A DBA's first PowerShell script: lessons learned.

David Cobb
SQL Consultant and Trainer

MCITP SQL 2008 DBA,Dev,BI
sql@davidcobb.net

About Dave

- Computer consultant since 1996
  - Background in technical support, web application design, network administration
  - Primarily Windows, SQL Server
  - Dabble in Linux, Opensource
- Microsoft Certified SQL Trainer since 2002
- MCITP, MCAD, MCSE 2000(I like taking tests.)
- Lead I.T. Engineer with CheckAlt Payment Solutions providing Check21 Remote Deposit Capture solutions.
- Enjoy helping my clients solve their I.T. problems
- PowerShell Student and Fan ☺
- http://daveslog.com
- sql@davidcobb.net I like emails.(..when they're from people)

PowerShell NOOB

- Familiar with Windows and SQL
- Some Development Background, more on the Admin side
- Interest in PowerShell from reading SQL bloggers
- Just needed a problem to solve to learn on the job
- Let's walk through that problem and my first solution
- Comments and discussion welcome

The Problem

- "Write us a script to restore the latest backups from the network drive onto the development servers"
- Alright should be easy.
- Well...

Environment

- SQL 2012 on Win 2008 R2
- I have Powershell 2 and SQLPS
- I can install Powershell 3 for development, use the Integrated Scripting Environment /ISE

My Initial Approach

- Break the problem into pieces, solve each piece:
  - Set starting variables
  - Connect to network share
  - Repeat for each requested server:
    - Repeat for each requested database:
      - Find matching backups
      - Find latest backup
      - Restore that backup
  - Done! Super easy piece of cake. Thank you for coming!

Where do I start!!??

- I jumped in by reading blogs with PowerShell tutorials.
- Copied and pasted their code, and ran it, tweaked and ran again.
- Used the ISE to set breakpoints, look at objects in the PowerShell pipeline.

- Be on the lookout for my new book:

How to Copy and Paste your way to an I.T. Career!*

Credit: Joe Lopez, former boss and great guy

But there were the Gotchas…

- Network drives
- Restore and filenames
- PowerShell and SQL version differences on different servers
- Execution Policy
- SQL Cmdlet bug!

Gotcha 1:
Network drives don't work consistently

- Initially using script over network drive works. Man I am a GURU!
- Then I get errors…
- After a few hours of frustration, I discover PSDrive ☺

New-PSDrive -name "Y" -PSProvider FileSystem -Root $BackupRoot

Gotcha 2:
Data and log files aren't in the same location in production and development

- Restoring a database when the paths for data and log are different means you need WITH MOVE command clause
- How do I move data files with RESTORE, I don’t know the logical name!
- My approach:
  See Code
- Next time: *(From Get-Help Restore-SQLDatabase –examples)*

```
$RelocateData = New-Object Microsoft.SqlServer.Management.Smo.RelocateFile("MyDB_Data", "c:\MySQLServer\MyDB.mdf")    $RelocateLog = New-Object Microsoft.SqlServer.Management.Smo.RelocateFile("MyDB_Log", "c:\MySQLServer\MyDB.ldf")
Restore-SqlDatabase -ServerInstance Computer\Instance -Database MyDB -BackupFile "\\share\folder\MyDB.trn" -RelocateFile @($RelocateData,$RelocateLog)
```

Client: "Oh can we have that in a SQL Job“

Trickier that it looked at first

- SQL Jobs can run PowerShell scripts, but.. I have to paste in the script to the job step, hard to update on multiple machines
- I can run a script stored in a central file share, and execute it with a CmdExec SQL Agent Job Task...but ExecutionPolicy prevents this.
- My ‘Solution’:
  - If I use ByPass to circumvent execution policy, like so:
    *powershell -version 2 -**ExecutionPolicy ByPass** -Command "& \\server\share$\RestoreDBsFromSource.ps1"*
  - *Runs fine from SQL Service Account with read permission on the network share.*
  - *Anything bother you about that?*

This raises a question!

- Hey, so I can run PowerShell scripts with BYPASS without Admin rights?
- Yep: http://blogs.technet.com/b/heyscriptingguy/archive/2011/01/04/run-powershell-scripts-stored-on-a-central-file-share.aspx

"...Interestingly, admin rights are not required to launch PowerShell in bypass mode. ... If you want to restrict it from users, then you should use software restriction policies. But keep in mind, that would preclude you from using Windows PowerShell for your logon scripts and for many other administrative purposes...."

- This bears further investigation ☺

- But the script works, client happy, but soon..

Client:

“Can you make it work in 2008 R2 for our 'new' Dev servers?”

Gotcha 3:

- Problem:
  SQL 2008 R2 Can't use **Import-Module SQLPS**
- Solution:
  - Michiel Wories and his **Initialize-SqlPsEnvironment.ps1**

Let’s just take a moment to say…

PowerShell Community is Awesome ☺

- The amount and quality of help resources out there is amazing.
- Find them, read them, apply what you learn and thank them. ☺

- Thanks PowerShell People!

Gotcha 4:

- I can connect to SQL now, but…
- I run and backups timeout after 30 seconds. Worked in SQL 2012!
- Now **Invoke-SqlCmd** times out, why?
- **Invoke-SqlCmd** bug in sql 2008 R2, doesn't respect timeout parameter
- http://www.scarydba.com/2010/04/16/powershell-smo-problem/
- Chad Miller to the rescue with **Invoke-SqlCmd2** (Thanks again!)

Client:

"Can you make it work with a parameter with a list of Databases we want?"

- Actually easier than I thought..

Change to the code:

```
param
([parameter(Mandatory = $true)]
[string[]]$DBList)
```

And the call:

```
powershell -version 2 -ExecutionPolicy ByPass -Command "&
\\davepc\scripts$\RestoreDBsFromSourceV2.ps1" Northwind,Products,Foo
```

- For V3, I plan to create parameters with defaults for my other initial variables.

My Process

- Break your scripting problem down to small tasks
- Solve each one in turn
- Someone out there has probably solved your problem first
- Good enough for now is OK, make a note to refine it later
- Find, copy, paste, UNDERSTAND, modify, test, refine, repeat!

Resources

- Whatever it is you're doing with PowerShell, someone has probably done it before and blogged about it!
- Take advantage of the excellent free resources out there for learning PowerShell.
- Read other people's code, and adapt for your needs.
- Use the tools!

PowerShell Help

- Stackoverflow.com
  http://stackoverflow.com/questions/tagged/PowerShell
  (Hard questions, great answers, great explanations)
- PowerShell Community Resources
  - Technet PowerShell Communities
  - PowerShellCommunity.org
  - Florida PowerShell User Group - Resources

- Just search the web ☺ many great bloggers, may great resources

References

- Dr Tobias Weltner's Mastering PowerShell
  http://PowerShell.com/Mastering-PowerShell.pdf
- PowerShell V2 Owners Manual
  http://bit.ly/P0s0g4
- Windows PowerShell 3.0 and Server Manager Quick Reference Guides
  http://bit.ly/LaojTT
- Stairway to SQL PowerShell
  http://bit.ly/MkM62G
- Running PowerShell 2 and 3 side by side
  http://bit.ly/tPkfAq