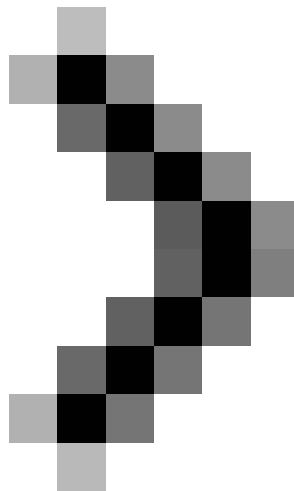
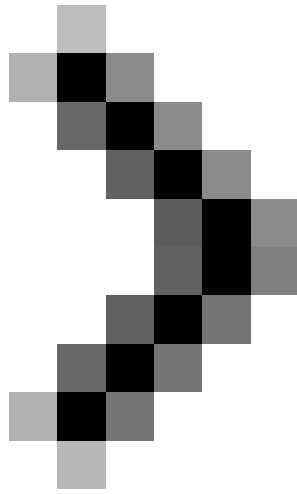


Managing Windows networks using Script - Part 13: The script returns all values

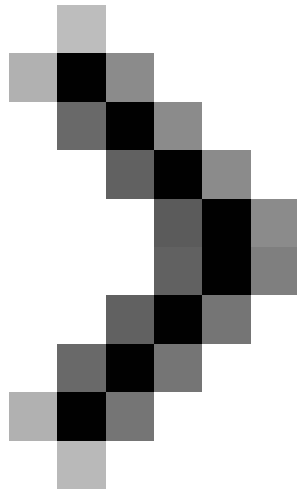
In the previous article of this series we came up with a script called DisplayClassProperties.vbs, which shows the names of the properties of the WMI class. This is what the script content is, by using Win32_BootConfiguration as a class, we are connecting to the WMI nickname.



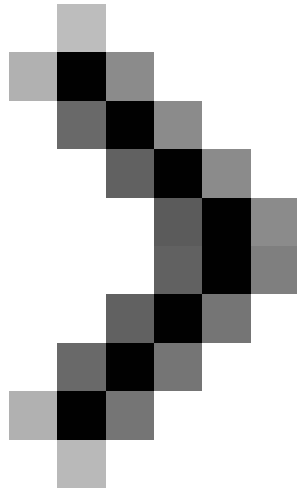
Part 1: Basic concepts



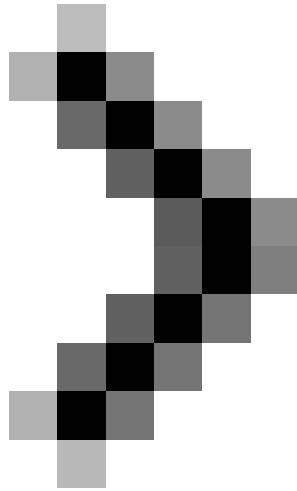
Part 2: Complete the script



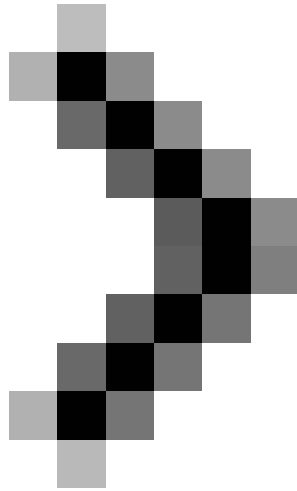
Part 3: Understanding WMI



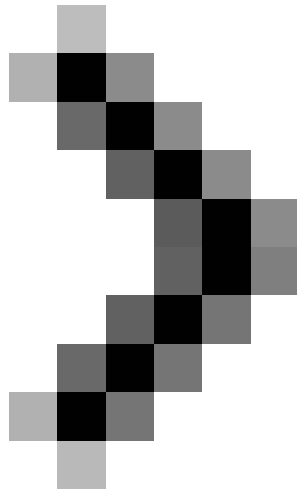
Part 4: Use Win32_NetworkAdapterConfiguration



