

holger SCHWICHTENBERG

WINDOWS

# PowerShell 4.0

DAS PRAXISBUCH



Im Internet: Codebeispiele, Forum,  
PowerShell-Kurzreferenz

HANSER

[www.IT-Visions.de](http://www.IT-Visions.de)

## Bleiben Sie auf dem Laufenden!



Unser **Computerbuch-Newsletter** informiert Sie monatlich über neue Bücher und Termine. Profitieren Sie auch von Gewinnspielen und exklusiven Leseproben. Gleich anmelden unter



[www.hanser-fachbuch.de/newsletter](http://www.hanser-fachbuch.de/newsletter)



**Hanser Update** ist der IT-Blog des Hanser Verlags mit Beiträgen und Praxistipps von unseren Autoren rund um die Themen Online Marketing, Webentwicklung, Programmierung, Softwareentwicklung sowie IT- und Projektmanagement. Lesen Sie mit und abonnieren Sie unsere News unter



[www.hanser-fachbuch.de/update](http://www.hanser-fachbuch.de/update)   



Holger Schwichtenberg

# Windows PowerShell 4.0

Das Praxisbuch

HANSER

Alle in diesem Buch enthaltenen Informationen, Verfahren und Darstellungen wurden nach bestem Wissen zusammengestellt und mit Sorgfalt getestet. Dennoch sind Fehler nicht ganz auszuschließen. Aus diesem Grund sind die im vorliegenden Buch enthaltenen Informationen mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Autor und Verlag übernehmen infolgedessen keine juristische Verantwortung und werden keine daraus folgende oder sonstige Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieser Informationen – oder Teilen davon – entsteht.

Ebenso übernehmen Autor und Verlag keine Gewähr dafür, dass beschriebene Verfahren usw. frei von Schutzrechten Dritter sind. Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Buch berechtigt deshalb auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.



Bibliografische Information der Deutschen Nationalbibliothek:

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Dieses Werk ist urheberrechtlich geschützt.

Alle Rechte, auch die der Übersetzung, des Nachdruckes und der Vervielfältigung des Buches, oder Teilen daraus, vorbehalten. Kein Teil des Werkes darf ohne schriftliche Genehmigung des Verlages in irgendeiner Form (Fotokopie, Mikrofilm oder ein anderes Verfahren) – auch nicht für Zwecke der Unterrichtsgestaltung – reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

© 2014 Carl Hanser Verlag München, [www.hanser-fachbuch.de](http://www.hanser-fachbuch.de)

Lektorat: Brigitte Bauer-Schiewek

Copy editing: Petra Kienle, Fürstenfeldbruck

Herstellung: Irene Weillhart

Umschlagdesign: Marc Müller-Bremer, [www.rebranding.de](http://www.rebranding.de), München

Umschlagrealisation: Stephan Rönigk

Gesamtherstellung: Kösel, Krugzell

Ausstattung patentrechtlich geschützt. Kösel FD 351, Patent-Nr. 0748702

Printed in Germany

Print-ISBN: 978-3-446-44070-8

E-Book-ISBN: 978-3-446-44120-0

**»Der Weltuntergang steht bevor,  
aber nicht so, wie Sie denken.  
Dieser Krieg jagt nicht alles in die Luft,  
sondern schaltet alles ab.«**



**Tom DeMarco  
Als auf der Welt das Licht ausging**

ca. 560 Seiten. Hardcover  
ca. € 19,99 [D] / € 20,60 [A] / sFr 28,90  
ISBN 978-3-446-43960-3  
Erscheint im November 2014

**Hier klicken zur  
Leseprobe**

Sie möchten mehr über Tom DeMarco und seine Bücher erfahren.  
Einfach reinklicken unter [www.hanser-fachbuch.de/special/demarco](http://www.hanser-fachbuch.de/special/demarco)

# Inhalt

<b>Vorwort zur vierten Auflage 2014</b> .....	<b>XIX</b>
Wer bin ich? .....	XIX
Wer sind Sie? .....	XIX
Was ist neu in diesem Buch? .....	XX
Wem ist zu danken? .....	XX
Woher bekommen Sie die Beispiele aus diesem Buch? .....	XXI
Wo kann ich mich schulen oder beraten lassen? .....	XXI
Zum Schluss des Vorworts ... ..	XXI
<b>Über den Autor Dr. Holger Schwichtenberg</b> .....	<b>XXIII</b>
<b>Über den Co-Autor Peter Monadjemi</b> .....	<b>XXIV</b>
<b>Teil A: PowerShell-Basiswissen</b> .....	<b>1</b>
<b>1 Erste Schritte mit der Windows PowerShell</b> .....	<b>3</b>
1.1 Was ist die Windows PowerShell? .....	3
1.2 Geschichte der Windows PowerShell .....	4
1.3 Motivation zur Windows PowerShell .....	4
1.4 Betriebssysteme mit vorinstallierter PowerShell .....	7
1.5 PowerShell herunterladen und installieren auf anderen Betriebssystemen .....	9
1.6 Die Windows PowerShell testen .....	12
1.7 Woher kommen die Commandlets? .....	17
1.8 PowerShell Community Extensions (PSCX) herunterladen und installieren .....	18
1.9 Den PowerShell-Editor „ISE“ verwenden .....	19
<b>2 Architektur der Windows PowerShell</b> .....	<b>23</b>
<b>3 Einzelbefehle der PowerShell</b> .....	<b>27</b>
3.1 Commandlets .....	27

