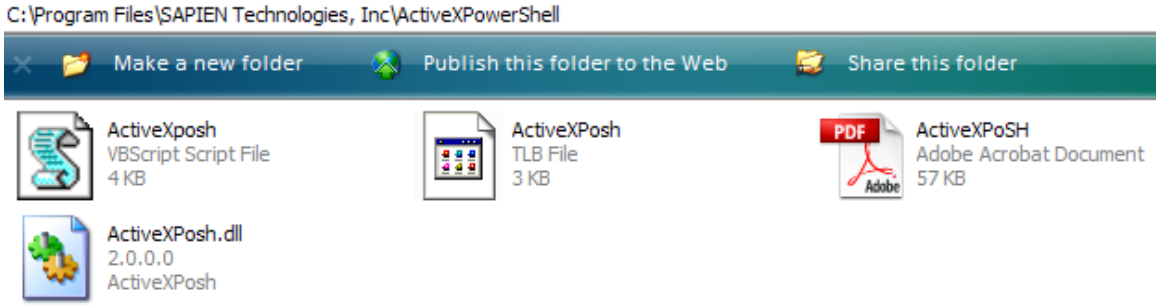
Winbatch -> Powershell via COM

I have long been interested in extending Winbatch with features from .NET, in particular a datagrid to display SQL query results. And while Winbatch cannot communicate with .NET directly, Powershell can, and to my surprise Sapien Technologies released a free ActiveX dll which connects the dots.

The dll can be downloaded from http://www.sapien.com/software/primalscript. You have to register and provide them an email address but then you can download their control which installs from an .msi. I installed it in the default

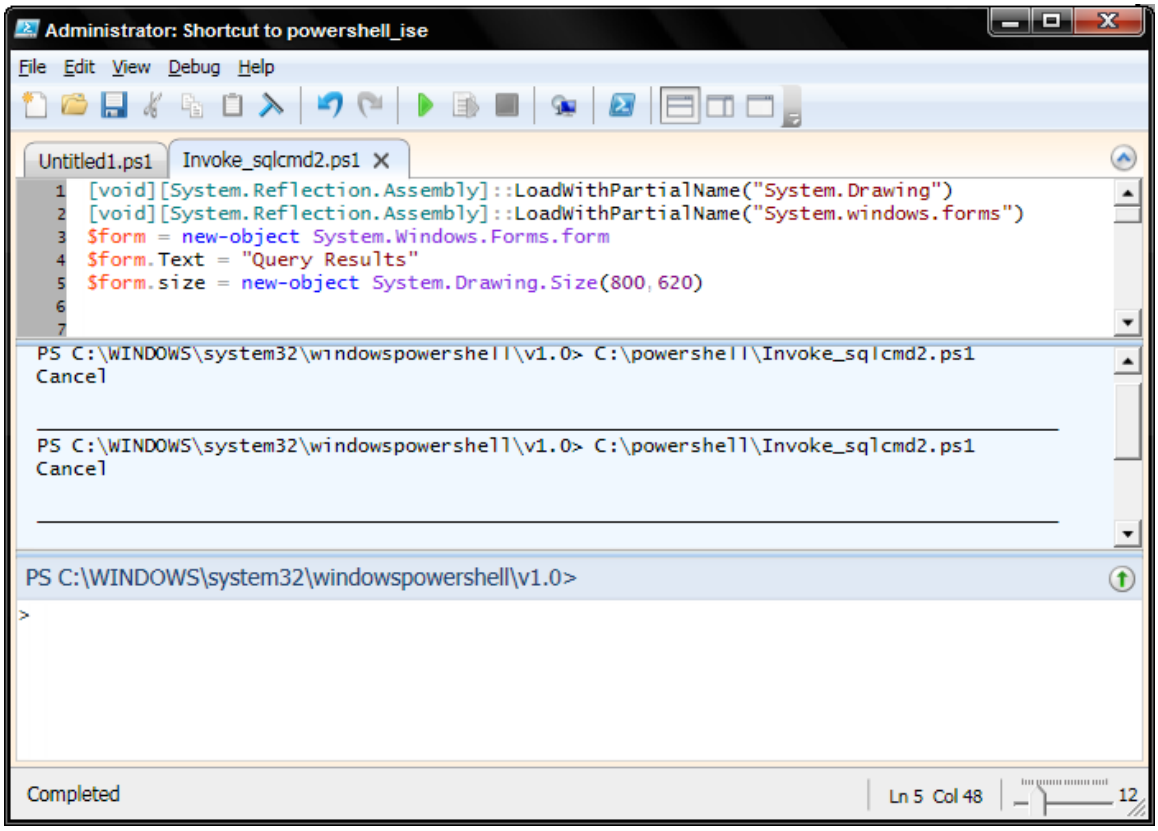


The PDF briefly explains the few methods and properties as the purpose of the control is making Powershell cmdlets and scripts callable from any COM-enabled scripting language.