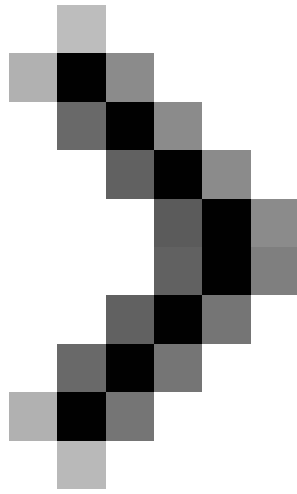
Part 5: Overcoming challenges



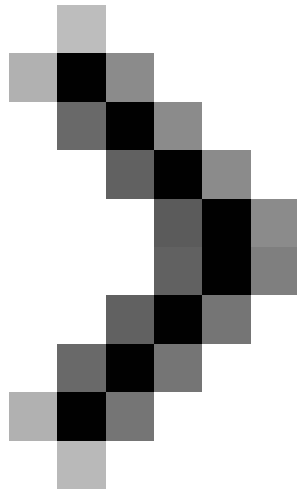
Part 6: The first steps for remote scripting



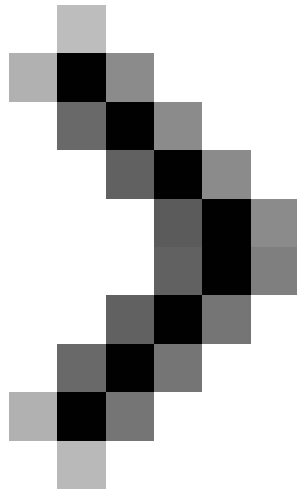
Part 7: Troubleshooting errors



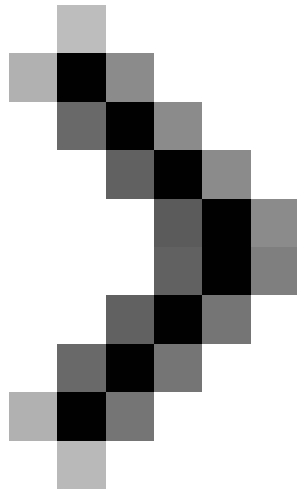
Part 8: Remote script error handling with Network Monitor 3.0



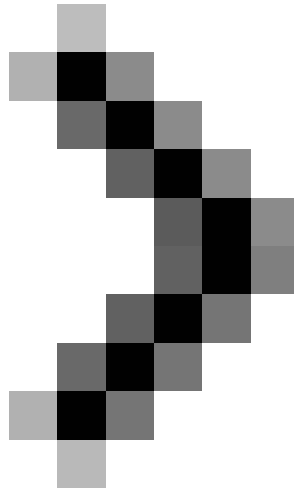
Part 9: Understanding remote control scenarios



Part 10: Tricks of remote control scenarios



Part 11: Other script tricks



Part 12: Properties of the WMI class

Mitch Tulloch

In the previous article of this series we came up with a script called `DisplayClassProperties.vbs`, which shows the names of the properties of the WMI class. This is what the script content is, by using `Win32_BootConfiguration` as a class, we are connecting to the WMI nickname :

```
Option Explicit
On Error Resume Next
Dim strComputer
Dim strWMINamespace
Dim strWMIQuery
Dim objWMIService
Dim colItems
Dim objItem
strComputer = "."
strWMINamespace = "root\CIMV2"
strWMIQuery = ": Win32_BootConfiguration"
Set objWMIService = GetObject ("winmgmts:" & strComputer & strWMINamespace & strWMIQuery)
WScript.Echo "Number of properties of" & strWMIQuery & "class is" &
objWMIService.Properties_.count
For Each objItem in objWMIService.Properties_
Wscript.Echo "Property:" & objItem.name
```

next

When running this script (using local administrative information) on a Windows XP computer (with Cscript.exe previously configured as a default Windows script), the results are as follows:

```
C: scripts>DisplayClassProperties.vbs  
Microsoft (R) Windows Script Host Version 5.6  
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.  
Number of properties of: Win32_BootConfiguration class is 9  
Property: BootDirectory  
Property: Caption  
Property: ConfigurationPath  
Property: Description  
Property: LastDrive  
Property: Name  
Property: ScratchDirectory  
Property: SettingID  
Property: TempDirectory
```

