



Anwendung und Nutzen von Microsoft PowerShell in der modernen Systemadministration

Die moderne Antwort auf die Frage: „Wie setze ich das in der IT-Administration effizient um?“ lautet in wachsendem Maße: „Mit PowerShell“. Viele Unternehmen sind auf dem Weg dieses mächtige Framework von Microsoft in voller Leistungsfähigkeit zu nutzen, aber der Weg ist steinig. PowerShell ist nicht trivial zu erschließen und glänzt mit vielen Hürden, teils aus Sicherheitsgründen, teils weil es nicht für eine Anwendung durch Jedermann gedacht ist. Ein Wissenstransfer ist aufwendig und so bleibt die Anwendung oft den hochqualifizierten Administratoren vorbehalten, was wiederum die Nutzung in der Breite einschränkt.

holistic-net nutzt die native Microsoft PowerShell in der täglichen Projektarbeit, insbesondere in Migrationsprojekten, dem Aufbau von Exchange Infrastrukturen on-premises oder in der Administration von O365. Wir erleben dabei die volle Wirksamkeit der PowerShell und machen uns diese nutzbar. Es erfordert jedoch endlos viel Disziplin PowerShell-Know-How in Form von PowerShell Scripts geeignet weiterzugeben. Immer wieder wird die Frage gestellt:

„Wer weiß, wie ein bestimmtes Problem per PowerShell zu lösen ist? Da gab es doch mal ein Script. Wo ist das? Passt das für die aktuelle Anforderung? Wer versteht es? Wer kann es anpassen? Wer darf es ausführen?“

Bei vielen Unternehmen bewirkt dies eine Abwehrhaltung, die den nutzbringenden Einsatz der PowerShell verhindert. Man weicht dem aus, indem zur Maus gegriffen wird. Und dann klingt es so:

„Ständig muss ich gleiche Clicks wiederholen, gibt es da nichts komfortableres?“ Gibt es!



„Mit dem **holistic powershell launch center** können Sie ihr Personal effizient einsetzen, ihre Systeme schnell und flexibel verwalten und Risiken in der täglichen Systemverwaltung minimieren.“

Welchen Grundansatz verfolgt die Microsoft PowerShell?

In erster Linie ist die PowerShell als ein von Microsoft frei angebotenes plattformübergreifendes Framework zur Bewältigung grundlegender administrativer Aufgaben in Windows-Systemen zu verstehen. Ab Version 6 unterstützt PowerShell zusätzlich die Plattformen Linux und MacOS und entwickelt sich in dieser Hinsicht zu einem mächtigen Werkzeug.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Alle Rechte vorbehalten.

Lernen Sie das neue plattformübergreifende PowerShell kennen - https://aka.ms/pscore6

PS C:\Users\Günter Mohren.HOLISTIC-NET>
```

(PowerShell Host)

Die im Framework zur Verfügung gestellten Funktionen ermöglichen es dem Anwender, anstehende Aufgaben zur Konfiguration und Verwaltung der Plattformen durch das Absetzen von Befehlen in der PowerShell eigenen Benutzerschnittstelle, dem PowerShell Host ‚Konsole‘, umzusetzen. Die PowerShell Engine ist der Kommandozeileninterpreter, der die Eingaben verarbeitet und damit das Herz der PowerShell bildet. Die Eingabebefehle werden Cmdlets (gesprochen Commandlets) genannt.



Gleiche Funktionen sind per Mausklick in den Systemeinstellungen von Windows meist komplizierter, manchmal gar nicht umzusetzen. Administrationsaufgaben gestalten sich mit Kenntnis der passenden Befehle und Übergabeparameter durch die PowerShell wesentlich effizienter in der Ausführung. Die Anzahl der vorinstallierten Cmdlets steigt stetig mit jeder neu ausgerollten PowerShell Version.

So arbeiten Sie **out of the box**:

```

111
112
113 $users | ForEach-Object {
114
115     $firstName = $_.givenName
116     $lastName = $_.surName
117     $displayName = $_.displayName
118     $samAN = $_.sAMAccountName
119     $SMTPAddress = $_.primarySMTP
120     $userPassword = $_.initialPassword
121
122
123     $scn = $displayName
124     $relativDistinguishedName = "CN=$scn"
125     $distinguishedName = "$relativDistinguishedName,$$searchBase"
126
127     $SUPN = $SMTPAddress
128
129     $SMTPDomain = $SMTPAddress.Split('@')[1]
130     $password = ConvertTo-SecureString $userPassword -AsPlainText -Force
131
132
133     # Check whether SAM-Account-Name is already in use in specified AD-Domain
134     # -----
135
136     $userCheck = Get-ADObject -Filter "samAccountName -eq '$samAN'" -Server $DCServer -Properties * -I
137
138     if ($userCheck -ne $null)
139     {
140         Write-Host "Check of user's AD object status shows conspicuity..."
141         Write-Host "Info: -> The specified user with SAM-Account-Name '$samAN' is already existing in
142
143         # AD
144         Remove-PSsession -session $SessionAD
145

```

So können Sie mit dem **holistic powershell launch center** arbeiten:

1 Input **2** Output

Create AD User (SelectionListsExam) Info

Creates a new user in selectable AD-Paths as a simple exam for Selection-Lists

TicketNo. Enable notification by email

First Name Last Name Display-Name SAM-Account-Name

SMTP Address Password Active Directory Path

OU=HPLC,DC=holistic-dev,DC=de

undefined

Account & Password Settings

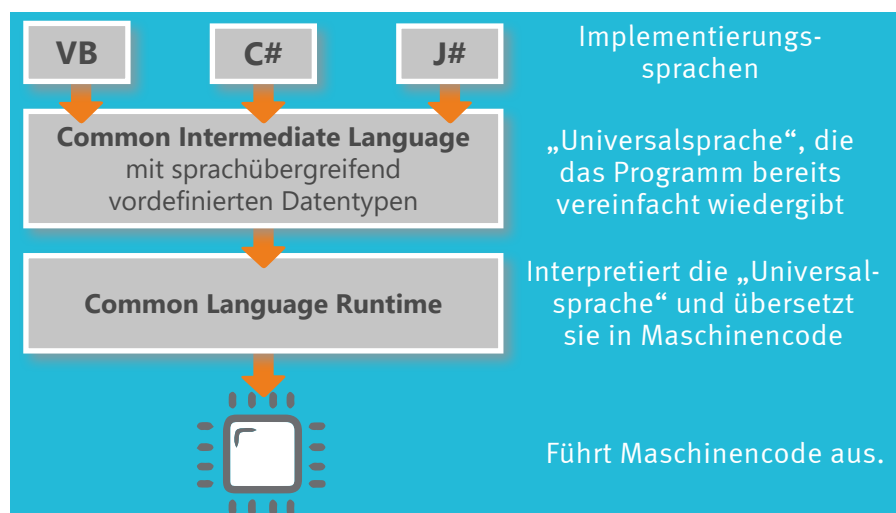
- User has to change PW at first login and frequently
- User is free to change PW - PW never expires
- User is free to change PW - PW expires
- User can not change PW - PW never expires
- User can not change PW - PW expires



Seine volle Stärke aber zeigt die PowerShell in der Möglichkeit zur Automatisierung abzubildender Use Cases. Dazu können komplexe Skript-Programme mit der eigens entwickelten PowerShell Scripting Language entwickelt werden. Die Eingabe erfolgt in einem weiteren PowerShell Host, der sogenannten PowerShell ISE (Integrated Scripting Environment, kurz ISE). Die ISE ist ein modernes Eingabefenster mit integriertem Skripteditor, Debugger und IntelliSense. Die PowerShell Standard-Cmdlets nehmen Cmdlet-Entwicklern Arbeit von Standardaufgaben ab und vereinfachen und beschleunigen die Entwicklung wesentlich.

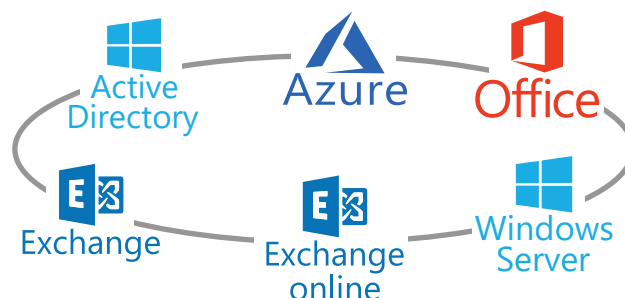
Der Anwender kann mit PowerShell-Skripts die ganze Stärke des .NET-Frameworks nutzen.

PowerShell arbeitet mit Objekten. Cmdlets sollen genau eine spezifische Aufgabe erfüllen und nur einen Objekttyp als Ergebnis ausgeben. Die Cmdlet-Befehlsbibliothek der PowerShell Engine bildet sich aus einer Sammlung von .NET-Klassen der objektorientierten Programmiersprache C# und basiert auf der Common Language Runtime (CLR) des .NET Frameworks. Die PowerShell erlaubt Zugriff auf WMI-Klassen, COM-Objekte sowie auf das gesamte .NET Framework.



PowerShell verwaltet die ganze Welt der Microsoft Dienste

Die durch die Installation der PowerShell zur Verfügung gestellten Befehle beschränken sich nicht nur auf Funktionalitäten zur Verwaltung von Windows Clients & Servern. Die Befehls-Library steuert zusätzlich u.a. die Verzeichnisdienste Active Directory & Azure AD, Exchange ,on premise/online‘ oder auch alle Teile des Microsoft Online Dienstes Office 365.





Kontrolle und Sicherheit stehen an oberster Stelle

Seine Stärken spielt die PowerShell aber in der Eigenschaft aus, Standardtätigkeiten und ebenso sehr komplexe Abläufe zu automatisieren. Die Abläufe laufen damit immer wieder in gleicher Form und unter voller Kontrolle der einzelnen Prozessschritte ab. Fehleinstellungen der Systeme werden ausgeschlossen und die Sicherheit des Betriebs der Systeme deutlich erhöht. Gleichzeitig steigert sich die Effizienz des Ressourceneinsatzes. Auch weniger ausgebildete Anwender können mit PowerShell Skripten in geringerer Zeit, bei maximaler Sicherheit und Kontrolle, die ihnen anvertraute Infrastruktur betreiben. Die Erstellung von Skripten kann ressourcenseitig von der Anwendung der Skripts getrennt werden.



