

February 2008

Caution trying to get FolderSize information w/Powershell...

February 7, 2008

Make sure to roughly test all Powershell script you may download from the Internet (even the ones I'm putting in my blog). I just found a couple of sample that will help display a list folder size (which Windows Explorer won't provide). Unfortunately the results displayed wasn't perfect. I had to use Windows Explorer to verify size information by doing the right-click on the folder to compare the results. Is this a bug? or just a mistake when creating complex cmdlet.

The problem is that, in most cases, the folders listed are fine. I only caught this issue when listings folders that contains lots of subfolders.

Here are some Powershell samples:

1. Powershell (PS) script code from the Internet: (building a function)

```
Function getFolderSize{
$items = Get-Childitem $args
foreach($item in $items){
if ($item -is [System.IO.DirectoryInfo]){
Add-member -inputobject $item -membertype ScriptProperty -Name Length -Value{$a = `
get-childitem $this.FullName -include *.* -recurse | measure-object -sum Length;
If($a -eq $null){return 0;} else {return $a.sum}}
write-output $item
}
}
}
```

This code work but still doesn't give you 100% correct information(maybe about 80%).

2. Another PS script I retouched from another source: (building a function)

```
Function DisplayFolderSize {
$folderTree=get-childitem $args;
[double] $fsize = 0;
[double] $fsz = 0;
[double] $TotalFSZ = 0;
Write-Host "";
Write-Host "Get Foldersize for $Args";
write-host "-----";
foreach ($folt in $folderTree) {
if($folt.mode -match "d") {
$fsz=((get-childitem $folt.fullname -recurse | Measure-object length -sum).sum);
$fsize="{0:N2}" -f $fsz;
write-host (" $fsize `t`t`t $folt ");
[string] $FolderRecords = $FolderRecords + $folt.name + ", " + $fsize + "`r`n";
$TotalFSZ=$TotalFSZ+$fsize;
}
}
write-host "-----";
write-host "Total Foldersize";
write-host $TotalFSZ;
write-host "";
}
```

This code gave me better result but still not a 100% correct information (maybe 90%).

3. Now, here's the winner. I created this PS script using the "Scripting.FileSystemObject" COM object. This way I can create a collection of members for the folder I'm querying the information. (notice this is not a function)
The following lines you can execute line at a time or just create a PS script file or create a function.

```
#####  
## objFolderSize.ps1 ##  
## Author: Max Trinidad, 02/07/2008 ##  
#####  
  
$FolderName = Read-Host -Prompt "Enter Foldername: "  
$FileSystem = new-object -com "Scripting.FileSystemObject";  
$getFolderAttr = $FileSystem.GetFolder($FolderName);  
$getSubFolderAttr = $getFolderAttr.SubFolders  
$getSubFolderAttr | select name, size | ft -auto  
  
## end of script## This code gave me a 100% correct information.
```

I did gather the results of all three PS scripts and copy/paste them into Excel. Then I verify the folder size result against Windows Explorer making the #3 PS Script the winner.

Did you notice how many lines of code I use in PS script #3 ? :)

Happy coding!!