As mentioned in the last section, this script can easily be customized to display the attribute name of any WMI class. For example, if you want to display all property names in the Win32_DiskPartition class, we just need to change the line:

```
strWMIQuery = ": Win32_BootConfiguration"
```

into:

```
strWMIQuery = ": Win32_DiskPartition"
```

When running the script again, the result will be:

```
C: scripts>DisplayClassProperties.vbs  
Microsoft (R) Windows Script Host Version 5.6  
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.  
Number of properties of: Win32_DiskPartition class is 34  
Property: Access  
Property: Availability  
Property: BlockSize  
Property: Bootable  
Property: BootPartition  
Property: Caption  
Property: ConfigManagerErrorCode  
Property: ConfigManagerUserConfig  
Property: CreationClassName  
Property: Description  
Property: DeviceID
```

Property: DiskIndex
Property: ErrorCleared
Property: ErrorDescription
Property: ErrorMethodology
Property: HiddenSectors
Property: Index
Property: InstallDate
Property: LastErrorCode
Property: Name
Property: NumberOfBlocks
Property: PNPDeviceID
Property: PowerManagementCapabilities
Property: PowerManagementSupported
Property: PrimaryPartition
Property: Purpose
Property: RewritePartition
Property: Size
Property: StartingOffset
Property: Status
Property: StatusInfo
Property: SystemCreationClassName
Property: SystemName
Property: Type

Displays the values ??of each attribute

Here we go back to using Win32_BootConfiguration as a class, if changed so that the script will list not only the names of all properties but also their values, you only need to change the line:

```
Wscript.Echo "Property:" & objItem.name
```

into:

```
Wscript.Echo "Property:" & objItem.name & vbTab & "Value:" & objItem.value
```

Results when running:

```
C: scripts>DisplayClassProperties.vbs  
Microsoft (R) Windows Script Host Version 5.6  
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.  
Number of properties of: Win32_BootConfiguration class is 9  
Property: BootDirectory Value:  
Property: Caption Value:  
Property: ConfigurationPath Value:  
Property: Description Value:  
Property: LastDrive Value:
```

Property: Name Value:
Property: ScratchDirectory Value:
Property: SettingID Value:
Property: TempDirectory Value:

The results are all empty. Why is that?

This is what is going on, take a look at this line:

```
Set objWMIService = GetObject ("winmgmts:" & strComputer & strWMINameSpace & strWMIQuery)
```

Including the values ??of each variable we can record this line as follows:

```
Set objWMIService = GetObject ("winmgmts: .rootCIMV2: Win32_BootConfiguration")
```

Note that we are connecting to a specific WMI class (Win32_BootConfiguration) in the WMI nickname to be able to return a set of properties of this class. We then want to display the name and value of each attribute. But the values ??returned NULL are because they are not connected to the specific *instance* of this class. WMI Glossary says that an *instance* is 'an *instance* of the real world that has managed an object with a specific class' and '*instances* include real values' and this real value is what we want. So how can you connect to an instance of a class?

To connect to an instance of a class, you need to specify a separate instance by using the *key property* of the class. We can see that the *key property is an attribute that provides the identifier needed for an instance of the class and the key properties are marked with a Key* qualifier in the MSDN documentation. Take a look at the MSDN page about configuring the Win32_BootConfiguration class to know more about the key property for this class. Figure 1 shows a section of this page, showing you how to distinguish the key property of the class:

