

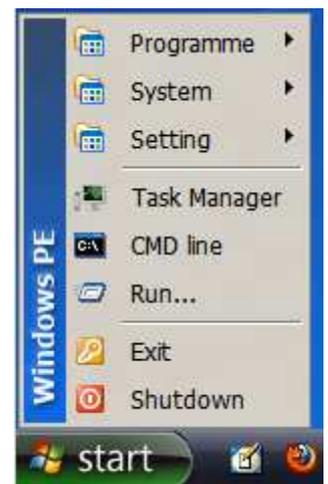
Erstellen einer Windows PE 3.0 CD ohne WinBuilder

Windows PE ist seit langem für viele IT-Fachleute ein unverzichtbares Instrument zur Wartung und Reparatur von PCs. Die aktuelle Version 3.0 basiert auf Windows 7. Für die Erstellung einer Windows-PE-CD mit WinBuilder benötigt man das Windows AIK und eine Windows-7-DVD. Entweder eine Retailversion oder eine Testversion. Für Anwender, denen keine Windows-7-DVD zur Verfügung steht oder für technisch Interessierte gibt es hier eine Anleitung zur Erstellung einer Windows-PE-CD ohne WinBuilder. Das Windows AIK wird aber benötigt. Diese Anleitung erklärt nicht die Zusammenstellung einer kompletten Rescue-CD sondern erläutert die grundsätzliche Vorgehensweise zur Erstellung einer Windows-PE-CD. Als Anwendungen kommen hauptsächlich Programme von PortableApps.com zum Einsatz. Die Shell ist der BSExplorer 2.0.0.1.

Vorbereitungen

- Am besten ist es, wenn man zuerst die Shell einrichtet. Dazu erstellt man einige temporäre Ordner z. B.: `C:\SysDrive`. In diesem Ordner werden zwei Unterverzeichnisse erstellt: `Program Files` und `Windows`. Im `Windows`-Ordner erstellt man nun noch den Ordner `System32`. Zuletzt wird noch ein Ordner `C:\CDDrive` mit einem Unterverzeichnis `Programs` erstellt.
- Jetzt werden die Programme, die man der PE-Umgebung hinzufügen möchte, installiert. Programme von PortableApps.com werden durch das Installieren nur in ein Verzeichnis entpackt. Alle Programme, die später im Bootabbild¹ enthalten sein sollen, werden nach `C:\SysDrive\Program Files` installiert. FirefoxPortable z. B. nach: `C:\SysDrive\Program Files\FirefoxPortable`. Beachten sollte man aber, dass das Bootabbild in einer RAM-Disk ausgeführt wird. Auf PCs mit wenig Arbeitsspeicher könnte es Probleme geben, wenn zu viele Programme im Bootabbild enthalten sind. Alle Programme, die nicht im Bootabbild enthalten sein sollen, werden nach `C:\CDDrive\Programs` installiert.

- Nun kann der BSExplorer eingerichtet werden. Der BSExplorer wird in folgendes Verzeichnis kopiert: `C:\SysDrive\Program Files\BSExplorer`. Die drei Konfigurationsdateien `bs_qlaunch.ini`, `bs_shell.ini` und `bs_start.ini` werden nun nach Bedarf angepasst. Im Downloadarchiv `Konfigurationsdateien.zip` habe ich diese Dateien exemplarisch für Total Commander und FirefoxPortable, als im Bootabbild enthaltene Programme und für XnViewPortable als ein nicht im Bootabbild enthaltene Programm erstellt. In die Konfigurationsdateien werden zunächst die lokalen Pfade zu den Programmen eingetragen. So zum Beispiel `ExeName=C:\SysDrive\Program Files\FirefoxPortable\FirefoxPortable.exe` Sind alle Menüeinträge vollständig gemacht, kann der BSExplorer gestartet werden. Jetzt können alle Menüeinträge überprüft werden: Werden alle Icons richtig dargestellt, funktionieren die Links, starten die Programme ordnungsgemäß usw. Wenn Fehler auftreten, den BSExplorer über Start → Exit beenden und die entsprechende Konfigurationsdatei korrigieren. Im Offline²-Abbild können Startmenüeinträge nur für Programme erzeugt werden, die im Bootabbild enthalten sind. Alle anderen Startmenüeinträge werden durch das Programm `shortcuts.exe` zur Laufzeit erzeugt (siehe nächsten Abschnitt).



Wenn das Startmenü einwandfrei funktioniert, müssen die Konfigurationsdateien an die Laufzeitumgebung angepasst werden. Dazu müssen alle Einträge, die Pfadangaben zu Testordnern haben geändert werden:

`C:\SysDrive` wird in `%SystemDrive%` umgeändert.

