WSH VBScript WMI FSO
ADSI CDO HTA CGI Perl

300165
Systems Administration Programming

```
set objWMI = GetObject("winmgmts:\\.\root\cimv2")
set fso = CreateObject("Scripting.FileSystemObject")
```

Lecture 4
Objects in VBScript

Pring page

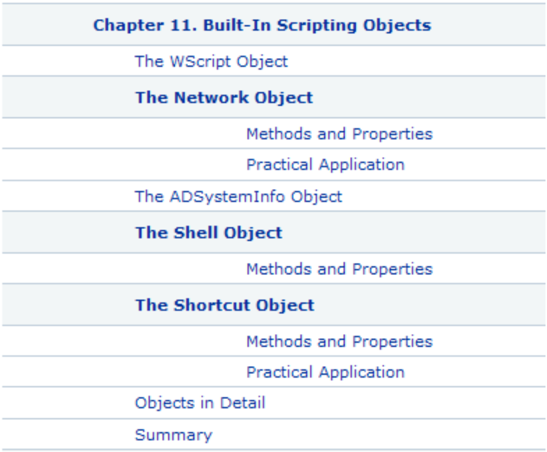
VBScript is a kind of Object-Oriented programming language. However, we do not write any class definition in VBScript. Instead, we use the objects that are provided by Windows Script Host Object Model. In this lecture, we will discuss how to use these objects in a script.

Key words

WSH, WSH Object, WSH Object Model

Reference to textbook chapters

The lecture covers Chapter 11 of the textbook. (Don Jones, VBScript, WMI and ADSI unleashed : using VBSscript, WMI, and ADSI to automate Windows administration eBook: Chapter 11. Built-In Scripting Objects).



The `WScript` Object

`WScript` is the root object of the Windows Script Host (WSH). You can use all its properties and methods directly, for instance `WScript.echo`. The WScript object allows you to:

- create objects via CreateObject Method
- connect to objects via ConnectObject Method
- disconnect from objects via DisconnectObject Method
- sync events via Sleep Method
- stop a script's execution programmatically via Quit Method
- output information to the default output device via Echo Method

Follow the above links to learn how to use these methods. We have known how to use WScript.echo for message output. We have also used `CreateObject` method to create `WScript.network` object in last lecture. Here is another example to use WScript object.

```
WScript.echo "Wait me for 5 minutes"
WScript.Sleep 300000
WScript.echo "I am ready now, let's go"
WScript.Quit
```

Copy and paste this code and run it on your computer.

The `Shell` Object

The Shell object is used to execute external applications, work with special folders and shortcuts, manipulate environment variables, write to the event log, read and write to the Registry, create timed dialog boxes, and even send keystrokes to another application.

Just like the Network object, the Shell object must be explicitly created and assigned to a variable.

```
'Create shell object
Set objShell = CreateObject("WScript.Shell")
```

WSH Shell `run` method

WSH Shell is so powerful that it can do whatever you can do with command line. In fact, WSH Shell object allows you to execute applications within your script. There are two shell methods to execute applications: `Exec` and `Run`.

Let's try the script dirRun.vbs first:

```
Dim objShell
Set objShell = CreateObject("WScript.Shell")
objShell.Run "cmd /c dir > test.txt"
```

where “cmd /c dir > test.txt” is a command that lists the current directory and output the result to a text file (the option "/C" means "Carries out the command specified by string and then terminates").

The Run method can take two more parameters in addition to the specified command. The following code runNotepad.vbs starts the notepad.exe program, and waits for you to complete and display a message:

```
'Create shell object
Dim objShell
Set objShell = CreateObject("WScript.Shell")

'call Notepad program
objShell.Run "notepad.exe",1,true
MsgBox "I know what you wrote :-)"
```

Read Microsoft VBScript WSH reference for more details.

WSH Shell `Exec` method

The Exec method is more advanced. It returns a WSHScriptExec object, which wraps the standard stream objects `stdIn`, `StdOut`, and `StdErr` services. This enables the script to read the output directly from the application without first having to save the output to a file.