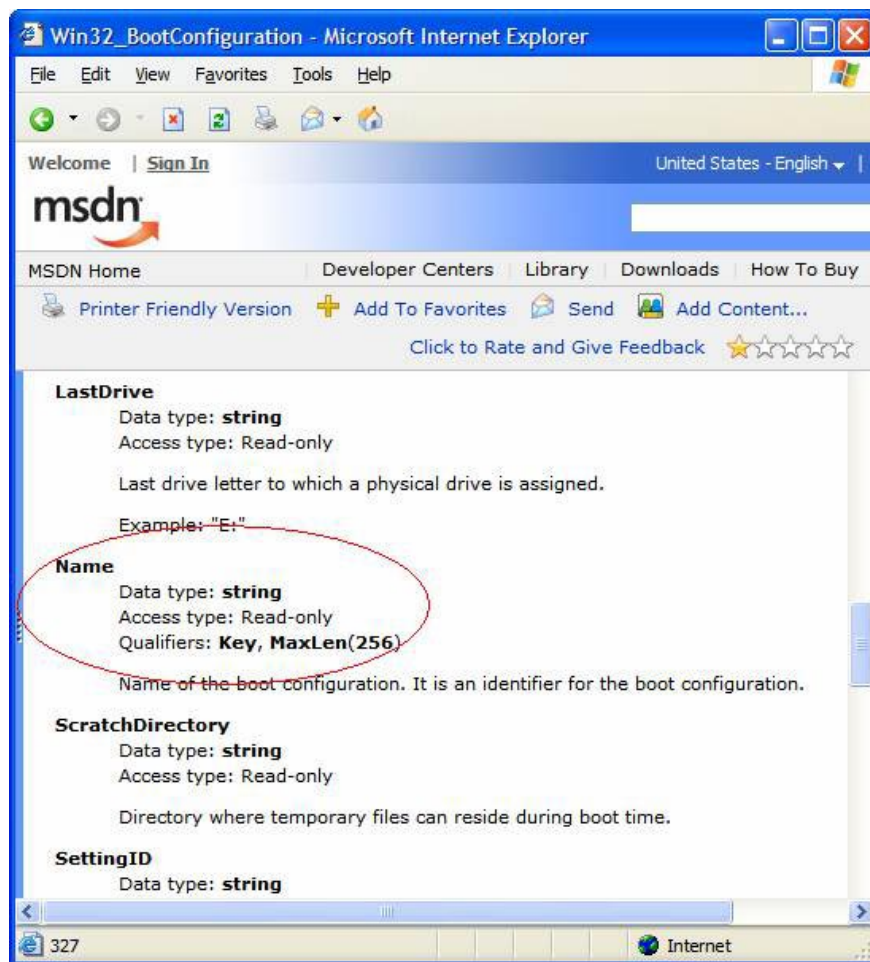


Figure 1: Key property for Win32_BootConfiguring class

From this MSDN page, we can see that the key property for the Win32_BootConfiguration class is Name. This means that we need to specify a value for this attribute in the WMI alias string if we want to connect to a specific instance of the class to get the values ??of each class attribute. In other words, all you need to do is change the line:

```
strWMIQuery = ": Win32_BootConfiguration"
```

into:

```
strWMIQuery = ": Win32_BootConfiguration.Name = 'SOMETHING'"
```

At 'SOMETHING' is the attribute value name of a specific instance in the class.

How can we find the value of the key property of a specific instance of the class? One way is to use Windows Management Instrumentation Tester (wbemtest.exe). Start by typing **wbemtest** at the command window so you can open the window as follows:

