



SQLintersection

Session: Sunday, June 8, 2021 9:00 AM

Tips and Tricks for the PowerShell DBA (2)

Ben J Miller

ben@benmiller.net

SQL Server & Azure SQL
Conference

POWERED BY
Microsoft & Next Gen



Get *Whova* for all 4 conferences

Official Event App

- **Access links** to join all sessions and keynotes virtually
- Explore the **professional profiles** of event speakers and attendees
- Send **in-app messages** and **exchange contact info**
- **Network and find attendees** with common affiliations, educations, shared networks, and social profiles
- Receive **update notifications** from organizers
- Access the **event agenda**, GPS guidance, maps, and parking directions at your fingertips

Download Whova and take your event mobile.

Get Whova from the App Store or Google Play.



Please sign up for the app with your social media account or email.

The event invitation code is:
damah

You will be asked for an event invitation code after installing Whova.



Sponsors & Exhibitors

Platinum Sponsors



Booth 7



ORANGE BEES

Booth 6

Quest

Gold Sponsors



Silver Sponsors



Booth 5

Exhibitors



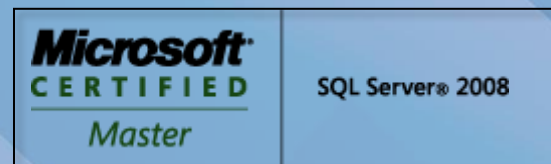
Booth 2



Booth 1

Speaker: Ben Miller - DBADuck

- Data Platform MVP since 2009
- Microsoft Certified Master in SQL Server
- Author – <https://dbaduck.com>
 - MVP Deep Dives #2
 - PowerShell MVP Deep Dives
 - PowerShell Stairway on SQLServerCentral.com
- Email : ben@benmiller.net
- Twitter: @DBADuck
- GitHub: github.com/dbaduck



Introduction

- **Loading SMO Assemblies and Getting Assemblies loaded**
- **SetDefaultInitFields**
- **Refresh**
- **Test-Path in the SQL Provider**
- **SQL Authentication and the Provider**
- **Invoke-SqlCmd**
- **Write-SqlTableData**
- **Using SQL Provider Context**
- **Showing System Objects in the Provider**

Loading Assemblies for SMO

- **No longer installed in the GAC on SSMS install**
- **They are available if SSMS is installed**
- **Load them from modules**
 - SQLServer
 - DBAtools
- **Load the latest or later than you are connecting to**
 - Load 15 if you are going to connect to 15,14,13,12,10,9
 - SSMS and Dbatools and SqlServer all have the latest ones available
- **Add-Type -Path dllpath.dll**

Where can you find them?

- **Import-Module dbatools**
- **Import-Module sqlserver**
- **C:\Program Files (x86)\Microsoft SQL Server Management Studio 18\Common7\IDE**

Getting Assemblies Loaded

- **Assemblies get loaded by PowerShell**
 - **Assemblies can be loaded by Modules**
 - **Assemblies can be loaded by You**
-
- **PowerShell is on top of .NET (Core or Full)**
 - **Assemblies have functionality you don't have to write**
-
- **[AppDomain]::CurrentDomain.GetAssemblies()**

SMO (Shared Management Objects)

- **SMO is a great tool and totally accessible from PowerShell**
- **Knowing when to use it is the key**
- **If you need to manipulate the object (Table, Database, etc.) use SMO**
- **If you need information, you can use SMO**
- **If you need information from a set of objects, testing is needed**
- **Some objects can be easily pulled and manipulated using**
 - `Server.SetDefaultInitFields`

SetDefaultInitFields

- **Method is on the Server object**
- **Parameters are Object Type and String Collection of fields**
- **Types are like this**
 - [Microsoft.SqlServer.Management.Smo.Database]
- **String collections are like this**
- **\$coll = new-object -TypeName System.Collections.StringCollection**
- **\$coll.Add("ID")**

Refresh() on SMO Objects

- **Objects are powerful in PowerShell**
- **Objects pull values from SQL and populate properties in memory**
- **Once in Memory they are not asked for again**
- **Use Refresh() on any object to signal to SMO to retrieve it again**
- **Normally needed when you are adding objects or if something is changing on the SQL side like Database Size or space used on the object**

Test-Path for existence

- Returns \$true or \$false
- SQLPS (deprecated) and SQL Server module load a Provider
- Providers represent a service in a Path-like Structure
- Providers manifest themselves as Drives (PSDrives)
- **SQLSERVER:** is the drive that gets loaded by the SqlServer module
- **SQLSERVER:\sql\servername\instancename\Databases\master**
 - If you are using a Default Instance the instance name is DEFAULT

Using SQL Authentication with SQL Provider

- **By Default, SQLSERVER: drive uses Windows Authentication**
- **Get-PSDrive to see which drives are present**
- **New-PSDrive to create a new drive**
- **SQL Provider recognizes the -Credential parameter for SQL Auth**
- **When you see SQLSERVER:\sql\path.....+sa**
 - The + indicates that you are using SQL authentication
- **Need the Root at least to the Instance level to use SQL Authentication but you can go further, even with Windows Authentication**

Invoke-SqlCmd

- **Using TSQL in PowerShell can be useful**
- **\$query = “select * from sys.databases”**
- **Use Invoke-SqlCmd to get the data**
 - ServerInstance servername
 - Database databasename
 - Query \$query
- **Special clauses in the SqlServer module now**
 - -OutputAs (DataRows (default), DataTable, DataSet)
- **Use QueryTimeout 0 if you intend it to take longer than 10 minutes**

Write-SqlTableData

- **This one is in the SqlServer module (NOT SQLPS)**
- **One like this is in the DBAtools module as well (Write-DbaDbTableData)**
- **Basically you can write data to a table in SQL from an object**
- **-Force can be used to create the table**
 - Strings become nvarchar(max) so not amazingly useful
- **Super fast, uses SqlBulkCopy to get the data in**
- **Automaps by position of the column, not by column name**

Using the SqlProvider Context

- **Commands in the SqlServer module understand context**
- **Inside SQLSERVER:\sql\server\instance\Databases you can use a command that understands where you are.**
- **Invoke-SqlCmd will use the context of the server you are in**
- **Backup-SqlDatabase will also use context and will also use settings in the instance**

Showing System Objects in SqlProvider

- `$SqlServerIncludeSystemObjects` is `$false` by default
- This controls whether the Provider shows system objects
- Simply Set it to `$true`
- `$SqlServerIncludeSystemObjects = $true`
- Next call will show system objects

Demo

Tips and Tricks for the PowerShell DBA

Part 2

Review

- **SMO**
- **SetDefaultInitFields**
- **Refresh**
- **Test-Path in the SQL Provider**
- **SQL Authentication and the Provider**
- **Invoke-SqlCmd**
- **Write-SqlTableData**
- **Using SQL Provider Context**
- **Showing System Objects in the Provider**

Questions?

Don't forget to complete an online evaluation!

Tips and Tricks for the PowerShell DBA (2)

Your evaluation helps organizers build better conferences
and helps speakers improve their sessions.

SQL Server & Azure SQL
Conference

POWERED BY
Microsoft & Next Gen

Thank you!

IN-PERSON EVENT

DECEMBER 7–9
2021

SQL Server & Azure SQL
Conference

POWERED BY
Microsoft & Next Gen

LAS VEGAS, NV
MGM GRAND