- Im nächsten Schritt werden die Konfigurationsdateien für Windows PE erzeugt und in die entsprechenden Ordner

```
Win7PEcdg - Notepad2
File Edit View Settings Help
[Main]
2:screen=1024x768
3:upoint=normal
4:mountprogram=yes
5:defaultshell=BS_Explorer
6
7
8 [shortcuts]
9 Grafik\%SystemDrive%\Programme\XnViewPortable\XnViewPortable.exe||
10 Desktop\Total Commander\%SystemDrive%\Program Files\Total Commander\TDCMD_EXE11
11 Desktop\FirefoxPortable\%SystemDrive%\Program Files\FirefoxPortable\FirefoxPortable.exe||
12 Desktop\%SystemDrive%\Programme\XnViewPortable\XnViewPortable.exe||
13
14 [autorun]
15
16 [registry]
17
18 [association]
19
20 [shell]
21 BS_Explorer\%SystemDrive%\Program Files\BS_Explorer\Explorer.exe
22
23 [postconfig]
24
25 [preconfig]
26
```

kopiert. Das ist zum einen die „autostart“-Datei für Windows PE: *winpeshl.ini* und zum anderen die Konfigurationsdatei *Win7PE.cfg*. Beide Dateien werden in den Ordner *C:\Sysdrive\Windows\System32* kopiert. Die Datei *winpeshl.ini* enthält als erstes einen Aufruf von *wpeinit.exe*. Dieses Programm initialisiert Windows PE und Plug-and-Play-Hardware. Wpeinit sucht nach einer Datei *unattend.xml* und arbeitet die darin enthaltenen Anweisungen ab. Diese Datei muss sich ebenso im Ordner *System32* befinden. Als Beispiel habe ich die TCP/IP-Konfiguration in diese Datei geschrieben. Sollen statische IP-Adressen verwendet werden, muss auch der DNS-Client konfiguriert werden. Der DNS-Client lässt sich bei Windows PE nicht über die Datei *unattend.xml* konfigurieren. Daher erfolgt an zweiter Stelle in der Datei *winpeshl.ini* ein Aufruf eines DNS-Konfigurationsscripts. Man könnte in diesem Script auch TCP/IP konfigurieren. Aber *wpeinit.exe* arbeitet schneller, wenn es eine *unattend*-Datei gibt. Deshalb wurde hier diese Trennung vorgenommen. Wird TCP/IP mit DHCP konfiguriert, entfällt der Aufruf von *dns-addr.cmd*. Die Desktopicons und die restlichen Startmenüeinträge werden durch das Programm *shortcuts.exe* erzeugt. Dieses muss sich, zusammen mit der Bibliothek *msobovm60.dll*, ebenfalls im Ordner *System32* befinden. In der Konfigurationsdatei *Win7PE.cfg* sind die zu erzeugenden Shortcuts aufgelistet. Als letztes wird die Shell, hier BSExplorer, gestartet.

ISO-Image erzeugen

- Damit sind die Vorbereitungen für die Erzeugung eines Windows PE-Abbilds abgeschlossen. Jetzt muss das Windows Automated Installation Kit (WAIK) installiert werden, falls es noch nicht erfolgt ist.
- Die Arbeiten zur Erstellung eines Windows PE-Abbilds werden hauptsächlich an der Kommandozeile ausgeführt und erfordern Administratorrechte. Dazu auf Start → Alle Programme → Zubehör → Eingabeaufforderung mit der rechten Maustaste klicken und Als Administrator ausführen anklicken. Nun zum Ordner *C:\Windows AIK\Tools\PETools* wechseln und mit dem Befehl

```
copy c:\winpe_x86 (für eine x86-Umgebung)
```

die Arbeitsumgebung erstellen. Es wird folgende Verzeichnisstruktur angelegt:

```
C:\winpe_x86
C:\winpe_x86\ISO
C:\winpe_x86\mount
```

- Mit dem Tool DISM wird jetzt das Wim-Archiv bereitgestellt: An der Eingabeaufforderung zum Ordner *C:\winpe_x86* wechseln und das Wim-Archiv mounten:

```
dism /mount-wim /wimfile:winpe.wim /index:1 /mountdir:mount
```

- Jetzt werden die vorbereiteten Dateien in das Archiv kopiert:

```
xcopy c:\SysDrive\*. * c:\winpe_x86\mount\ /e /h
xcopy c:\CDDrive\*. * c:\winpe_x86\ISO\ /e /h
```

Das Kopieren kann auch mit dem Windows-Explorer oder einem anderen Dateimanager durchgeführt werden. Für den Kopiervorgang werden Administratorrechte benötigt. Anschließend wird das Wim-Archiv entladen:

```
dism /unmount-wim /mountdir:mount /commit
```