Very little will be said about writing Powershell cmdlets the good news being there is an abundance of them on the Internet which covers SQL, WMI, AD, Office and pretty much anything you currently do in Winbatch. Version 3.0 has been released for Windows 8, but works on Win7 and Vista (sorry XP) with cmdlets for JSON and other enhancements.

The script I will be discussing works with Powershell 2.0 and has been tested on both XP and Win7. Powershell comes installed on the latter, and you may have to install it if you have XP. If you only have 1.0 installed, I would recommend installing 2.0.

In addition to the Powershell cmd shell, it has nice GUI (Powershell_ise) which is a convenient way to test or develop scripts (and it is free to download or comes with Win7):



I used the GUI to ensure the Winbatch script would execute the Powershell correctly.

To sum up: You will need Winbatch, Powershell 2.0 and the free ActiveX control [registered] to continue. The Winbatch script is specific to sending a query from a SQL Server Express Table into a .NET datagrid via a Powershell template. Might seem confusing but it is the tip of an enormous iceberg of potential for accessing .NET from Winbatch.

O.K…. let's Begin

The goal: Query a Table in SQL Server Express 2008 R2 and display the results in a .NET datagrid.

The Query: select * from USZip where state='FL' and the results:

Florida Zipcodes

Zip	Latitude	Longitude	City	Alt_City	State	County
32087	+30.29103	-082.33401	SANDERSON		FL	Baker
32091	+29.92814	-082.17965	STARKE		FL	Bradford
32192	+29.32682	-082.10463	SPARR		FL	Marion
32193	+29.46362	-081.64052	WELAKA		FL	Putnam
32195	+29.00614	-081.87785	WEIRSDALE		FL	Marion
32198	+29.02272	-081.17216	DAYTONA BEACH	FL REG LIB BL|F...	FL	Volusia
32833	+28.50877	-081.07033	ORLANDO	UNION PARK|AL...	FL	Orange
32834	+28.56633	-081.26081	ORLANDO	ALAFAYA|	FL	Orange
32835	+28.52038	-081.48354	ORLANDO		FL	Orange
32836	+28.41153	-081.52504	ORLANDO		FL	Orange
32837	+28.39023	-081.43315	ORLANDO		FL	Orange
32839	+28.48993	-081.40614	ORLANDO	EDGEWOOD|	FL	Orange
32853	+28.56633	-081.26081	ORLANDO		FL	Orange
32854	+28.56633	-081.26081	ORLANDO		FL	Orange
33586	+27.87196	-082.43884	SUN CITY		FL	Hillsborough
33587	+27.87196	-082.43884	SYDNEY		FL	Hillsborough
33592	+28.09277	-082.27897	THONOTOSASSA	LAKE THONATO...	FL	Hillsborough
33593	+28.32479	-082.48176	TRILBY		FL	Pasco
33882	+28.02940	-081.73213	WINTER HAVEN		FL	Polk
33883	+28.00255	-081.61864	WINTER HAVEN	WINTER HVN|	FL	Polk
33884	+28.05332	-081.66857	WINTER HAVEN	CYPRESS GAR...	FL	Polk
33885	+28.00255	-081.61864	WINTER HAVEN		FL	Polk
32750	+28.70592	-081.34063	LONGWOOD		FL	Seminole
32751	+28.62547	-081.36459	MAITLAND	EATONVILLE|EA...	FL	Orange
34668	+28.29797	-082.70410	PORT RICHEY		FL	Pasco
34669	+28.36008	-082.62074	HUDSON	PORT RICHEY|	FL	Pasco
34673	+28.32479	-082.48176	PORT RICHEY		FL	Pasco
34674	+28.32479	-082.48176	HUDSON	PORT RICHEY|	FL	Pasco
34677	+28.05032	-082.67737	OLDSMAR		FL	Pinellas
32435	+30.60377	-086.12111	DEFUNIAK SPRI...		FL	Walton
32437	+30.45127	-085.87239	EBRO		FL	Washington
32438	+30.45245	-085.49195	FOUNTAIN		FL	Bay
32439	+30.56126	-086.17060	FREEPORT		FL	Walton
32440	+30.88299	-085.36111	GRACEVILLE		FL	Jackson
32442	+30.71903	-085.08362	GRAND RIDGE		FL	Jackson
32443	+30.85920	-085.14758	GREENWOOD		FL	Jackson

