will need to register the adapter with BizTalk through the administration console. Open up the BizTalk administration console. Navigate to the Console Root\BizTalk Server 2006 Administration\BizTalk Group\Platform Settings\Adapters folder.

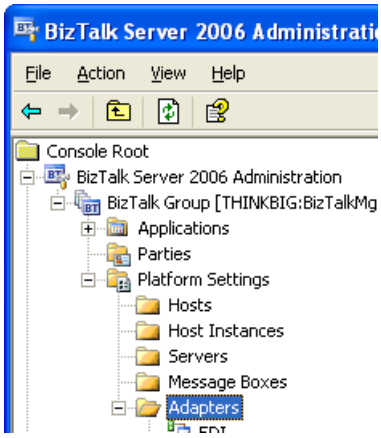

Figure 1

Right click the Adapters folder and select New > Adapter. For the name type Script. Select Script from the drop down menu. For description enter 'Thinkscape Script Adapter'. Now click OK. BizTalk should add the adapter to the list. Make sure you restart the BizTalk instance before you run the samples.

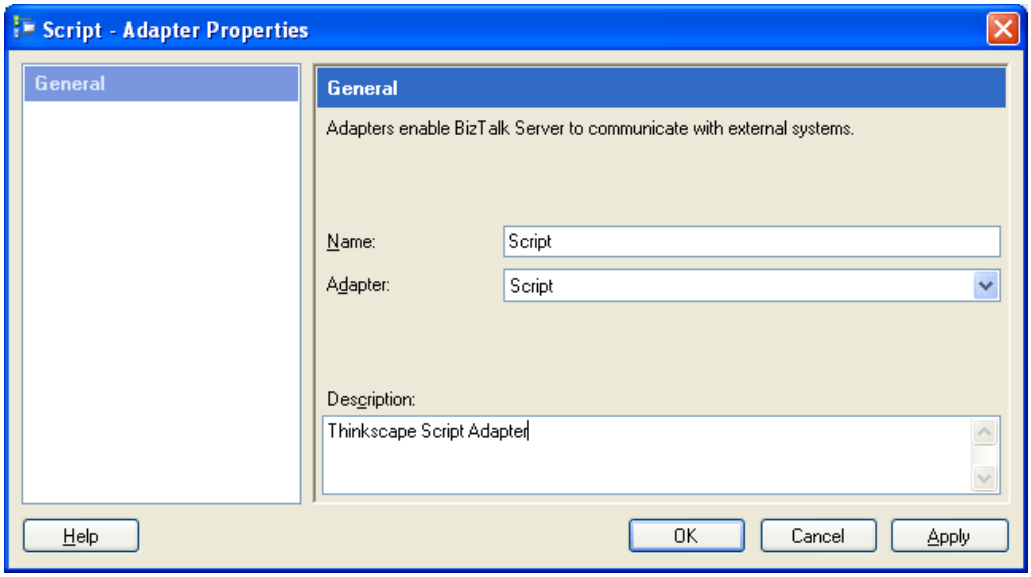

Figure 2

We recommend that you create a new BizTalk instance specifically to host the Script Adapter in a production environment. This is mostly as a protection against memory leaks and fatal errors: because scripts tend to make use of COM a lot, memory leaks and fatal errors are more likely to occur than in say, .Net managed code.

Samples are included with the Script Adapter. To install them, refer to section 7.

4 SCRIPT ADAPTER OBJECT MODEL

The Script Adapter invokes the OnMessage procedure in your script passing a number of parameters to it. The parameters are automation objects with properties and methods that your script can interrogate or call.

If you are implementing a one way send script then use the following signature:

```
Sub OnMessage (Message, Configuration)
  '[Enter your code here]'
End Sub
```

For solicit response scripts use this signature:

```
Function OnMessage (Message, Configuration, Encoding)
  '[Enter your code here]'
  OnMessage = "my return value"
End Function
```

Note that a solicit response script returns a value to the Script Adapter by setting the OnMessage variable to a string. OnMessage is an implicit variable that signifies to VBScript that the value it is set to is the return value of the function.

One way send scripts don't return any values.