holistic powershell launch center (hplc)

ist die Antwort auf die Herausforderung vieler Unternehmen, die eigene Systemlandschaft effizient, kontrolliert und vor allem mit ausreichender Sicherheit zu betreiben.

Aus wertvollen Erfahrungen entwickelte holistic-net eine Idee

Seit vielen Jahren unterstützt holistic-net eine Vielzahl seiner Kunden erfolgreich in der Migration von HCL Notes zu Microsoft Outlook. Basis dieses Erfolgs ist das Vorhandensein exzellenter Erfahrungen in der Installation und dem Betrieb von Infrastrukturen auf Basis Notes/Domino und Outlook/Exchange. Wer effizient sein will, der benötigt exzellente Tools und das Wissen sinnvoll anzuwendender Migrationspfade. Migrationen jeglicher Größenordnung verlangen nach Sicherheit, Kontrolle und Effizienz. Ohne die kompetente Nutzung von PowerShell, könnte holistic-net seinen Kunden diese Unterstützung nicht bieten. Hier spielt die PowerShell all ihre Stärken voll aus, die holistic-net zu nutzen weiß.

Wichtige Use Cases mit hohem Sicherheitsbedarf

- User Provisioning, z.B. Onboarding & Offboarding
- Resource Management, z.B. Besprechungsraumverwaltung
- Mailboxen und Verteilergruppen einrichten
- Abwesenheitsnachrichten für Mailboxen einstellen
- Windows Services verwalten

Das Potential der PowerShell endet jedoch nicht mit dem Abschluss einer Migration. Für unsere Kunden öffnet sich eine neue Welt, die es zu verwalten gilt.

Das Active Directory wird in seinem Spektrum an Einträgen erweitert, die Exchange-Umgebung kommt hinzu. Es stellen sich den System-Administratoren vielfältige Aufgaben, die in wiederkehrenden Abläufen sicher zu prozessieren sind.



holistic powershell launch center (hplc) ist die Lösung für die tägliche moderne Systemadministration. hplc ermöglicht Jedem die kontrollierte Ausführung von PowerShell Skripten, ohne dass PowerShell Kompetenz nötig ist.

Mehr Sicherheit, Kontrolle & Effizienz ist nicht möglich!



Die meisten Unternehmen, unabhängig von ihrer Größe, können das Wissen über die Administration aller in Nutzung befindlicher Zielsysteme nur schwer vorhalten und auf mehrere Schultern verteilen. Das gilt nicht nur für die Firmen, die eine Migration der Mailing-Plattform abgeschlossen haben. Insbesondere für viele mittelständische Unternehmen ist dies eine Herausforderung. Systeme entwickeln sich, werden in ihrer Funktion und Anwendung komplexer, aber auch mächtiger. Es ist stets auf eine sichere Ausführung aller Vorgänge zu achten. Verständlich, wenn die IT-Abteilungen vieler Unternehmen nicht in der Lage sind, die nötigen Ressourcen und vor allem das unbedingt notwendige fachliche Wissen vorzuhalten. Weit entfernt scheint vielfach das Ziel, diese Kompetenzen redundant im Zugriff zu haben.

holistic-net konnte durch die eigene praktische Entwicklung und Anwendung teils komplexer PowerShell Skripte, das breite Spektrum der Vorteile der PowerShell in der täglichen operativen Betreuung der wichtigsten Systeme erfahren. Aus dieser wertvollen Erfahrung heraus wurde eine Idee geboren.

Unser Produkt holistic powershell launch center nutzt die wesentlichen Möglichkeiten & Eigenschaften der PowerShell und kann diese gebündelt in einer anwenderfreundlichen Web-UI Applikation unseren Kunden zum Vorteil weiterreichen.

The screenshot displays the hplc web application interface. On the left is a dark sidebar with the 'hplc' logo and user information for 'Peter Meier@holistic-dev.de'. Below this is a 'Scripts' section with a search bar and a list of categories: AAD, AD Group, AD User and Contact, Exchange, Server, Teams, Test, and Tools. The main content area has a dark blue header with the 'hplc' title and 'holistic powershell launch center' subtitle. Below the header is a terminal window showing PowerShell commands and their outputs:

```
PS COMPANYY > hplc --version
hplc Version 1.0.8062.28026

PS COMPANYY> hplc --help
This tool provides a collection of helpful PowerShell scripts

Usage:
Click on a category on the left side to get an overview of available scripts
Then, click on a specific script to fill a script with parameters
Finally, click on the 'Run' button to execute the script

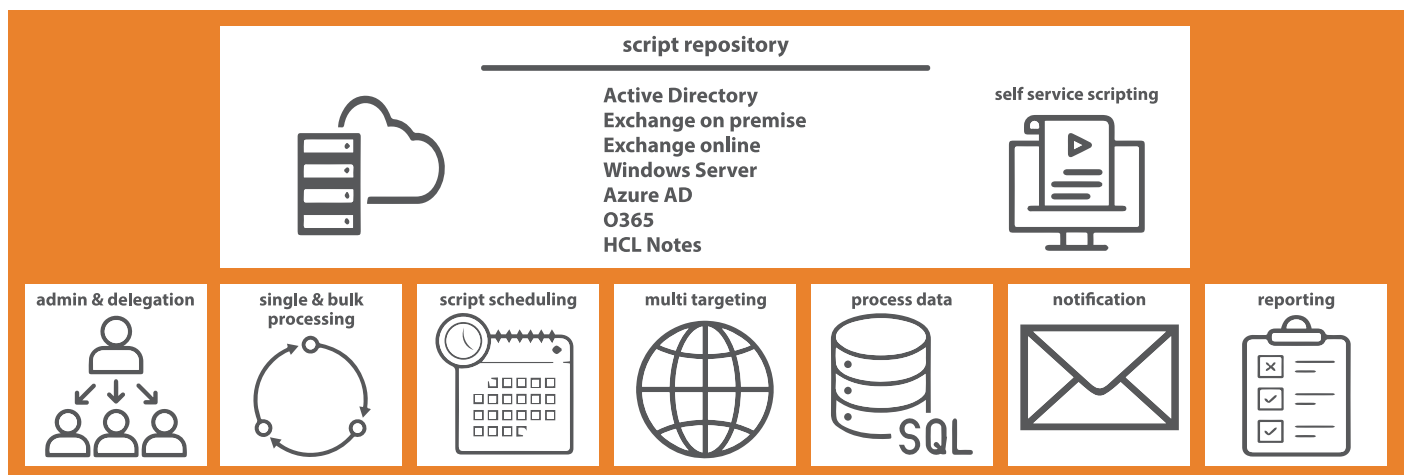
Good luck and be careful :)

PS COMPANYY>
```

Below the terminal window is a dashboard with five summary cards:

- 41 Available scripts** (with a bar chart icon)
- 7 Current jobs** (with a checkmark icon)
- 2 Current templates** (with a person icon)
- Jan 27, 2022** Last repository update
- 0 Open user deletion requests** (with a person icon and a red 'X')

At the bottom of the interface, there is a footer with the text: 'Copyright © 2022 holistic-net. All rights reserved.' and 'Version 1.0.8062.28026'.



Das holistic powershell launch center nutzt das mächtige Framework der Microsoft PowerShell und dessen Leistungsfähigkeit in Kombination mit funktionalen Erweiterungen wie z.B.

- Delegation
- Bulk processing
- Script repository
- Script scheduling
- eMail-notification
- Process data reporting
- Credential management

hplc – Vorteile & Nutzen

- Anwendbarkeit auf die wichtigsten aktuellen und zukünftigen Zielsystemplattformen
- PowerShell- & Zielsystemkompetenzen sind nicht Voraussetzung
- Vermeidung von Fehleinstellungen mit allen nachteiligen Folgen
- Schaffung einer Ressourcenredundanz durch Delegation
- Effizienzsteigerung
- **Senkung der Total Cost of Ownership**

Wurde Ihr Interesse geweckt?

Dann kontaktieren Sie: Günter Mohren, +49 511 760728-46, guenter.mohren@holistic-net.de