3.2	Aliase	35
3.3	Ausdrücke	42
3.4	Externe Befehle	43
3.5	Dateinamen	44
<b>4</b>	<b>Hilfefunktionen</b>	<b>47</b>
4.1	Auflisten der verfügbaren Befehle	47
4.2	Erläuterungen zu den Befehlen	50
4.3	Hilfe zu Parametern	51
4.4	Hilfe mit Show-Command	51
4.5	Hilfefenster	52
4.6	Aktualisieren der Hilfedateien	53
4.7	Online-Hilfe	55
4.8	Dokumentation der .NET-Klassen	55
<b>5</b>	<b>Objektorientiertes Pipelining</b>	<b>59</b>
5.1	Pipeline-Operator	59
5.2	.NET-Objekte in der Pipeline	60
5.3	Pipeline Processor	62
5.4	Pipelining von Parametern	63
5.5	Pipelining von klassischen Befehlen	65
5.6	Anzahl der Objekte in der Pipeline	66
5.7	Zugriff auf einzelne Objekte aus einer Menge	66
5.8	Zugriff auf einzelne Werte in einem Objekt	68
5.9	Methoden ausführen	69
5.10	Analyse des Pipeline-Inhalts	71
5.11	Filtern	82
5.12	Zusammenfassung von Pipeline-Inhalten	85
5.13	„Kastrierung“ von Objekten in der Pipeline	85
5.14	Sortieren	86
5.15	Duplikate entfernen	87
5.16	Gruppierung	88
5.17	Berechnungen	89
5.18	Zwischenschritte in der Pipeline mit Variablen	90
5.19	Verzweigungen in der Pipeline	91
5.20	Vergleiche zwischen Objekten	93
5.21	Praxisbeispiele	93
<b>6</b>	<b>PowerShell-Skripte</b>	<b>95</b>
6.1	Skriptdateien	95
6.2	Start eines Skripts	97

6.3	Aliase für Skripte verwenden	98
6.4	Parameter für Skripte	99
6.5	Skripte dauerhaft einbinden (Dot Sourcing)	100
6.6	Sicherheitsfunktionen für PowerShell-Skripte	101
6.7	Skripte anhalten	104
<b>7</b>	<b>PowerShell-Skriptsprache</b>	<b>105</b>
7.1	Hilfe zur PowerShell-Skriptsprache	105
7.2	Befehlstrennung	106
7.3	Kommentare	106
7.4	Variablen	107
7.5	Variablenbedingungen	113
7.6	Zahlen	114
7.7	Zeichenketten	116
7.8	Reguläre Ausdrücke	121
7.9	Datum und Uhrzeit	128
7.10	Arrays und assoziative Arrays (Hash-Tabelle)	129
7.11	Operatoren	132
7.12	Überblick über die Kontrollkonstrukte	134
7.13	Schleifen	134
7.14	Bedingungen	139
7.15	Unterroutinen (Prozedur/Funktionen)	141
7.16	Eingebaute Funktionen	147
7.17	Fehlerbehandlung	147
<b>8</b>	<b>Ausgaben</b>	<b>155</b>
8.1	Ausgabe-Commandlets	155
8.2	Out-GridView	157
8.3	Standardausgabe	159
8.4	Einschränkung der Ausgabe	161
8.5	Seitenweise Ausgabe	161
8.6	Ausgabe einzelner Werte	162
8.7	Details zum Ausgabeoperator	164
8.8	Benutzerdefinierte Tabellenformatierung	167
8.9	Ausgabe von Methodenergebnissen und Unterobjekten in Pipelines	168
8.10	Ausgabe von Methodenergebnissen und Unterobjekten in Zeichenketten	168
8.11	Unterdrückung der Ausgabe	169
8.12	Ausgaben an Drucker	170
8.13	Ausgaben in Dateien	170
8.14	Umleitungen (Redirection)	171

<b>9</b>	<b>Benutzereingaben</b>	<b>173</b>
9.1	Read-Host	173
9.2	Grafischer Eingabedialog	174
9.3	Dialogfenster	174
9.4	Authentifizierungsdialog	175
<b>10</b>	<b>Das PowerShell-Navigationsmodell</b>	<b>177</b>
10.1	Einführungsbeispiel: Navigation in der Registrierungsdatenbank	177
10.2	Provider und Laufwerke	178
10.3	Navigationsbefehle	180
10.4	Pfadangaben	181
10.5	Beispiel	183
10.6	Eigene Laufwerke definieren	184
<b>11</b>	<b>PowerShell-Werkzeuge</b>	<b>185</b>
11.1	PowerShell-Standardkonsole	185
11.2	PowerShell Integrated Scripting Environment (ISE)	190
11.3	CIM Explorer for PowerShell ISE	196
11.4	PowerShell Web Access (PSWA)	197
11.5	PowerShellPlus	203
11.6	PoshConsole	206
11.7	PowerGUI	207
11.8	PowerShell Analyzer	208
11.9	PrimalScript	209
11.10	PowerShell Help	211
11.11	PowerShell Help Reader	211
11.12	PowerTab	212
11.13	NuGet Package Manager	212
11.14	PowerShell Tools for Visual Studio	213
11.15	PowerShell Remoting	214
11.16	Vergleich der Skripteditoren	214
<b>Teil B: PowerShell-Aufbauwissen</b>		<b>217</b>
<b>12</b>	<b>Fernausführung (Remoting)</b>	<b>219</b>
12.1	Fernabfrage ohne WS-Management	220
12.2	Anforderungen	221
12.3	Rechte für Fernaufrufe	222
12.4	Einrichten von WinRM	223
12.5	Überblick über die Fernausführungs-Commandlets	225
12.6	Interaktive Fernverbindungen im Telnet-Stil	225
12.7	Fernausführung von Befehlen	226