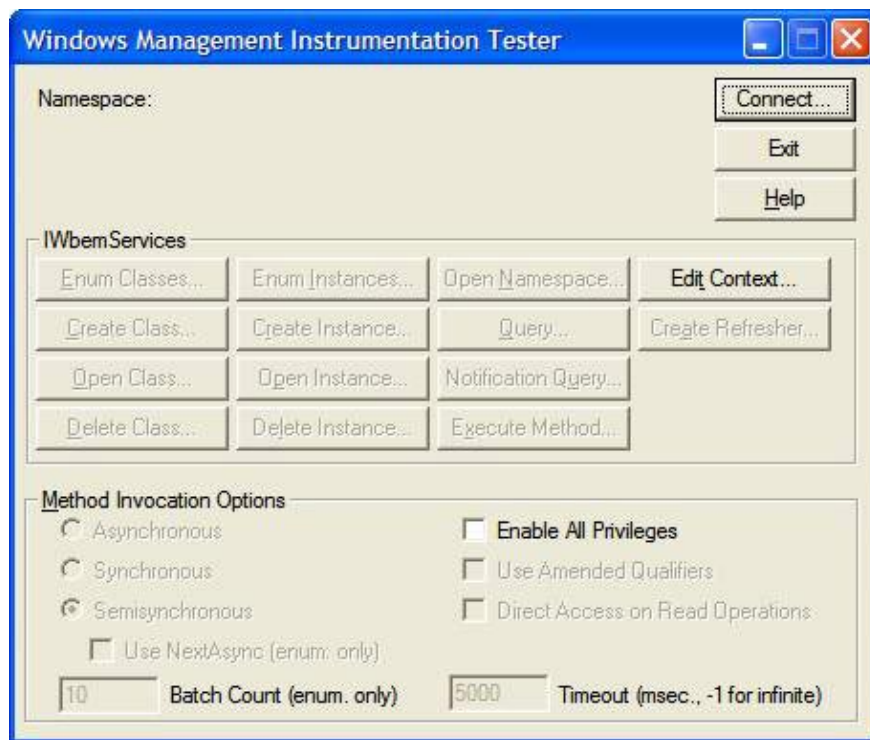
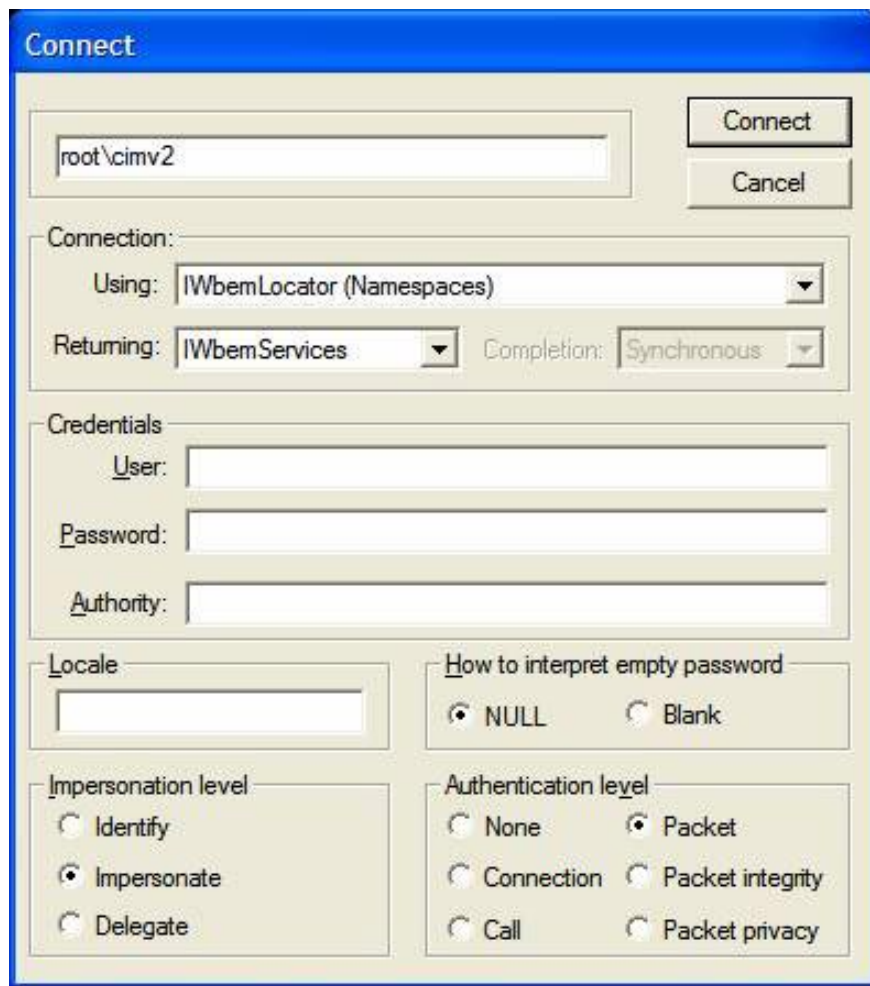


Figure 2: Windows Management Instrumentation Tester

Click the Connect button and connect to the rootcimv2 namespace:



The image shows a 'Connect' dialog box with a blue title bar. At the top left, there is a text box containing 'root\cimv2'. To its right are 'Connect' and 'Cancel' buttons. Below this is a 'Connection:' section with a dropdown menu set to 'IWbemLocator (Namespaces)'. Underneath are two more dropdowns: 'Returning:' set to 'IWbemServices' and 'Completion:' set to 'Synchronous'. The 'Credentials' section has three text boxes labeled 'User:', 'Password:', and 'Authority:'. Below that is a 'Locale' text box. To the right of the locale box is a section titled 'How to interpret empty password' with two radio buttons: 'NULL' (selected) and 'Blank'. At the bottom left is an 'Impersonation level' section with three radio buttons: 'Identify', 'Impersonate' (selected), and 'Delegate'. At the bottom right is an 'Authentication level' section with six radio buttons: 'None', 'Packet' (selected), 'Connection', 'Packet integrity', 'Call', and 'Packet privacy'.