The following script execPing.vbs illustrates how to use Shell Exec method to check network using command line Ping:

```
WScript.Echo("Check Network using command line Ping")
Set objShell = CreateObject("WScript.Shell")
Set objWshScriptExec = objShell.Exec("cmd /c ping 127.0.0.1")

Set objStdOut = objWshScriptExec.StdOut
While Not objStdOut.AtEndOfStream
    strLine = objStdOut.ReadLine
    strOutput=strOutput & strLine & vbcrlf
Wend
WScript.Echo(strOutput)
```

In this code, the `Exec` method returns a `WSHScriptExec` object, named `objWshScriptExec`, which outputs the outcome of the command line to a standard stream object. The loop simply reads each line from the standard stream and puts it into a string. Note that `vbcrlf` is a VB constant for a new line (like \n in C++).

WSH Shell `SendKeys` method

You can use the SendKeys method to type keystrokes to applications that have no automation interface. Most keyboard characters are represented by a single keystroke. Some keyboard characters are made up of combinations of keystrokes (CTRL+SHIFT+HOME, for example). To send a single keyboard character, send the character itself as the string argument. For example, to send the letter x, send the string argument "x".

Run the following script for fun:

```
Dim objShell
Set objShell = CreateObject("WScript.Shell")

objShell.Run "notepad.exe"
WScript.Sleep 5000
objShell.AppActivate "Notepad"
objShell.SendKeys "Ghost writing is fun."
```

Learn more about SendKeys method from Microsoft VBScript WSH reference. Practice with this code calculator.vbs

Learn more about WSH Shell methods from Microsft VBScrip WSH reference.

The Shortcut object

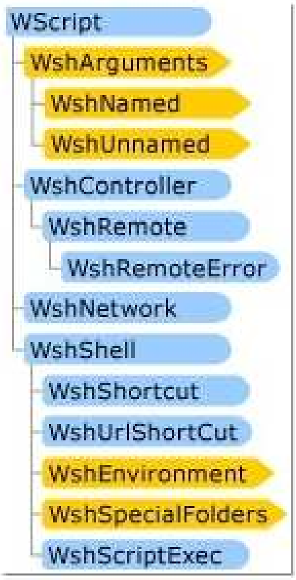
Shortcut object allows you to create a shortcut programmatically. Learn how to use the object by using the following script shortcut.vbs.

```
set WshShell = WScript.CreateObject("WScript.Shell")
strDesktop = WshShell.SpecialFolders("Desktop")

set oShellLink = WshShell.CreateShortcut(strDesktop & "\IE.lnk")
with oShellLink
.TargetPath = "C:\Program Files\Internet Explorer\IEXPLORE.EXE"
.WindowStyle = 1
.Hotkey = "Ctrl+Alt+I"
    .IconLocation = "C:\Program Files\Internet Explorer\IEXPLORE.EXE, 0"
    .Description = "Internet Explorer"
.WorkingDirectory = strDesktop
.Save
end with

set oUrlLink = WshShell.CreateShortcut(strDesktop & "\Script Programming.url")
oUrlLink.TargetPath = "http://www.scm.uws.edu.au/~jianhua/SAP10/"
oUrlLink.Save
```

The script firstly creates a WSH Shell object and calls its method `SpecialFolders` to get the actual path of your Desktop folder (different user has difference path). The second part creates a shortcut for Internet Explorer and puts it on your Desktop. The third part of the code creates a shortcut for the unit webpage of Systems Administration Programming.

Summary

In this lecture, we learnt how to use WSH built-in objects, such as WScript, Shell and Shortcut. In fact, the Windows Script Host object model provides a logical, systematic way to perform many administrative tasks. The following is the hierarchy of WSH objects:



Learn more about WSH objects from Microsoft website. Microsoft's Philosophy: Create a minimal core WSH structure that can interact with any number of separate object models for extention. For instance, Windows Management Instrumentation (WMI, See Lecture 5), the FileSystemObject (FSO, See Lecture 6), Active Directory Service Interfaces (ADSI, See Lecture 7), Collaboration Data Objects (CDO, See Lecture 8) object models are separate from the WSH object model (i.e. WMI, FSO, ADSI, and CDO are not built in WSH).

What's the time now? Have a kit-kat. ...