12.8	Fernausführung von Skripten .....	230
12.9	Ausführung auf mehreren Computern .....	231
12.10	Sitzungen .....	232
12.11	Zugriff auf entfernte Computer außerhalb der eigenen Domäne .....	236
12.12	Verwaltung des WS-Management-Dienstes .....	240
<b>13</b>	<b>Verwendung von .NET-Klassen .....</b>	<b>243</b>
13.1	Microsoft Developer Network (MSDN) .....	243
13.2	Erzeugen von Instanzen .....	244
13.3	Parameterbehaftete Konstruktoren .....	245
13.4	Initialisierung von Objekten .....	246
13.5	Nutzung von Attributen und Methoden .....	247
13.6	Statische Mitglieder in .NET-Klassen und statische .NET-Klassen .....	249
13.7	Zugriff auf bestehende Objekte .....	251
13.8	Laden von Assemblies .....	252
13.9	Objektanalyse .....	253
13.10	Auflistungen (Enumerationen) .....	254
13.11	Verknüpfen von Aufzählungswerten .....	255
<b>14</b>	<b>Verwendung von COM-Klassen .....</b>	<b>257</b>
14.1	Erzeugen von Instanzen .....	257
14.2	Nutzung von Attributen und Methoden .....	258
14.3	Holen bestehender Instanzen .....	259
<b>15</b>	<b>Zugriff auf die Windows Management Instrumentation (WMI) ..</b>	<b>261</b>
15.1	Einführung in WMI .....	261
15.2	WMI in der Windows PowerShell .....	287
15.3	Abruf von WMI-Objektmengen .....	288
15.4	Fernzugriffe .....	289
15.5	Filtern und Abfragen .....	289
15.6	Liste aller WMI-Klassen .....	293
15.7	Hintergrundwissen: WMI-Klassenprojektion mit dem PowerShell-WMI-Objektadapter .....	294
15.8	Beschränkung der Ausgabeliste bei WMI-Objekten .....	297
15.9	Zugriff auf einzelne Mitglieder von WMI-Klassen .....	299
15.10	Werte setzen in WMI-Objekten .....	299
15.11	Umgang mit WMI-Datumsangaben .....	301
15.12	Methodenaufrufe .....	302
15.13	Neue WMI-Instanzen erzeugen .....	303
15.14	Instanzen entfernen .....	304
15.15	Commandlet Definition XML-Datei (CDXML) .....	305

<b>16</b>	<b>Dynamische Objekte</b> .....	<b>307</b>
16.1	Erweitern bestehender Objekte .....	307
16.2	Komplett dynamische Objekte .....	309
<b>17</b>	<b>Einbinden von C# und VB.NET</b> .....	<b>311</b>
<b>18</b>	<b>Win32-API-Aufrufe</b> .....	<b>313</b>
<b>19</b>	<b>Fehlersuche</b> .....	<b>317</b>
19.1	Detailinformationen .....	317
19.2	Einzelstschrittmodus .....	318
19.3	Zeitmessung .....	319
19.4	Ablaufverfolgung .....	320
19.5	Kommandozeilenbasiertes Script-Debugging .....	320
<b>20</b>	<b>Transaktionen</b> .....	<b>323</b>
20.1	Commandlets für Transaktionen .....	323
20.2	Start und Ende einer Transaktion .....	324
20.3	Zurücksetzen der Transaktion .....	325
20.4	Mehrere Transaktionen .....	326
<b>21</b>	<b>Hintergrundaufträge („Jobs“)</b> .....	<b>327</b>
21.1	Voraussetzungen .....	327
21.2	Architektur .....	328
21.3	Starten eines Hintergrundauftrags .....	328
21.4	Hintergrundaufträge abfragen .....	329
21.5	Warten auf einen Hintergrundauftrag .....	330
21.6	Abbrechen und Löschen von Aufträgen .....	330
21.7	Analyse von Fehlermeldungen .....	331
21.8	Fernausführung von Hintergrundaufträgen .....	331
<b>22</b>	<b>Geplante Aufgaben und zeitgesteuerte Jobs</b> .....	<b>333</b>
22.1	Geplante Aufgaben (Scheduled Tasks) .....	333
22.2	Zeitgesteuerte Jobs .....	337
<b>23</b>	<b>PowerShell-Workflows</b> .....	<b>343</b>
23.1	Ein erstes Beispiel .....	343
23.2	Unterschiede zu einer Function bzw. einem Skript .....	348
23.3	Einschränkungen bei Workflows .....	348
23.4	Workflows in der Praxis .....	350
23.5	Workflows in Visual Studio erstellen .....	357

<b>24</b>	<b>Ereignissystem</b>	<b>375</b>
24.1	WMI-Ereignisse	375
24.2	WMI-Ereignisabfragen	375
24.3	WMI-Ereignisse ab PowerShell 1.0	377
24.4	Registrieren von WMI-Ereignisquellen seit PowerShell 2.0	378
24.5	Auslesen der Ereignisliste	379
24.6	Reagieren auf Ereignisse	381
24.7	WMI-Ereignisse ab PowerShell-Version 3.0	383
24.8	Registrieren von .NET-Ereignissen	383
24.9	Erzeugen von Ereignissen	384
<b>25</b>	<b>Datenbereiche und Datendateien</b>	<b>387</b>
25.1	Datenbereiche	387
25.2	Datendateien	389
25.3	Mehrsprachigkeit/Lokalisierung	390
<b>26</b>	<b>Desired State Configuration (DSC)</b>	<b>393</b>
26.1	Grundprinzipien	394
26.2	Ressourcen	394
26.3	Eigenschaften eines Ressourcentyps	395
26.4	Aufbau einer DSC-Datei	395
26.5	Ein erstes DSC-Beispiel	396
26.6	Anwendung einer DSC-Datei	396
26.7	Parameter für DSC-Dateien	398
26.8	Verfügbare DSC-Ressourcentypen	399
26.9	Commandlets für die Arbeit mit DSC	401
26.10	DSC-Interna	401
26.11	DSC-Praxisbeispiel 1: IIS installieren	402
26.12	DSC-Praxisbeispiel 2: Software installieren	403
26.13	DSC-Praxisbeispiel 3: Software deinstallieren	405
<b>27</b>	<b>PowerShell-Snap-Ins</b>	<b>407</b>
27.1	Einbinden von Snap-Ins	407
27.2	Liste der Commandlets	411
27.3	Doppeldeutige Namen	412
<b>28</b>	<b>PowerShell-Module</b>	<b>413</b>
28.1	Überblick über die Commandlets	413
28.2	Modulararchitektur	414
28.3	Module installieren	415
28.4	Auflisten der verfügbaren Module	415

