Using remote objects

Working with function libraries can become quite tedious, as each tes must have them associated. Deployment issues may arise when tes are copied to other machines. For instance, resources may be missir due to misconfiguration (for example, undefined search paths).

If we could store the code on a server, so that associating function libraries would not be necessary, we would gain three main benefits:

- Code maintenance and deployment would be simplified and her become more efficient
- The dependency of tests on the association of function libraries would vanish
- Tests would run on any machine that has Internet connectivity thus reach the code server

In this recipe, we will examine a clever way to accomplish this.

Getting ready

From the **File** menu, navigate to **New** | **Test**, or use the *Ctrl + N* shortcut. Create a new file named `CROWrapper.wsc` (`C` stands for class and `RO` for remote object).

> **Note**
>
> A **Windows Script Component** (**WSC**) file is actually a special XML file that can store components and packages written in various languages that are supported by the **Windows Script Host** (**WSH**).

How to do it...

Write the following code in the component (`wsc`) file:

```
<?xml version="1.0"?>
-<component id="CROWrapper">
<?component error="true" debug="true"?>
 <registration classid="{D11841E9-B794-4627-AEE6-CA552DFF11C8}" version="
<![CDATA[ Function PrintMe(str) msgbox "In WSC file!!! - " & str End Func
 </script> </component>
```

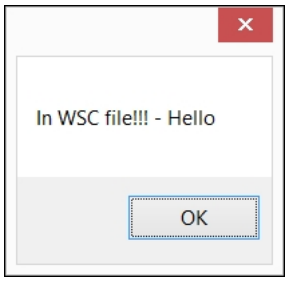
Store the file on a remote machine (upload to a server). For the purpose of this demonstration, the file can be found at:

http://www.advancedqtp.com/COM/CROWrapper.wsc
(http://www.advancedqtp.com/COM/CROWrapper.wsc)

Now, write the following code in your test:

```
Const rootURL = "script:http://www.advancedqtp.com/COM/"

Dim oROWrapper

Set oROWrapper = GetObject(rootURL&"CROWrapper.wsc")

Call oROWrapper.PrintMe("Hello")

Set oROWrapper = Nothing
```

Run the test. A message box like the following should appear:



How it works…

Basically, a component defined in a `wsc` file is equivalent to a class. The statement

```
Set oROWrapper = GetObject(rootURL&""CROWrapper.wsc"")
```

actually instantiates the component, so we now have an object of type `CROWrapper`. As such, we can call its public methods, as we would with any object. Note that, as mentioned previously, this means that we can actually use code stored in a server without having to associate it with the test.

Recommended / Queue
Feedback (http://community
PREV Implementing a c…
NEXT Utility statements