4.1 MESSAGE OBJECT

Name	Signature	Description
Text	[object].Text	The message body as a string
Id	[object].Id	The message id as a string
Context	[object].Context	A collection of the message context properties. See Context object

4.2 CONTEXT OBJECT

The Context object is a collection and can therefore be used with the VBScript For Each statement. It contains all the context properties associated with the message.

```
For Each Prop in Message.Context
  '[Enter your code here]'
Next
```

Name	Signature	Description
Read	[object].Read (Name, Namespace)	Returns the value of a given context property identified by its name and namespace or Nothing if it does not exist

4.3 PROPERTY OBJECT

The Property object represents a context property associated with the message. These take the form of a name, namespace, and value. An example property is ReceivedFileName that the FILE and FTP adapters promote. Properties are accessed from the Context object.

Name	Signature	Description
Name	[object].Name	The name of the property
Namespace	[object].Namespace	The namespace of the property
Value	[object].Value	The value of the property. Could be any of the VBScript data types such as Long, String, Array, Boolean etc.

4.4 ENCODING OBJECT

Name	Signature	Description
Type	[object].Type	The encoding used when converting your return message in to bytes. Valid values are utf-7, utf-8, utf-16, ascii The type is set to utf-8 by default

5 PORT CONFIGURATION

When creating a new send port, set the transport type to “Script”. This tells BizTalk to use the Script Adapter.

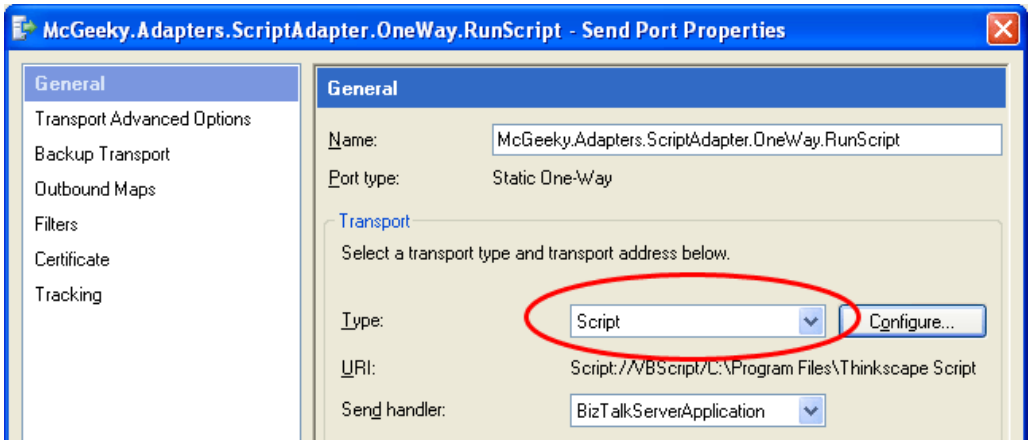

Figure 3

To configure the Script Adapter once the transport type has been selected, click on the Configure button in the send port properties dialog. The Script Adapter supports three properties. All are relevant to both one way and solicit response ports. The configuration property is optional. The language and script property are mandatory.

- ◊ *Language* - the type of language your script is written in. E.g. VBScript or JScript.
- ◊ *Script* - the location of your script file. E.g. c:\myscripts\sendemail.vbs.
- ◊ *Configuration* - the parameter that is passed to the `OnMessage` method for your script. Can be any value you want.