28.5	Importieren von Modulen .....	417
28.6	Entfernen von Modulen .....	420
<b>29</b>	<b>Ausgewählte PowerShell-Erweiterungen .....</b>	<b>421</b>
29.1	PowerShell-Module in Windows 7 und Windows Server 2008 R2 .....	421
29.2	PowerShell-Module in Windows 8.x und Windows Server 2012/2012 R2 .....	423
29.3	BITSTransfer .....	427
29.4	Windows Server Backup .....	428
29.5	PowerShell Diagnostics .....	428
29.6	Windows PowerShell Community Extensions .....	429
29.7	PowerShellPack .....	433
29.8	www.IT-Visions.de PowerShell Extensions .....	434
29.9	Quest Management Shell for Active Directory .....	435
29.10	Microsoft Exchange Server .....	436
29.11	System Center Virtual Machine Manager .....	437
29.12	PowerShell Management Library for Hyper-V (pshyperv) .....	438
29.13	Powershell Outlook Account Manager .....	439
29.14	PowerShell Configurator (PSConfig) .....	439
29.15	Weitere Erweiterungen .....	440
<b>30</b>	<b>Tipps und Tricks zur PowerShell .....</b>	<b>441</b>
30.1	Befehlsgeschichte .....	441
30.2	System- und Hostinformationen .....	442
30.3	Alle Anzeigen löschen .....	443
30.4	Anpassen der Eingabeaufforderung (Prompt) .....	443
30.5	Profileinstellungen für die PowerShell-Konsole .....	444
30.6	PowerShell als Gruppenrichtlinien-Skripte .....	448
30.7	Einblicke in die Interna der Pipeline-Verarbeitung .....	451
<b>Teil C: PowerShell im Praxiseinsatz .....</b>	<b>453</b>	
<b>31</b>	<b>Dateisystem .....</b>	<b>455</b>
31.1	Laufwerke .....	456
31.2	Ordnerinhalte .....	460
31.3	Kurznamen .....	462
31.4	Dateisystemoperationen .....	462
31.5	Dateieigenschaften lesen .....	463
31.6	Datei-Hash .....	464
31.7	Finden von Duplikaten .....	464
31.8	Dateieigenschaften verändern .....	467
31.9	Verknüpfungen im Dateisystem .....	469

31.10	Komprimierung	472
31.11	Dateisystemfreigaben	474
31.12	Überwachung des Dateisystems	483
<b>32</b>	<b>Festplattenverschlüsselung mit Bitlocker</b>	<b>485</b>
32.1	Übersicht über das BitLocker-Modul	486
32.2	Verschlüsseln eines Laufwerks	487
<b>33</b>	<b>Dokumente</b>	<b>489</b>
33.1	Textdateien	489
33.2	Binärdateien	490
33.3	CSV-Dateien	490
33.4	INI-Dateien	492
33.5	XML-Dateien	493
33.6	HTML-Dateien	501
<b>34</b>	<b>Datenbanken</b>	<b>503</b>
34.1	ADO.NET-Grundlagen	503
34.2	Beispieldatenbank	509
34.3	Datenzugriff mit den Bordmitteln der PowerShell	510
34.4	Datenzugriff mit den PowerShell-Erweiterungen	521
34.5	Datenbankzugriff mit SQLPS	524
34.6	Datenbankzugriff mit SQLPSX	524
<b>35</b>	<b>Microsoft SQL Server-Administration</b>	<b>525</b>
35.1	PowerShell-Integration im SQL Server Management Studio	526
35.2	SQL-Server-Laufwerk „SQLSERVER:“	527
35.3	Die SQLPS-Commandlets	530
35.4	Die SQL Server Management Objects (SMO)	532
35.5	SQLPSX	535
35.6	Microsoft SQL Server-Administration mit der PowerShell in der Praxis	543
<b>36</b>	<b>ODBC-Datenquellen</b>	<b>549</b>
36.1	ODBC-Treiber und -Datenquellen auflisten	550
36.2	Anlegen einer ODBC-Datenquelle	551
36.3	Zugriff auf eine ODBC-Datenquelle	552
<b>37</b>	<b>Registrierungsdatenbank (Registry)</b>	<b>555</b>
37.1	Schlüssel auslesen	555
37.2	Schlüssel anlegen und löschen	556
37.3	Laufwerke definieren	556
37.4	Werte anlegen und löschen	557

37.5	Werte auslesen .....	558
37.6	Praxisbeispiel .....	558
<b>38</b>	<b>Computerverwaltung .....</b>	<b>561</b>
38.1	Computerinformationen .....	561
38.2	Computername und Domäne .....	563
38.3	Herunterfahren und Neustarten .....	564
38.4	Wiederherstellungspunkte verwalten .....	564
<b>39</b>	<b>Hardwareverwaltung .....</b>	<b>567</b>
39.1	Hardwarebausteine .....	567
39.2	Druckerverwaltung (ältere Betriebssysteme) .....	569
39.3	Druckerverwaltung (Windows 8.x und Windows Server 2012/2012 R2) .....	570
<b>40</b>	<b>Softwareverwaltung .....</b>	<b>573</b>
40.1	Softwareinventarisierung .....	573
40.2	Installation von Anwendungen .....	576
40.3	Deinstallation von Anwendungen .....	577
40.4	Praxisbeispiel: Installationstest .....	577
40.5	Versionsnummer ermitteln .....	578
40.6	Servermanager .....	579
40.7	Softwareeinschränkungen mit dem PowerShell-Modul „AppLocker“ .....	590
<b>41</b>	<b>Prozessverwaltung .....</b>	<b>597</b>
41.1	Prozesse auflisten .....	597
41.2	Prozesse starten .....	598
41.3	Prozesse beenden .....	600
41.4	Warten auf das Beenden einer Anwendung .....	600
<b>42</b>	<b>Systemdienste .....</b>	<b>603</b>
42.1	Dienste auflisten .....	603
42.2	Dienstzustand ändern .....	605
42.3	Diensteigenschaften ändern .....	606
<b>43</b>	<b>Netzwerk .....</b>	<b>607</b>
43.1	Netzwerkconfiguration (ältere Betriebssysteme) .....	607
43.2	Netzwerkconfiguration (ab Windows 8.x und Windows Server 2012 (inkl. R2) .....	609
43.3	DNS-Client-Konfiguration .....	611
43.4	DNS-Namensauflösung .....	615
43.5	Erreichbarkeit prüfen (Ping) .....	616
43.6	Windows Firewall .....	618