Figure 3: Connect to the Win32_BootConfiguration class

Click Connect to return to the main window, where all the buttons are displayed here:

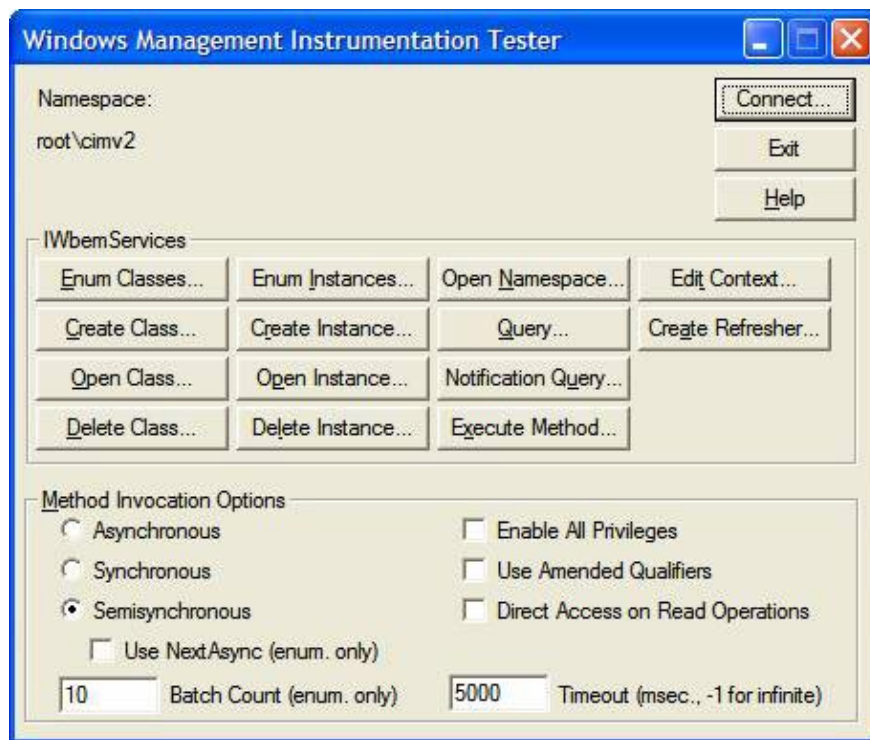


Figure 4: Connected to class

Click the Enum Instances button and type the class name to display all instances of the class:

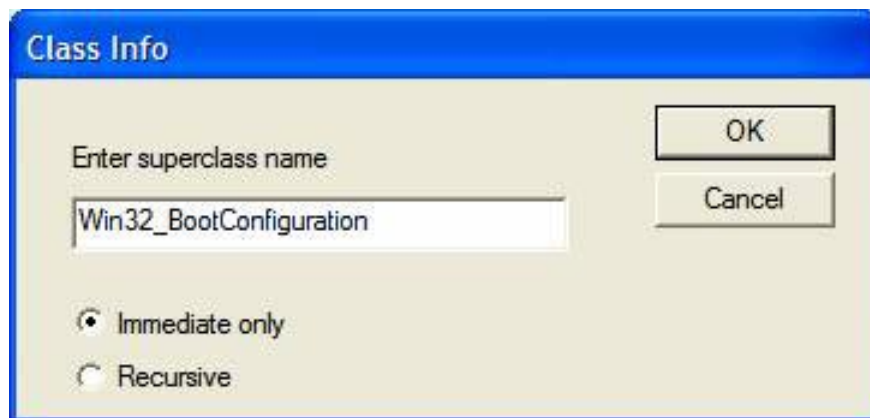


Figure 5: Displaying instances of the class

Finally, click OK to display all instances of the class as listed by their key property (Name):