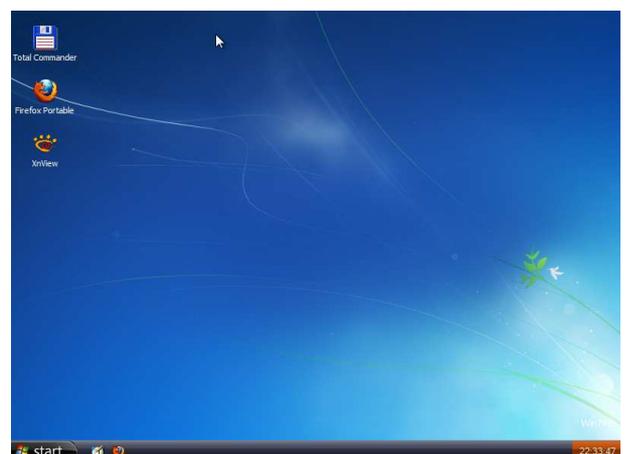
Das angepasste Wim-Archiv wird nun noch in den Ordner *c:\winpe_x86\ISO\sources* kopiert und in *boot.wim* umbenannt.

- Im letzten Schritt wird das ISO-Image erzeugt:

```
oscdimg -n -bc:\winpe_x86\etfsboot.com
c:\winpe_x86\ISO c:\winpe_x86\winpe_x86.iso
```

Das Tool *oscdimg.exe* ist Bestandteil des WAIK.

- Bevor das ISO-Image auf einen Datenträger gebrannt wird, sollte es getestet werden. Dazu eignet sich ein



Virtualisierungsprogramm wie zum Beispiel Virtual Box. Es wird eine neue virtuelle Maschine erzeugt und bei Massenspeicher ein neuer Controller eingefügt. An den Controller wird ein CD-Laufwerk angeschlossen. In dieses CD-Laufwerk kann das soeben erstellte ISO-Image geladen werden. Für eine Internetverbindung ist es notwendig, dass ein passender Treiber für die Netzwerkkarte geladen wird. In VBox kann man den emulierten Netzwerkadapter einstellen. Windows PE hat eine Reihe von Netzwerktreibern an Bord. Unter anderem auch für Intel PRO/1000 MT Desktop, der auch von VBox emuliert werden kann. Für die korrekte Netzwerkanbindung von VBox an das Hostsystem muss ich auf das Handbuch von VBox verweisen, weil eine derartige Erläuterung den Rahmen dieser Abhandlung sprengen würde.

Hinzufügen von Treibern

- Windows PE enthält nur eine begrenzte Anzahl an Treibern. Einen Treiber hinzufügen kann man nach dem Mounten des Wim-Archivs. Der Treiber kann sich in einem beliebigen Verzeichnis auf der Festplatte befinden. Mit dem Tool DISM wird der Treiber hinzugefügt. DISM erwartet als Parameter den Pfad zum gemounteten Image und den Pfad zur inf-Datei des Treibers.

```
dism /image:mount /add-driver /driver:c:\amd\netamd.inf
```

Exemplarisch wurde hier der Treiber einer Netzwerkkarte zu Windows PE hinzugefügt. Der Treiber für AMD PCnet-PCI II ist standardmäßig nicht in Windows PE enthalten. VBox kann auch diese Netzwerkkarte emulieren, so dass leicht überprüft werden kann, ob das Einfügen des zusätzlichen Treibers erfolgreich war.

Da man aber unmöglich alle Treiber für alle möglichen PCs in das Bootabbild integrieren kann gibt es das Tool DRVLOAD. Damit kann man Treiber zur Laufzeit nachladen. DRVLOAD ist ein Kommandozeilen-tool und Bestandteil von Windows PE. Das Tool erwartet als Parameter nur den Pfad zur inf-Datei des Gerätetreibers, z. B.:

```
drvload c:\treiber.inf
```

Geräte manager

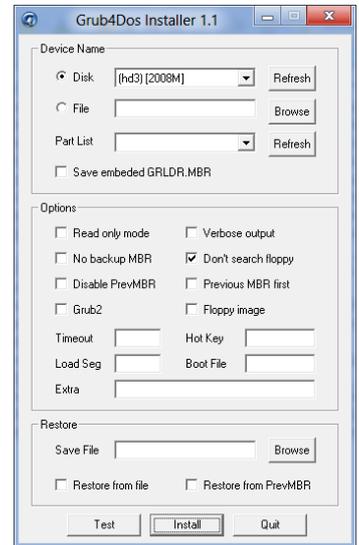
- Ein Manko von Windows PE 3.0 ist, dass es keinen Geräte manager gibt. Hier bietet sich die Freeware-Alternative DevManView an. DevManView wird bei der Installation nur in ein Verzeichnis entpackt. Dieses Verzeichnis kann man dann in der Vorbereitungsphase nach *C:\CDDrive\Programs* kopieren oder vor der Erstellung des ISO-Images direkt nach *C:\winpe_x86\ISO\Programs*. DevManView erlaubt zwar keine Installation von Gerätetreibern, aber man kann Geräte deaktivieren und aktivieren. Auch eine Deinstallation von Gerätetreibern ist möglich.