43.7	E-Mails senden (SMTP) .....	624
43.8	Abruf von Daten von einem HTTP-Server .....	626
43.9	Aufrufe von SOAP-Webdiensten .....	627
<b>44</b>	<b>Ereignisprotokolle .....</b>	<b>631</b>
44.1	Protokolleinträge auslesen .....	631
44.2	Ereignisprotokolle erzeugen .....	633
44.3	Protokolleinträge erzeugen .....	633
44.4	Protokollgröße festlegen .....	633
44.5	Protokolleinträge löschen .....	633
<b>45</b>	<b>Leistungsdaten .....</b>	<b>635</b>
45.1	Zugriff auf Leistungsindikatoren über WMI .....	635
45.2	Get-Counter .....	636
<b>46</b>	<b>Sicherheitseinstellungen .....</b>	<b>639</b>
46.1	Grundlagen .....	639
46.2	Zugriffsrechtlisten auslesen .....	644
46.3	Einzelne Rechteinträge auslesen .....	646
46.4	Besitzer auslesen .....	647
46.5	Benutzer und SID .....	647
46.6	Hinzufügen eines Rechteintrags zu einer Zugriffsrechtliste .....	651
46.7	Entfernen eines Rechteintrags aus einer Zugriffsrechtliste .....	653
46.8	Zugriffsrechtliste übertragen .....	655
46.9	Zugriffsrechtliste über SDDL setzen .....	655
<b>47</b>	<b>Active Directory .....</b>	<b>657</b>
47.1	Benutzer- und Gruppenverwaltung mit WMI .....	658
47.2	Einführung in System.DirectoryServices .....	659
47.3	Basiseigenschaften .....	670
47.4	Benutzer- und Gruppenverwaltung im Active Directory .....	672
47.5	Verwaltung der Organisationseinheiten .....	680
47.6	Suche im Active Directory .....	681
47.7	Navigation im Active Directory mit den PowerShell Extensions .....	688
47.8	Verwendung der Active-Directory-Erweiterungen von www.IT-Visions.de .....	689
47.9	PowerShell-Modul „Active Directory“ (ADPowerShell) .....	691
47.10	PowerShell-Modul „ADSDeployment“ .....	710
47.11	Informationen über die Active-Directory-Struktur .....	713
<b>48</b>	<b>Gruppenrichtlinien .....</b>	<b>717</b>
48.1	Verwaltung der Gruppenrichtlinien .....	718
48.2	Verknüpfung der Gruppenrichtlinien .....	719

48.3	Gruppenrichtlinienberichte .....	721
48.4	Gruppenrichtlinienvererbung .....	722
48.5	Weitere Möglichkeiten .....	723
<b>49</b>	<b>Virtuelle Systeme mit Hyper-V .....</b>	<b>725</b>
49.1	Das Hyper-V-Modul von Microsoft .....	726
49.2	Die ersten Schritte mit dem Hyper-V-Modul .....	728
49.3	Virtuelle Maschinen anlegen .....	732
49.4	Umgang mit virtuellen Festplatten .....	738
49.5	Konfiguration virtueller Maschinen .....	741
49.6	PowerShell Management Library for Hyper-V (für ältere Betriebssysteme) ....	745
<b>50</b>	<b>Internet Information Server (IIS) .....</b>	<b>749</b>
50.1	Überblick .....	749
50.2	Navigationsprovider .....	751
50.3	Anlegen von Websites .....	753
50.4	Massenanlegen von Websites .....	754
50.5	Ändern von Eigenschaften von Websites .....	756
50.6	Anwendungspool anlegen .....	757
50.7	Virtuelle Verzeichnisse und IIS-Anwendungen .....	758
50.8	Website-Zustand ändern .....	758
50.9	Anwendungspools starten und stoppen .....	759
50.10	Löschen von Websites .....	759
<b>51</b>	<b>Microsoft Exchange Server .....</b>	<b>761</b>
51.1	Daten abrufen .....	761
51.2	Postfächer verwalten .....	762
51.3	Öffentliche Ordner verwalten .....	763
<b>52</b>	<b>Optimierungen und Problemlösungen .....</b>	<b>765</b>
52.1	PowerShell-Modul „TroubleshootingPack“ .....	765
52.2	PowerShell-Modul „Best Practices“ .....	769
<b>53</b>	<b>Grafische Benutzeroberflächen .....</b>	<b>771</b>
53.1	Eingabemasken .....	771
53.2	Universelle Objektdarstellung .....	773
53.3	Zwischenablage .....	774
53.4	WPF PowerShell Kit (WPK) .....	775
<b>54</b>	<b>Entwicklung von Commandlets in der PowerShell-Sprache .....</b>	<b>787</b>
54.1	Aufbau eines skriptbasierten Commandlets .....	787
54.2	Parameterfestlegung .....	790

