

DOS-CRASHKURS

In Zeiten, in denen Betriebssysteme wie Windows mit ihrer grafischen Benutzeroberfläche den Markt dominieren, vergisst man allzu leicht den Urvater aller MS-Betriebssysteme: DOS.

Auch wenn man mittlerweile recht wenig mit diesem Betriebssystem in Kontakt kommt, ist es doch sehr sinnvoll, sowohl für Neueinsteiger als auch für erfahrene User, das Grundwissen über DOS aufzufrischen. Die Kenntnisse über DOS könnten insbesondere bei einem Crash nützlich sein, wenn Windows z.B. nicht mehr startet oder man ein Uralt-DOS-PC-Spiel noch mal spielen will. Dieser Crashkurs soll allen Usern noch einmal die Grundzüge und Bedienung von DOS vor Augen führen. Er soll vielmehr als solide Startbasis dienen, um das alte DOS noch einmal neu zu erfahren.

Viel Glück und Erfolg beim Ausprobieren ;-)

ÜBERSICHT:

Sobald man in DOS ist, erwartet einen die DOS-Eingabeaufforderung.

z.B. c:\

Der erste Buchstabe gibt dabei den Namen des Laufwerks an. In diesem Bsp. heißt unser Laufwerk c. Diskettenlaufwerke haben in aller Regel den Buchstaben a und CD-Rom Laufwerke den Buchstaben d (kann sich aber auch verschieben).

Es herrscht ferner das

8.3 Format - Dateinamen können nur 8 Buchstaben lang sein und Dateierweiterungen nur 3 Buchstaben. Verzeichnisse nur 8 Buchstaben.

Jedes weitere „\“ hinter dem Laufwerksbuchstaben gibt dabei das jeweils aktive Verzeichnis an, in dem man sich momentan befindet. Momentan befinden wir uns also im Hauptverzeichnis von C.

Weitere Beispiele.:

c:\temp Wir befinden uns im „temp“-Verzeichnis auf dem Laufwerk c.

c:\temp\hard Wir befinden uns im Verzeichnis Hard, welches das Unterverzeichnis von temp ist. Beide befinden sich auf dem Laufwerk c.

d:\temp\hard Das Gleiche wie im obigen Beispiel, nur dass wir uns nun auf dem Laufwerk d befinden .

DOS-BEFEHLE

Jeder Befehl in DOS wird mit „Enter“ bestätigt.

Ein Befehl kann um zahlreiche Parameter ergänzt werden, die dazu helfen einen Befehl mit einer alternativen/zusätzlichen Funktion auszuführen.

...

Ein Parameter wird in der Regel mit einem „/“ hinter einem Befehl aufgeführt. In seltenen Fällen muss man aber auch ein „-“ oder „|“ drücken.

Wichtige Parameter:

- /? Ruft die Hilfe für den jeweiligen Befehl auf (z.B. dir/? oder format/?)
Dadurch erfährt man in der Regel noch weitere Parameter für den jeweiligen Befehl.
- /p Seitenweise Darstellung der Ergebnisse, falls der Monitor nicht genügend Platz bietet und die Ergebnisse nur so nach unten rauschen (z.B. dir/p oder mem/p)
- | more manchmal funktioniert der /p Befehl bei einigen Kommandos nicht. In diesem Fall bedient man sich dieses Parameters, um die Bildschirmanzeige seitenweise anzuzeigen. (z.B. type win.ini | more)
- /y Bestätigungsfunktion. Anstatt bei jedem Ja/Nein-Dialog (Möchten Sie diese Platte formatieren? ...) manuell sein Ja einzugeben, kann man mit diesem Parameter automatisch jeden Befehl bestätigen.
- /s Alle weiteren Unterverzeichnisse werden in den Befehl mit eingeschlossen. (z.B. dir /s, del readme.txt /s etc.)

Hier die wichtigsten Befehle, die jeder kennen sollte

Navigation:

dir (directory)

Dieser Befehl gibt einem das Inhaltsverzeichnis über alle Dateien und Verzeichnisse im jeweils aktiven Verzeichnis zu erkennen. Verzeichnisse werden mit <DIR> vermerkt. Dateien hingegen werden mit ihren Endungen angezeigt. Der „dir“-Befehl kann auch für die **Dateisuche** benutzt werden.