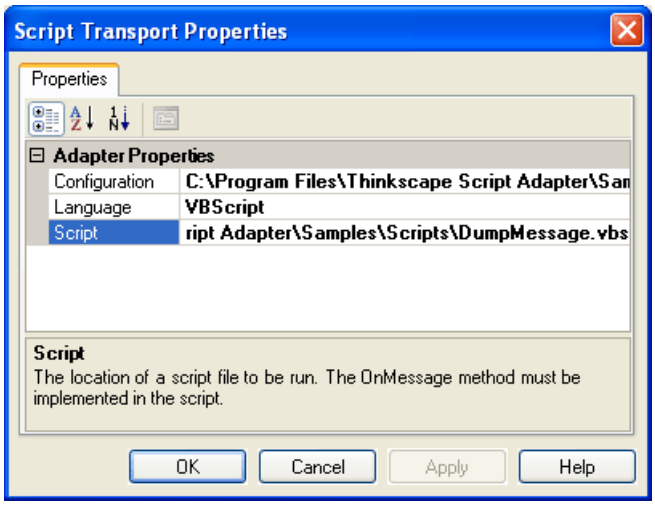

Figure 4

6 SCRIPTING TOOLBOX

There are a number of useful automation objects in the scripting world that can be used in your scripts. Here is a selection of the most useful ones:

6.1 SCRIPTING.FILESYSTEMOBJECT

Used for manipulating files and folders, and for reading and writing files.

```
Dim FileSystem, Folder
Set FileSystem = CreateObject ( "Scripting.FileSystemObject" )
Set Folder = FileSystem.GetFolder ( "c:\temp" )
```

6.2 DOMDOCUMENT

For manipulating XML documents - essential for BizTalk! There are a whole bunch of XML automation objects that come with MSXML. Because BizTalk requires this you should already have it installed.

```
Dim XmlDoc, Order
Set XmlDoc = CreateObject ( "Msxml2.DOMDocument.4.0" )
XmlDoc.LoadXml ( "<root><order id = '1'/></root>" )
Set Order = XmlDoc.SelectSingleNode ( "/root/order" )
```

6.3 ADO

If you want to do any database work then use ADO. See this link to get you started:

http://www.activexperts.com/activmonitor/windowsmanagement/adminscripts/enterprise/databases/

6.4 CDO

Collaboration Data Objects. You can use them for sending email.

```
'Sending a text email with an attached file
set objMessage = CreateObject("CDO.Message")
objMessage.Subject = "Example CDO Message"
objMessage.Sender = "me@my.com"
objMessage.To = "test@paulsadowski.com"
objMessage.TextBody = "This is some sample message text."
objMessage.Send
```

More examples are at http://www.paulsadowski.com/WSH/cdo.htm.

7 SAMPLES

There are sample orchestrations and scripts in the [INSTALLDIR]\Samples folder. The samples demonstrate using the Script Adapter to run scripts.

1) The Request Response orchestration demonstrates a solicit response script. The orchestration receives a message, sends it to the solicit response script that transforms it, the message received back from that script is then sent to a file by the orchestration.

2) The One Way orchestration demonstrates a one way script usage. The orchestration receives a message, sends it to the one way script that dumps the message details to the [INSTALLDIR]\Samples\Ports\Out folder.

The orchestrations are in [INSTALLDIR]\Samples\Orchestrations folder. The scripts themselves (VBScripts) are in the [INSTALLDIR]\Samples\Scripts folder. The folders that the ports are configured to listen to and drop messages in are in the [INSTALLDIR]\Samples\Ports folder.

7.1 BUILDING AND INITIALIZING THE SAMPLES

To build and initialize the samples;

1) In a command window, navigate to the following folder: [INSTALLDIR]\Samples.

2) Run the file Setup.bat, which performs the following actions:

   - Compiles and deploys the Visual Studio 2005 project, containing both orchestrations, in to a new BizTalk application called UsingScriptAdapter.

   - Creates and binds the BizTalk Server receive locations and the send and receive ports in the UsingScriptAdapter application.

7.2 RUNNING THE SAMPLES

1. Copy and paste the [INSTALLDIR]\Samples\Request.xml file in to the folder [INSTALLDIR]\Samples\Ports\In1 to process the solicit response orchestration.

2. After a short time a file ending with .xml will appear in the folder [INSTALLDIR]\Samples\Ports\Out. This is the response message returned by the script TransformMessage.vbs to the orchestration which then writes it out through a FILE send port.

3. Copy and paste the [INSTALLDIR]\Samples\Request.xml file in to the folder [INSTALLDIR]\Samples\Ports\In2 to process the one way orchestration.

4. After a short time a file ending with .txt will appear in the folder [INSTALLDIR]\Samples\Ports\Out. This is a “dump” file created by the script DumpMessage.vbs. Open up the file and you should see the context properties of the message as well as the actual message body itself.