54.3	Auszeichnung der Parameterdefinitionen .....	795
54.4	Dokumentation .....	798
<b>Teil D: Profiwissen – Erweitern der PowerShell .....</b>		<b>785</b>
<b>55</b>	<b>Entwicklung eigener Commandlets mit C# .....</b>	<b>801</b>
55.1	Technische Voraussetzungen .....	801
55.2	Grundkonzept der .NET-basierten Commandlets .....	803
55.3	Schrittweise Erstellung eines minimalen Commandlets .....	805
55.4	Erstellung eines Commandlets mit einem Rückgabeobjekt .....	813
55.5	Erstellung eines Commandlets mit mehreren Rückgabeobjekten .....	814
55.6	Erstellen eines Commandlets mit Parametern .....	818
55.7	Verarbeiten von Pipeline-Eingaben .....	820
55.8	Verkettung von Commandlets .....	823
55.9	Fehlersuche in Commandlets .....	827
55.10	Statusinformationen .....	830
55.11	Unterstützung für Sicherheitsabfragen .....	835
55.12	Festlegung der Hilfeinformationen .....	837
55.13	Erstellung von Commandlets für den Zugriff auf eine Geschäftsanwendung ...	841
55.14	Konventionen für Commandlets .....	842
55.15	Weitere Möglichkeiten .....	844
<b>56</b>	<b>Hosting der Windows PowerShell .....</b>	<b>845</b>
56.1	Voraussetzungen für das Hosting .....	846
56.2	Hosting mit PSHost .....	846
56.3	Vereinfachtes Hosting seit PowerShell 2.0 .....	850
<b>57</b>	<b>PowerShell-Module erstellen .....</b>	<b>853</b>
57.1	Erstellen eines Skriptmoduls .....	853
57.2	Erstellen eines Moduls mit Binärdateien .....	855
57.3	Erstellen eines Moduls mit Manifest .....	855
<b>Anhang .....</b>		<b>863</b>
<b>Crashkurs „Objektorientierung“ .....</b>		<b>865</b>
<b>Crashkurs „.NET Framework“ .....</b>		<b>873</b>
<b>Literatur .....</b>		<b>881</b>
<b>Weitere Informationen im Internet .....</b>		<b>885</b>
<b>Stichwortverzeichnis .....</b>		<b>887</b>



# Vorwort zur vierten Auflage 2014

Liebe Leserin, lieber Leser,

willkommen zur vierten Auflage dieses PowerShell-Buchs! Das vor Ihnen liegende Buch behandelt die Windows-PowerShell-Version 4.0 von Microsoft sowie ergänzende Werkzeuge von Microsoft und Drittanbietern (z. B. PowerShell Community Extensions, PowerShellPack, PowerShellPlus).

## ■ Wer bin ich?

Mein Name ist Holger Schwichtenberg, ich bin derzeit 40 Jahre alt und habe im Fachgebiet Wirtschaftsinformatik promoviert. Ich lebe (in Essen, der europäischen Kulturhauptstadt 2010) davon, dass mein Team und ich im Rahmen unserer Firma *www.IT-Visions.de* anderen Unternehmen bei der Entwicklung von .NET- und PowerShell-Anwendungen beratend und schulend zur Seite stehen. Zudem entwickeln wir im Rahmen der 5minds IT-Solutions GmbH & Co. KG Software im Auftrag von Unternehmen.

Es ist mein Hobby und „Nebenberuf“, IT-Fachbücher zu schreiben. Dieses Buch ist, unter Mitzählung aller nennenswerten Neuauflagen, das 59. Buch, das ich allein oder mit Co-Autoren geschrieben habe. Meine weiteren Hobbys sind Mountain Biking, Fotografie und Reisen. Natürlich verstehe ich das Bücherschreiben auch ein wenig als Werbung für die Arbeit unseres Unternehmens und wir hoffen, dass der ein oder andere von Ihnen uns beauftragen wird, Sie durch Beratung, Schulung und Auftragsentwicklung zu unterstützen.

## ■ Wer sind Sie?

Damit Sie den optimalen Nutzen aus diesem Buch ziehen können, möchte ich – so genau es mir möglich ist – beschreiben, an wen sich dieses Buch richtet. Hierzu habe ich einen Fragebogen ausgearbeitet, mit dem Sie schnell erkennen können, ob das Buch für Sie geeignet ist.

Sind Sie Systemadministrator in einem <b>homogenen</b> Windows-Netzwerk?	<input type="radio"/> Ja <input type="radio"/> Nein
Laufen die für Sie relevanten Computer mit den von PowerShell 4.0 unterstützten Betriebssystemen? (Windows 7, Windows 8/8.1, Windows Server 2008 R2/2012/2012 R2)	<input type="radio"/> Ja <input type="radio"/> Nein
Sie besitzen zumindest rudimentäre Grundkenntnisse im Bereich des (objektorientierten) Programmierens?	<input type="radio"/> Ja <input type="radio"/> Nein
Wünschen Sie einen kompakten Überblick über die Architektur, Konzepte und Anwendungsfälle der PowerShell?	<input type="radio"/> Ja <input type="radio"/> Nein
Sie können auf Schritt-für-Schritt-Anleitungen verzichten?	<input type="radio"/> Ja <input type="radio"/> Nein
Sie können auf formale Syntaxbeschreibungen verzichten und lernen lieber an aussagekräftigen Beispielen?	<input type="radio"/> Ja <input type="radio"/> Nein
Sie erwarten nicht, dass in diesem Buch alle Möglichkeiten der PowerShell detailliert beschrieben werden?	<input type="radio"/> Ja <input type="radio"/> Nein
Sind Sie, nachdem Sie ein Grundverständnis durch dieses Buch gewonnen haben, bereit, Detailfragen in der Dokumentation der PowerShell, von .NET und WMI nachzuschlagen, da das Buch auf 1000 Seiten nicht alle Details erläutern kann?	<input type="radio"/> Ja <input type="radio"/> Nein

Wenn Sie alle obigen Fragen mit „Ja“ beantwortet haben, ist das Buch richtig für Sie. In anderen Fällen sollten Sie sich erst mit einführender Literatur beschäftigen.

