

Image Management (DISM, Sysprep, SCCM, Autopilot etc.)

Prio mit Emojis:

↓ niedrig

ohne Emoji = normal

↑ wichtiger

- [DISM Editionswechsel](#)
- [↓ 3rd-party Programme für Betriebssystemimages](#)
- [Platten für SCCM vorbereiten](#)
- [WQL Abfragen](#)

DISM Editionswechsel

Hier:

<https://notizlo.ch/eine-windows-10-enterprise-iso-herunterladen/>

Die Überschrift hat sich die letzten Jahre als gewinnbringender herausgestellt als direkt mit DISM zu machen. Hatte halt damals noch niemand als "Enterprise ISO download" verkauft.

In short:

bei laufenden Systemen

```
dism /online /Set-Edition:Enterprise /productkey:NPPR9-FWDCX-D2C8J-H872K-2YT43
```

bei ISO-Files

- Media Creation Tool Download

```
Dism /Mount-Image /ImageFile:C:\temp\install.wim /MountDir:C:\wimfile  
Dism /Image:C:\wimfile /Set-Edition:Enterprise  
Dism /Unmount-Image /MountDir:C:\wimfile /Commit
```

Ist ganz praktisch, weil man dann auch gleich die künftige C-Platte manipulieren kann. Für direkte Programminjektionen würde ich aber Sysprep empfehlen. [Kreuzverweis auf c't 04/2014 hier](#), die haben mir damals die ganze Thematik schon näher gebracht.

Lose Weitersammlung zusammengehöriger Dinge:

Windows 10 KMS Keys: [https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/jj612867\(v%3Dws.11\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/jj612867(v%3Dws.11))

Windows 11 KMS Keys: <https://docs.microsoft.com/en-us/windows-server/get-started/kms-client-activation-keys>

Aufgebohrtes MCT: <https://github.com/AveYo/MediaCreationTool.bat>

↓ 3rd-party Programme für Betriebssystemimages

Kurze Liste der Dinge die ich mit in Windows-Images integrieren würde, weil sie doch recht zentrale Aufgaben erfüllen.

Damit meine ich keine fetten Biester oder irgendwelche Anwender-Software, sondern Admin-Extrafunktionen, überwiegend Kommandozeile:

- [ffmpeg](#)
- [yt-dlp](#)
- [usbdm](#)
- [chocolatey](#) (eigentlich sollte man ja winget verwenden)
- [git](#)
- [gpg4win](#)
- [sudo](#) ☐

Platten für SCCM vorbereiten

Bevor ihr so etwas macht überlegt euch ob ihr die Computer nicht mit dem bestehenden Image vom Hersteller ohne große Neuinstallationen an SCCM und Intune ranflanschen könnt. Spart ganz gehörig Zeit und damit Lohnkosten und geht auch manchen Problemen aus dem Weg, weil man sich dann nicht groß um Treiber Gedanken machen muss. Booten, Intune-Join, Autopilot oder anderes Softwarerollout ran und fertig. Windows bietet mit der OOBE (Erststart neuer Rechner) für Unternehmens-Konten genau das...

Der Endpoint Configuration Manager braucht eine gewisse Plattenstruktur aus verschiedenen Partitionen. Existiert diese nicht schlägt die festgelegte Task Sequence fehl.

Dies ist die Plattenstruktur die der MECM erwartet, als ausführbares Powershell script.

Ihr müsst Diskpart ohne /s ausführen wenn ihr das händisch macht, EOF wäre das Endzeichen fürs Diskpart-Script.

F8 ist der Hotkey für die Kommandozeile

Deutsches Keyboard Layout:

```
wpeutil SetKeyBoardLayout 0407:00000407
```

Doppelpunkt ist das große Ö auf der deutschen Tastatur wenn die Software ein englisches Layout interpretiert.

Und fünf folgende Nullen...

```
# PowerShell script
diskpart /s <<EOF
Select Disk 0
Clean
Convert gpt
Create partition efi size=200
Assign letter=s
Format quick fs=FAT32
Create partition msr size=128
Create Partition Primary
Select Partition 3
Active
```

```
Assign letter=c  
Format quick fs=NTFS  
Exit  
EOF
```

Für Legacy lässt man die EFI-Partition weg

```
diskpart /s <<EOF  
Select Disk 0  
Clean  
Create Partition Primary  
Select Partition 1  
Active  
Assign letter=c  
Format quick fs=NTFS  
Exit  
EOF
```

gestohlen von [SCCM OSD DiskPart Commands Troubleshooting - Prajwal.org](https://prajwal.org/sccm-osd-diskpart-commands-troubleshooting/)

Dies ist meiner Erfahrung nach für ca. 70 % aller SCCM Rollout Fehler die Lösung, ist dies nicht erfolgreich überprüft die Treiber die vor Installation in das Image injected werden müssen. Dann fehlen dort Treiber

(oder ihr habt die falschen Laptops, die solch Drittanbieter-Installation-Injects noch brauchen)

WQL Abfragen

Testen

Im Endpoint Configuration Manager müssen für die Tasksequenzen in den Optionen keine Shell Befehle eingetragen werden sondern WQL queries.

```
wmic computersystem get model
```

wird also zu SQL ähnlichem:

```
SELECT * FROM Win32_Computersystem WHERE model LIKE "..."
```

mit wbemtest.exe kann dies auf den Geräten selbst geprüft werden.

Dazu muss die WMI Klasse gefunden werden, die ab und an nicht so heißt wie dessen Abfrage in der Shell.

Tasksequenzen für Treiber

1. Treiberimport
2. Treiberpaket definieren
3. Tasksequenz für dieses Treiberpaket an WQL Query binden

Lenovo X1 Carbon

Eigentlich mag ich die Serie, halt auch wegen ihrer Treiber. Es ist äußerst selten dass Lenovo Treiber einen BSOD produzieren.

Der Mensch der aber entschieden hat in mehreren Generationen nacheinander die Model numbers gleich zu belassen gehört gefeuert. Das ist sonst bei Lenovo nirgends der Fall, ist mir zumindest nicht untergekommen. Gen9, Gen10 und Gen11 in gleicher Ausstattungsstufe aber mit gleichen model numbers zu benennen... grr....

Tasksequenzoptionen

```
SELECT * FROM Win32_ComputerSystemProduct WHERE Vendor LIKE "%LENOVO%"
```

```
SELECT * FROM Win32_ComputerSystemProduct WHERE Version LIKE "%ThinkPad X1 Carbon Gen 11%"
```

Was genau zurückgegeben wird scheint wohl egal zu sein. Hier sind es ja ganze Tabellen.
Hauptsache die Anfrage ist erfolgreich und das LIKE kann sich auflösen.