At this point it is understandable that rather than ‘wow’ your reaction is more ‘big deal’… after all Winbatch can already put data into a variety of grids – but (1) those grids aren’t free and many have a learning curve (2) you can get the same results with just Powershell so you are acclimated to a new language with .NET benefits.

Since the Winbatch script simply turns control over to Powershell it is more or less a template to launch any number of data or administrative tasks. It performs the following: (1) initialize the ActiveX control (2) load the PS script and replace several markers (3) execute the PS script. I will address these individually, but the complete WB and PS scripts are included at the end of this document.

```
;constants for output
OUTPUT_CONSOLE = 0
OUTPUT_WINDOW = 1
OUTPUT_BUFFER = 2
oPosh=0

If ! initPosh() Then Exit ;basic init routine
```

The constants are specific to the ActiveX, the default is OUTPUT_BUFFER and will not affect this script. InitPosh():

```
#DefineSubRoutine initPosh()
IntControl(73,1,0,0,0)
If ! IsInReg("SAPIEN.ActiveXPoSH")
   Display(1,"Cannot Continue","SAPIEN.ActiveXPoSH Not Registered")
   Return(0)
EndIf

oPosh = CreateObject("SAPIEN.ActiveXPoSH")
success = oPosh.Init(0)
If success <> 0
   Display(1,"Cannot Continue","Init failed")
   oPosh=0
   Return(0)
EndIf

If !oPosh.IsPowerShellInstalled Then
   Display(1,"Cannot Continue","PowerShell Not Installed")
   oPosh=0
   Return(0)
EndIf

Return(1)

:WBERRORHANDLER
ErrorProcessing(1,1,0,0)
Exit

#EndSubRoutine
```

This udf checks if the ActiveX is registered, if it can be initialized and if Powershell is installed. It does call

```
#DefineFunction isInReg(cProg)
Return( RegExistKey(@RegClasses,cProg) )
#EndFunction
```

Next, loading the PS script and replacing markers

```
cFile=Dirscript():"qryTest.ps1"
If ! FileExist(cFile)
   oPosh=0
   Terminate(@TRUE,"Cannot Continue",cFile:" is missing")
Else
   cScript = FileGet(cFile)
Endif
```

```
title="Florida Zipcodes"
srv="localhost\SQLExpress"
db="Main"
qry="select * from USZip where state='FL'"
t=30

cScript=StrReplace(cScript,"|title|",title)
cScript=StrReplace(cScript,"|srv|",srv)
cScript=StrReplace(cScript,"|db|",db)
cScript=StrReplace(cScript,"|qry|",qry)
cScript=StrReplace(cScript,"|t|",t)
```

[and here is a peek at the section of the PS script that gets replaced]

```
$form.Text = "|title|"
$ServerInstance = "|srv|"
$Database = "|db|"
$Query = "|qry|"
$QueryTimeout=|t|
```

Finally, execute the revised PS script.

```
oPosh.ClearOutput()
oPosh.OutputMode = OUTPUT_BUFFER
oPosh.ClearOutput()
oPosh.Execute(cScript)
oPosh=0
Exit
```

Summing up….

With slight modifications to the basic Winbatch script you can run almost any Powershell snippet you can find… and a good place to start is http://poshcode.org. You will also find tutorials, or a quick cheat sheet here: http://www.dummies.com/how-to/content/windows-powershell-2-for-dummies-cheat-sheet.navId-397925.html.

There will arise issues with error-handling on the PS side; how to communicate back to WB once a PS script is finished; and possibly is it more trouble than it is worth. I will try to address these in a future document.