## ■ Was ist neu in diesem Buch?

Gegenüber der vorherigen Auflage zur PowerShell 3.0 wurde das Buch um die neuen Funktionen in PowerShell 4.0 erweitert (z. B. ein Kapitel zur Desired State Configuration) und an einigen Stellen inhaltlich optimiert. Neu ist der Verlag: Die ersten drei Auflagen dieses Buchs erschienen bei Addison-Wesley. Dieser Verlag hat aber zum 31.12.2013 seinen Betrieb eingestellt. Nun hat das Buch beim Carl Hanser Verlag eine schöne neue Heimat gefunden.

## ■ Wem ist zu danken?

Folgenden Personen möchte ich meinen Dank für ihre Mitwirkung an diesem Buch aussprechen:

- meinem Kollegen und Freund Peter Monadjemi, der rund 100 Seiten mit Beispielen zu diesem Buch beigetragen hat (Themen: Workflows, Bitlocker, ODBC, Hyper-V, DNS-Client, Firewall und SQL-Server-Administration),

- Frau Bauer-Schiewek, die dieses Buchprojekt beim Carl Hanser Verlag koordiniert und vermarktet,
- Frau Petra Kienle, die meine Tippfehler gefunden und sprachliche Ungenauigkeiten eliminiert hat,
- meiner Frau und meinen Kindern dafür, dass sie mir das Umfeld geben, um neben meinem Hauptberuf an Büchern wie diesem zu arbeiten.

## ■ Woher bekommen Sie die Beispiele aus diesem Buch?

Unter <http://www.powershell-doktor.de/leser> biete ich ein **ehrenamtlich betriebenes** Webportal für Leser meiner Bücher an. In diesem Portal können Sie

- die Codebeispiele aus diesem Buch herunterladen,
- eine PowerShell-Kurzreferenz „Cheat Sheet“ (zwei DIN-A4-Seiten als Hilfe für die tägliche Arbeit) kostenlos herunterladen,
- Feedback zu diesem Buch geben (Bewertung abgeben und Fehler melden) und
- technische Fragen in einem Webforum stellen.

Alle registrierten Leser erhalten auch Einladungen zu kostenlosen Community-Veranstaltungen sowie Vergünstigungen bei unseren öffentlichen Seminaren zu .NET und zur PowerShell. Bei der Registrierung müssen Sie das Kennwort **Aeon Flux** angeben.

## ■ Wo kann ich mich schulen oder beraten lassen?

Unter der E-Mail-Adresse [buero@IT-Visions.de](mailto:buero@IT-Visions.de) stehen mein Team und ich für Anfragen bezüglich Schulung, Beratung und Entwicklungstätigkeiten zur Verfügung.

## ■ Zum Schluss des Vorworts ...

wünsche ich Ihnen viel Spaß und Erfolg mit der PowerShell 4.0!

*Dr. Holger Schwichtenberg*

*Essen-Byfang im Januar 2014*



# Über den Autor

## Dr. Holger Schwichtenberg



[www.IT-Visions.de](http://www.IT-Visions.de)  
Dr. Holger Schwichtenberg



- Studienabschluss Diplom-Wirtschaftsinformatik an der Universität Essen
- Promotion an der Universität Essen im Gebiet komponentenbasierter Softwareentwicklung
- Seit 1996 selbstständig als unabhängiger Berater, Dozent, Softwarearchitekt und Fachjournalist
- Leiter des Berater- und Dozententeams bei *www.IT-Visions.de*
- Leitung der Softwareentwicklung im Bereich Microsoft/.NET bei der 5minds IT-Solutions GmbH & Co. KG ([www.5minds.de](http://www.5minds.de))
- 59 Fachbücher beim Carl Hanser Verlag, bei Microsoft Press und Addison-Wesley sowie mehr als 750 Beiträge in Fachzeitschriften
- Gutachter in den Wettbewerbsverfahren der EU gegen Microsoft (2006–2009)
- Ständiger Mitarbeiter der Zeitschriften iX (seit 1999), dotnetpro (seit 2000) und Windows Developer (seit 2010) sowie beim Online-Portal *heise.de* (seit 2008)
- Regelmäßiger Sprecher auf nationalen und internationalen Fachkonferenzen (z.B. TechEd, Microsoft IT Forum, BASTA, BASTA-on-Tour, Advanced Developers Conference, .NET-Entwicklerkonferenz, OOP, VS One, Wirtschaftsinformatik, Net.Object Days, Windows Forum, DOTNET-Konferenz, XML-in-Action)
- Zertifikate und Auszeichnungen von Microsoft:
  - Microsoft Most Valuable Professional (MVP)
  - Microsoft Certified Solution Developer (MCSD)

- Thematische Schwerpunkte:
  - Microsoft .NET Framework, Visual Studio, C#, Visual Basic
  - .NET-Architektur/Auswahl von .NET-Technologien
  - Einführung von .NET Framework und Visual Studio/Migration auf .NET
  - Webanwendungsentwicklung mit IIS, ASP.NET und JavaScript/AJAX
  - Enterprise .NET, verteilte Systeme/Webservices mit .NET
  - Relationale Datenbanken, XML, Datenzugriffsstrategien
  - Objektrelationales Mapping (ORM), insbesondere ADO.NET Entity Framework
  - Windows PowerShell (WPS) und Windows Management Instrumentation (WMI)
- Ehrenamtliche Community-Tätigkeiten:
  - Vortragender für die International .NET Association (INETA)
  - Betrieb diverser Community-Websites: [www.dotnetframework.de](http://www.dotnetframework.de), [www.entwickler-lexikon.de](http://www.entwickler-lexikon.de), [www.windows-scripting.de](http://www.windows-scripting.de), [www.aspnetdev.de](http://www.aspnetdev.de) u. a.
- Firmenwebsites: <http://www.IT-Visions.de> und <http://www.5minds.de>
- Weblog: <http://www.dotnet-doktor.de>
- Kontakt: [hs@IT-Visions.de](mailto:hs@IT-Visions.de) sowie Telefon 02 01-64 95 90-90

## Über den Co-Autor Peter Monadjemi



Peter Monadjemi ist freiberuflicher Trainer und Autor mit den Schwerpunkten PowerShell, Windows-Server-Administration und Software-Entwicklung auf der Grundlage des .NET Framework. Er kennt die PowerShell bereits seit der Zeit, in der sie als Vorabversion noch unter dem Projektnamen „Monad“ lief – ein Codename, der für ihn eine Art „Wink des Schicksals“ war. Seitdem hat er viel Zeit mit der PowerShell verbracht und Hunderte von Skripten geschrieben. Er lebt in Esslingen am Neckar.