If the files do not appear then check the application event log for errors. BizTalk will suspend the messages & orchestrations just as it would with any other.

7.3 UNINSTALLING THE SAMPLES

To uninstall the samples sample run the batch file Cleanup.bat in the [INSTALLDIR]\Samples folder.

8 TESTING YOUR SCRIPTS

The Script Adapter doesn't provide helpful error messages when your script goes wrong. This is not the Script Adapter's fault - it’s the way the script control works. You'll get an error message in the event log stating say, that there is an invalid cast in your script but it won't indicate at which line it occurred. This can be annoying to debug. Therefore, a script testing framework is also included with the Script Adapter. You can test your scripts outside of BizTalk with the test framework.

The test framework and sample test files are included in the [INSTALLDIR]\Samples\Tests folder.

There is a script file called Model.vbs in the test framework. This contains class definitions mirroring the object model that the Script Adapter creates. With these classes it is now possible to call the OnMessage procedure outside of the Script Adapter.

In the samples folder there are some test scripts. These end with the extension wsf. Wsf files are a special kind of file that enables a script to include other script files.

Try out one of the tests. Open up a command prompt on the [INSTALLDIR]\Samples\Tests folder and type the following: cscript DumpMessageTest.wsf

A file called 1234567890.txt is written in to the [INSTALLDIR]\Samples\Ports\Out folder.

To implement a test script of your own:

1. Start by copying one of the samples and renaming it
2. At the top of the file are two include statements. Keep the one that imports Model.vbs whilst changing the other to reference your script file
3. Change the script in the wsf file to build up test data
4. The very last statement in the wsf file is a call to `OnMessage`
5. Save the changes and run the wsf file from the command line as follows to test your script:
   `cscript MyScriptTest.wsf`

Note that the Context.Read procedure was not implemented in the test framework. So if you make use of it and want to test your script you'll have to figure something out yourself. Sorry!

9 USING THE SCRIPT ADAPTER WITHOUT ORCHESTRATIONS

You don't have to use orchestrations with the Script Adapter. You can use both one way and solicit response send ports without an orchestration. For example; you can create a one way send port using the Script Adapter to call your VBScript of choice that is bound to a FILE receive location directly. Anytime a file is received in your receive location your script will be called. The send port is filtered on the name of the receive port using the standard BizTalk filtering feature.

It is possible to use a solicit response Script Adapter send port that is not bound to an orchestration and have other ports respond to the returned message. You could implement a load routine in your script and return a message saying all was okay, for example. First you have to deploy the message schema in an assembly. BizTalk will promote the MessageType property of the schema. It's possible that you could implement your own pipeline to promote other values. Now create a Script Adapter solicit response send port filtered on the MessageType. You can now send a message to your solicit response, have it load data, return an event, and have something listen for that event - all without an orchestration.

10 PERFORMANCE

There is no doubt that an adapter implemented with scripts is going to be slower than its compiled cousin.

In this release, the Script Adapter makes no attempt to cache the scripts between script executions. They are loaded each time.

11 ARCHITECTURE

The Script Adapter uses the templates created by the Adapter Wizard http://bvdzwan.nl/blog/index.php?/archives/1-Adapter-wizard-for-BizTalk-Server-2006.html.

The Windows script control is integrated using .Net's support for COM. The Script Adapter uses the Interop.MSScriptControl.dll library to execute the scripts.

Before the OnMessage procedure of the script is invoked, the Script Adapter creates an object model that is passed to the script. This is a very powerful feature of the script control - any .Net object can be automated from a script!

12 KNOWN ISSUES

At the moment there are some points that can be improved in future:

1. The Script Adapter loads the entire message body in to a string whether or not the script requires it (whether or not the script calls the Message.Text property).
2. The Script Adapter loads in the entire message body in to a string without using streaming. So, the larger the message the more memory will be consumed.
3. The adapter does not cache a script. Each time the script is executed it is loaded in to the scripting control.
4. A script must be in the one script file (it can't include other scripts).