Stan Littlefield, October 12, 2011

```
;============== Winbatch Script  WB_PS_Datagrid.wbt  ===================


;
```

```
Winbatch 2011A - Automate Poweshell with WB via COM
;
;uses the freeware ActivexPosh Control from Sapien
;(you have to register with them then you can download)
;http://www.sapien.com/software/primalscript
;
;The control allows you to manage Powershell .NET classes through
;either cmdlets or running full scripts
;
;This script uses an example .ps1 file which displays an SQL in a .NET
;datagrid.
;
;Script run on XP SP3 with Powershell 2.0, Win7 with Powershell 3.0
;
;Stan Littlefield, October 12 2011
;////////////////////////////////////////////////////////////////////////////////////////////////
gosub udfs
IntControl(73,1,0,0,0)

;constants for output
OUTPUT_CONSOLE = 0
OUTPUT_WINDOW = 1
OUTPUT_BUFFER = 2
oPosh=0

If ! initPosh() Then Exit ;basic init routine

cFile=Dirscript():"qryTest.ps1"
If ! FileExist(cFile)
   oPosh=0
   Terminate(@TRUE,"Cannot Continue",cFile:" is missing")
Else
   cScript = FileGet(cFile)
Endif

title="Florida Zipcodes"
srv="localhost\SQLExpress"
db="Main"
qry="select * from USZip where state='FL'"
t=30

cScript=StrReplace(cScript,"|title|",title)
cScript=StrReplace(cScript,"|srv|",srv)
cScript=StrReplace(cScript,"|db|",db)
cScript=StrReplace(cScript,"|qry|",qry)
cScript=StrReplace(cScript,"|t|",t)
```

```
oPosh.ClearOutput()
oPosh.OutputMode = OUTPUT_BUFFER
oPosh.ClearOutput()
oPosh.Execute(cScript)
oPosh=0
Exit

:WBERRORHANDLER
ErrorProcessing(1,1,0,0)
Exit

;////////////////////////////////////////////////////////////////////////////////////////////////////
:udfs
#DefineSubRoutine initPosh()
IntControl(73,1,0,0,0)
If ! IsInReg("SAPIEN.ActiveXPoSH")
   Display(1,"Cannot Continue","SAPIEN.ActiveXPoSH Not Registered")
   Return(0)
EndIf

oPosh = CreateObject("SAPIEN.ActiveXPoSH")
success = oPosh.Init(0)
If success <> 0
   Display(1,"Cannot Continue","Init failed")
   oPosh=0
   Return(0)
EndIf

If !oPosh.IsPowerShellInstalled Then
   Display(1,"Cannot Continue","PowerShell Not Installed")
   oPosh=0
   Return(0)
EndIf

Return(1)

:WBERRORHANDLER
ErrorProcessing(1,1,0,0)
Exit

#EndSubRoutine

#DefineFunction isInReg(cProg)
Return( RegExistKey(@RegClasses,cProg) )
#EndFunction
```

```
#DefineSubroutine ErrorProcessing(deleteIni,showerr,logfile,Err_Array)
If Vartype(Err_Array) ==256
   WbError = Err_Array[0]
   wberrorhandlerline = Err_Array[1]
   wberrorhandleroffset = Err_Array[2]
   wberrorhandlerassignment = Err_Array[3]
   wberrorhandlerfile = Err_Array[4]
   wberrortextstring = Err_Array[5]
   wberroradditionalinfo = Err_Array[6]
   wberrorinsegment = Err_Array[7]
Else
   WbError = LastError()
Endif
WbTextcode = WbError
If WbError==1668||WbError==2669||WbError==3670
   WbError = ItemExtract(1,IntControl(34,-1,0,0,0),":")
   WbTextcode = -1
EndIf
WbErrorString = IntControl(34,WbTextcode,0,0,0)
WbErrorDateTime = TimeYmdHms()
If deleteIni
   WbErrorFile = StrCat(ShortCutDir( 'AppData', 0, 0 ),'\WinBatch\Settings\')
   If ! DirExist(WbErrorFile) Then WbErrorFile = DirWindows(0)
   WbErrorFile = StrCat(WbErrorFile,"WWWBATCH.INI")
   FileDelete(WbErrorFile)
   IniWritePvt(WbErrorDateTime,"CurrentScript",WbErrorHandlerFile      ,WbErrorFile)
   IniWritePvt(WbErrorDateTime,"ErrorValue"   ,WbError                 ,WbErrorFile)
   IniWritePvt(WbErrorDateTime,"ErrorString"  ,WbErrorString           ,WbErrorFile)
   IniWritePvt(WbErrorDateTime,"ScriptLine"   ,WbErrorHandlerLine      ,WbErrorFile)