# A

## Teil A: PowerShell-Basiswissen

Dieser Buchteil informiert über die Basiskonzepte der PowerShell, insbesondere Commandlets, Pipelines, Navigation und Skripte. Außerdem werden am Ende dieses Teils Werkzeuge vorgestellt.



# 1

## Erste Schritte mit der Windows PowerShell

Das DOS-ähnliche Kommandozeilenfenster hat viele Windows-Versionen in beinahe unveränderter Form überlebt. Mit der Windows PowerShell (WPS) besitzt Microsoft nun endlich einen Nachfolger, der es mit den Unix-Shells aufnehmen kann und diese in Hinblick auf Eleganz und Robustheit in einigen Punkten auch überbieten kann. Die PowerShell ist eine Adaption des Konzepts von Unix-Shells auf Windows unter Verwendung des .NET Frameworks und mit Anbindung an die Windows Management Instrumentation (WMI).

### ■ 1.1 Was ist die Windows PowerShell?

In einem Satz: Die Windows PowerShell (WPS) ist eine neue, .NET-basierte Umgebung für interaktive Systemadministration und Scripting auf der Windows-Plattform.

Die Kernfunktionen der PowerShell sind:

- Zahlreiche eingebaute Befehle, die „Commandlets“ genannt werden
- Zugang zu allen Systemobjekten, die durch COM-Bibliotheken, das .NET Framework und die Windows Management Instrumentation (WMI) bereitgestellt werden
- Robuster Datenaustausch zwischen Commandlets durch Pipelines basierend auf typisierten Objekten
- Ein einheitliches Navigationsparadigma für verschiedene Speicher (z.B. Dateisystem, Registrierungsdatenbank, Zertifikatsspeicher, Active Directory und Umgebungsvariablen)
- Eine einfach zu erlernende, aber mächtige Skriptsprache mit wahlweise schwacher oder starker Typisierung
- Ein Sicherheitsmodell, das die Ausführung unerwünschter Skripte unterbindet
- Integrierte Funktionen für Ablaufverfolgung und Debugging
- Die PowerShell kann um eigene Befehle erweitert werden.
- Die PowerShell kann in eigene Anwendungen integriert werden (Hosting).

## ■ 1.2 Geschichte der Windows PowerShell

Das Active Scripting ist einigen Administratoren zu komplex, weil es viel Wissen über objektorientiertes Programmieren und das Component Object Model (COM) voraussetzt. Die vielen Ausnahmen und Ungereimtheiten im Active Scripting erschweren das Erlernen von Windows Script Host (WSH) und der zugehörigen Komponentenbibliotheken.

Schon im Zuge der Entwicklung des Windows Server 2003 gab Microsoft zu, dass man Unix-Administratoren zum Interview über ihr tägliches Handwerkszeug gebeten hatte. Das kurzfristige Ergebnis war eine große Menge zusätzlicher Kommandozeilenwerkzeuge. Langfristig setzt Microsoft jedoch auf eine Ablösung des DOS-ähnlichen Konsolenfensters durch eine neue Scripting-Umgebung.

Mit dem Erscheinen des .NET Frameworks im Jahre 2002 wurde lange über einen WSH.NET spekuliert. Microsoft stellte jedoch die Neuentwicklung des WSH für das .NET Framework ein, als abzusehen war, dass die Verwendung von .NET-basierten Programmiersprachen wie C# und Visual Basic .NET dem Administrator nur noch mehr Kenntnisse über objektorientierte Softwareentwicklung abverlangen würde.

Microsoft beobachtete in der Unix-Welt eine hohe Zufriedenheit mit den dortigen Kommandozeilen-Shells und entschloss sich daher, das Konzept der Unix-Shells, insbesondere das Pipelining, mit dem .NET Framework zusammenzubringen und daraus eine .NET-basierte Windows Shell zu entwickeln. Diese ist so einfach wie eine Unix-Shell, aber kann so mächtig wie das .NET Framework sein.

In einer ersten Beta-Version wurde die neue Shell schon unter dem Codenamen „Monad“ auf der Professional Developer Conference (PDC) im Oktober 2003 in Los Angeles vorgestellt. Nach den Zwischenstufen „Microsoft Shell (MSH)“ und „Microsoft Command Shell“ trägt die neue Skriptumgebung seit Mai 2006 den Namen „Windows PowerShell“.

Die PowerShell 1.0 erschien am 6.11.2006 zeitgleich mit Windows Vista, war aber dort nicht enthalten, sondern muss heruntergeladen und nachinstalliert werden.

Die PowerShell 2.0 ist zusammen mit Windows 7/Windows Server 2008 R2 erschienen am 22.7.2009.

Die PowerShell 3.0 ist zusammen mit Windows 8/Windows Server 2012 erschienen am 15.8.2012.

Die PowerShell 4.0 ist zusammen mit Windows 8.1/Windows Server 2012 R2 am 9.9.2013 erschienen.

## ■ 1.3 Motivation zur Windows PowerShell

Falls Sie eine Motivation brauchen, sich mit der PowerShell zu beschäftigen, wird dieses Kapitel sie Ihnen liefern. Es stellt die Lösung für eine typische Scripting-Aufgabe sowohl im „alten“ Windows Script Host (WSH) als auch in der „neuen“ Windows PowerShell vor.