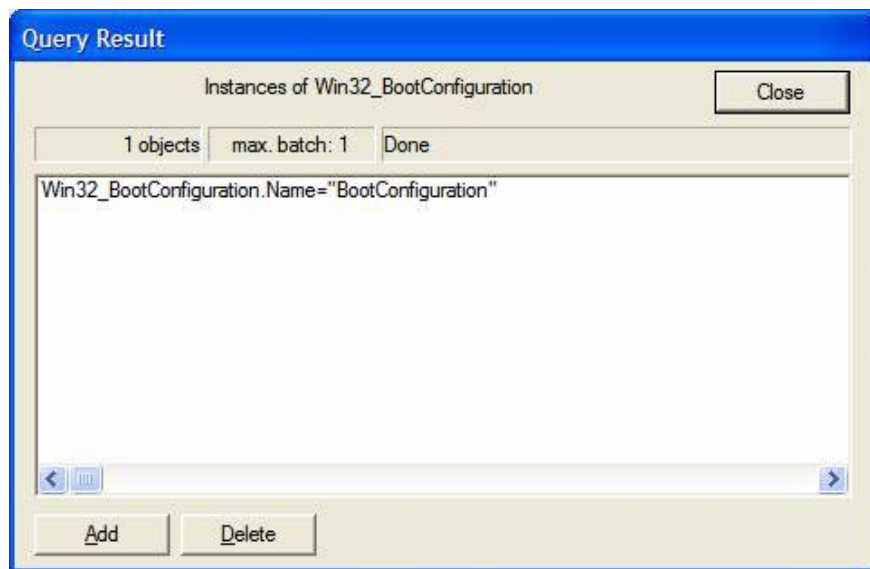


Figure 6: Instances of Win32_BootConfiguration

After all has been removed, there is only one instance of this class on the computer and the Name attribute of this instance has the value "BootConfiguration"! this means that to display the values ??of the properties of the instance of the Win32_BootConfiguration class on our computer, all that's needed is to change this line:

```
strWMIQuery = " : Win32_BootConfiguration"
```

into:

```
strWMIQuery = " : Win32_BootConfiguration.Name = 'BootConfiguration'"
```

Alternatively, the DisplayClassProperties.vbs script is reviewed now as follows:

```
Option Explicit
On Error Resume Next
Dim strComputer
Dim strWMINamespace
Dim strWMIQuery
Dim objWMIService
Dim colItems
Dim objItem
strComputer = "."
strWMINamespace = "rootCIMV2"
strWMIQuery = " : Win32_BootConfiguration.Name = 'BootConfiguration'"
Set objWMIService = GetObject ("winmgmts:" & strComputer & strWMINamespace & strWMIQuery)
WScript.Echo "Number of properties of" & strWMIQuery & "class is" &
objWMIService.Properties_.count
For Each objItem in objWMIService.Properties_
Wscript.Echo "Property:" & objItem.name & vbTab & "Value:" & objItem.value
```

next

When running this script, it will not only display the names of the properties but also their values:

```
C: scripts>DisplayClassProperties.vbs  
Microsoft (R) Windows Script Host Version 5.6  
Copyright (C) Microsoft Corporation 1996-2001. All rights reserved.  
Number of properties of: Win32_BootConfiguration.Name = 'BootConfiguration' class  
is 9  
Property: BootDirectory Value: WINDOWS  
Property: Caption Value: DeviceHarddisk0Partition1  
Property: ConfigurationPath Value: WINDOWS  
Property: Description Value: DeviceHarddisk0Partition1  
Property: LastDrive Value: C:  
Property: Name Value: BootConfiguration  
Property: ScratchDirectory Value: C: WINDOWSsystem32configssystemprofileLocal SettingsTemp  
Property: SettingID Value:  
Property: TempDirectory Value: C: WINDOWSsystem32configssystemprofileLocal SettingsTemp
```

Put this information in the table for easier reading:

BootDirectory

WINDOWS

Caption

DeviceHarddisk0Partition1

ConfigurationPath

WINDOWS

description

DeviceHarddisk0Partition1

LastDrive

C:

Name

BootConfiguration

ScratchDirectory

C: WINDOWSsystem32configssystemprofileLocal SettingsTemp

SettingID

Value:

TempDirectory

C: WINDOWSsystem32configsystemprofileLocal SettingsTemp

Conclude

We can see that this simple 'return all values' script gave us useful information about the computer. Here's an exercise you can do yourself: instead of connecting to an instance of the Win32_BootConfiguration class (there is only one instance of this class), try to connect to an instance of the Win32_DiskPartition class (the class has a few instance if your computer has multiple partitions. To do that, you first need to use wbemtest to display instances of this class (learn more about the other key property with these instances) and then change the DisplayClassProperties.vbs script so it can display attributes and values ??of a specific instance of this class (ie the disk partition you specified).

% Over %

You finished reading the article "**Managing Windows networks using Script - Part 13: The script returns all values**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.