   IniWritePvt(WbErrorDateTime,"ScriptOffset" ,WbErrorHandlerOffset    ,WbErrorFile)

IniWritePvt(WbErrorDateTime,"VarAssignment",WbErrorHandlerAssignment,WbError
File)
   IniWritePvt(WbErrorDateTime,"VarInSegment" ,WbErrorInSegment,WbErrorFile)
   IniWritePvt("","","",WbErrorFile)
Endif
WbErrorMsgText = StrCat(WbErrorDateTime,@CRLF)
WbErrorMsgText = StrCat(WbErrorMsgText,"Current Script:
",WbErrorHandlerFile,@CRLF)
WbErrorMsgText = StrCat(WbErrorMsgText,"Error# [",WbError,"]",@CRLF)
WbErrorMsgText = StrCat(WbErrorMsgText,"Error Text: ",wberrortextstring,@CRLF)
WbErrorMsgText = StrCat(WbErrorMsgText,"[Extended Information]
",wberroradditionalinfo,@CRLF,@CRLF)
WbErrorMsgText = StrCat(WbErrorMsgText,"On
Line:",@CRLF,WbErrorHandlerLine,@CRLF)
```

```
;WbErrorMsgText = StrCat(WbErrorMsgText,"Offset:
",WbErrorHandlerOffset,@CRLF)
If (WbErrorHandlerAssignment>"") Then %WbErrorHandlerAssignment% =
"UNKNOWN"
WbErrorMsgText = StrCat(WbErrorMsgText,"Assignment/Variable:
",WbErrorHandlerAssignment,@CRLF)
If (WbErrorInSegment>"") Then WbErrorMsgText = StrCat(WbErrorMsgText,"In
UDF/UDS: ",WbErrorInSegment,@CRLF)
If logfile
   cSep = StrCat(StrFill("=",50),@CRLF)
   cLogFile = StrCat(dirscript(),"log.err")
   If ! FileExist(cLogFile) Then FilePut(cLogFile,StrCat("Error Log",@CRLF,cSep))
   FilePut(cLogFile,StrCat(FileGet(cLogFile),WbErrorMsgText,cSep))
   display(2,"An Error Occured",StrCat("written to ",cLogFile))
Else
   If showerr
      WbErrorMsgText = StrCat(WbErrorMsgText,"[THIS ERROR NOT WRITTEN TO
LOG FILE]",@CRLF)
      Message("An Error Was Encountered",WbErrorMsgText)
   Endif
Endif
Return(1)
#EndSubroutine

Return
;////////////////////////////////////////////////////////////////////////////////////////////////////


====== qrytest.ps1

<#
This is a blocked comment (version 2.0). This section of the powershell script
loads the required .NET assemblies to create a window
#>
[void][System.Reflection.Assembly]::LoadWithPartialName("System.Drawing")
[void][System.Reflection.Assembly]::LoadWithPartialName("System.windows.forms")
$form = new-object System.Windows.Forms.form
$form.size = new-object System.Drawing.Size(800,620)

<#
This section allows you to replace markers from
your Winbatch code permitting a degree of reuseable code
where the powerscript file acts as a template
#>
$form.Text = "|title|"
$ServerInstance = "|srv|"
```

```
$Database = "|db|"
$Query = "|qry|"
$QueryTimeout=|t|

<#
This section creates .NET objects to process the SQL Query
and return the results into a dataset. Then the connection
is closed.
#>
$conn=new-object System.Data.SqlClient.SQLConnection
$conn.ConnectionString="Server={0};Database={1};Integrated Security=True" -f
$ServerInstance,$Database
$conn.Open()
$cmd=new-object system.Data.SqlClient.SqlCommand($Query,$conn)
$cmd.CommandTimeout=$QueryTimeout
$ds=New-Object system.Data.DataSet
$da=New-Object system.Data.SqlClient.SqlDataAdapter($cmd)
[void]$da.fill($ds)
$conn.Close()

<#
Put the dataset into a grid and add the grid
to the form.
#>
$grid = New-Object System.Windows.Forms.DataGridView
$grid.Location = new-object System.Drawing.Size(10,10)
$grid.size = new-object System.Drawing.Size(980,600)
$grid.AutoSizeRowsMode = "AllHeaders"
$form.Controls.Add($grid)
$grid.DataSource = $ds.Tables[0].DefaultView

<#
Finally, display the window. Note this is merely a
display of the data. You could add buttons and other
controls to the form, or even make it editable.
#>
$form.autoscroll = $true
$form.Add_Shown({$form.Activate()})
$form.ShowDialog()
#once window is closed you are returned to Winbatch
```