USB-Stick

- Soll Windows PE von einem USB-Stick gestartet werden, gibt es zwei Möglichkeiten. Entweder man kopiert den gesamten Inhalt des Ordners *C:\winpe_x86\ISO* auf den USB-Stick oder man kopiert das ISO-Image auf den USB-Stick. Der USB-Stick muss vorher bootfähig gemacht werden. Das geschieht mit dem Programm Grub4Dos.
- USB-Stick, falls erforderlich, partitionieren und mit FAT32 formatieren. Das kann mit der Datenträgerverwaltung von Windows erfolgen (mit Administratorrechten ausführen). Beim Partitionieren auf Folgendes achten: Es muss eine primäre Partition angelegt werden und diese muss aktiviert werden. Ein schon benutzter USB-Stick ist bereits partitioniert. In der Datenträgerverwaltung wird auch das Dateisystem angezeigt, sollte es nicht FAT 32 sein, muss der Stick neu formatiert werden. Grub4Dos kann nicht auf NTFS-Partitionen zugreifen.



- USB-Stick an PC anstecken und Grub4Dos Installer (*grubinst_gui.exe*) als Administrator starten. Ganz oben bei Disk den USB-Stick auswählen, sollte der letzte Eintrag mit (hd...) sein, zur Sicherheit anhand der Speichergröße identifizieren. Die Checkbox Don't search floppy aktivieren und auf Install klicken. Der Masterbootrecord wird jetzt auf den Stick geschrieben. Mit Qjüt den Installer verlassen. Nun die Datei *grldr* auf den Stick kopieren. Als Nächstes wird die Datei *menu.lst* erzeugt und auf den Stick übertragen. Das ist eine einfache Textdatei und kann mit jedem Editor, z. B. Notepad, erzeugt werden. Der Inhalt der Datei *menu.lst* ist davon abhängig, wie vom Stick gebootet werden soll:



- 1) gesamter Ordnerinhalt wurde auf den Stick kopiert:

```
color black/cyan yellow/cyan
title Windows PE
find --set-root /bootmgr
chainloader /bootmgr
title Neustart
reboot
title Herunterfahren
halt
```

- 2) ISO-Boot

```
color black/cyan yellow/cyan
title Windows PE ISO
find --set-root /winpe_x86.iso
map (hd0,0)/winpe_x86.iso (hd32)
map --hook
chainloader (hd32)
title Neustart
reboot
title Herunterfahren
halt
```

Man kann auch beide Bootvarianten in eine Datei schreiben. Dazu werden die Zeilen `title...` bis `chainloader...` hinter dieselben Zeilen in der anderen Datei geschrieben.

WICHTIG: Bei der ISO-Boot-Variante ist es nicht möglich, Programme von `%CDDrive%` zu starten, da nach dem Bootvorgang nicht mehr auf das ISO-Image zugegriffen werden kann.

Um dennoch auf die Programme zugreifen zu können, benötigt man einen Treiber für ein virtuelles CD-Laufwerk. Hier bietet sich `ImDisk` an. Dieses Tool ist zwar keine echte Portable-Anwendung benötigt nach der Installation jedoch keinen Neustart des Rechners. `ImDisk` kann über die Systemsteuerung oder über die Kommandozeile gesteuert werden. Windows PE besitzt keine Systemsteuerung also bleibt nur die Kommandozeile. Um ein CD-Image zu mounten benötigt `ImDisk` nur zwei Parameter: 1. den Pfad und den Namen des Images und 2. den Mountpoint.

```
imdisk -a -f k:\winpe_x86.iso -m l:
```

In diesem Beispiel befindet sich das ISO-Image auf Laufwerk K: und wird als Laufwerk L: gemountet. Auf die Programme in `%CDDrive%` kann jetzt über `l:\programs` zugegriffen werden.



- 1) Bootabbild
Das Bootabbild ist eine Windows-Image-Datei (WIM), die Windows PE und die Programme enthält, die zur Laufzeit in einer RAM-Disk ausgeführt werden.
- 2) Offline
Als Offline-Abbild wird das gemountete Bootabbild bezeichnet.