/p Mit dem Parameter /p (also dir/p) kann man die Verzeichnisanzeige seitenweise anzeigen.

/w Mit diesem Parameter kann man „dir“ zwingen die Anzeige QUER zu legen.

Beispiele:

Anzeige des momentanen Inhaltsverzeichnisses
dir/P

Anzeige aller Dateien mit den Endungen TXT (mit Hilfe von Wildcards;
Erklärung von Wildcards weiter unten)
dir *.txt /p

Anzeige aller Dateien mit dem Namen wintotal

...

dir wintotal.* /P

Dateisuche

Um die Datei readme.txt im momentanen Verzeichnis zu durchsuchen, gibt man folgenden Befehl ein.
dir readme.txt

Um die **Suche auf alle weiteren Unterverzeichnisse** zu erweitern, gibt man den Parameter /s mit ein.
dir readme.txt /s

Der undokumentierte Befehl dir, (auf das Komma achten) zeigt einem auch versteckte Systemdateien an.
dir,

L: Laufwerkswechsel (L = Name des Laufwerks, auf das gewechselt werden soll). z.B. a: oder d:

cd v cd (change directory)
Verzeichniswechsel (v = Name des Verzeichnisses)
z.B. „cd windows“ bringt einen ins Windows-Verzeichnis, wenn man diesen Befehl im Hauptverzeichnis der Festplatte ausführt.
In welches Verzeichnis man wechseln kann, darüber kann der „dir“-Befehl Hinweise geben. Verzeichnisse werden beim Ausführen des „dir“-Befehls mit <DIR> vermerkt
(z.B. cd c:\windows
cd d:\windows\backup
cd c:\temp\level\one)

cd.. Wechsel zum nächsthöheren Verzeichnis
Z.B. man befindet sich im Verzeichnis c:\windows\temp:

Durch den Befehl „cd..“ gelangt man zum Verzeichnis c:\windows\
Durch nochmaliges Ausführen dieses Befehls gelangt man dann ins Hauptverzeichnis c.

**cd ** Wechsel zum Hauptverzeichnis des jeweiligen Laufwerks
Nützlich, wenn man sich in einem tiefen Verzeichnisbaum befindet und man schnell ins Hauptverzeichnis springen möchte.
Z.B. man befindet sich im c:\windows\temp-Verzeichnis.
Durch diesen Befehl spart man sich 2 „cd..“-Befehle und kommt direkt zum Hauptverzeichnis.

VERZEICHNISBEFEHLE

md v md (make directory)
Verzeichnis erstellen (v = Name des zu erstellenden Verzeichnisses, Achtung!!
Verzeichnisname kann nur 8 Buchstaben haben wegen dem 8.3 Format; z.B. „md temp“ oder „md wintotal“)

rd v rd (remove directory)
Verzeichnis entfernen (v = Name des zu entfernenden Verzeichnisses). ...

Nur LEERE Verzeichnisse können entfernt werden !!
Vor dem Entfernen eines Verzeichnisses müssen also alle Daten im Verzeichnis gelöscht werden.
(Will man alle Verzeichnisse mit Unterverzeichnissen entfernen, bedient man sich des mächtigen „deltree“-Befehls - Erklärung weiter unten).
z.B.
Entfernen des Verzeichnisses c:\temp
rd c:\temp
Entfernen des Verzeichnisses c:\wintotal
rd c:\wintotal

cd v cd (change directory)
Verzeichniswechsel (v = Name in das zu wechselnde Verzeichnis)
Siehe obige Beschreibung.

deltree v Löscht ein Verzeichnis mit ALLEN Unterverzeichnissen.
Sehr mächtiger Befehl.
z.B. deltree „c:\windows“ , um das komplette Windows-Verzeichnis samt Unterverzeichnissen unwiderrufflich zu löschen.

xcopy Dieser Befehl kann ganze Verzeichnisse oder Dateiformate hin- und herkopieren.
xcopy a b /s a = Quellverzeichnis z.B. c:\temp
b = Zielverzeichnis z.B. d:\temp

Würde das gesamte Verzeichnis a mit samt allen Unterverzeichnissen in das Verzeichnis b kopieren.
/s Der Parameter s bezieht dabei ALLE Unterverzeichnisse der Quelle mit in den Kopiervorgang ein
/e erzeugt ein Unterverzeichnis im Ziel selbst dann, wenn dieses leer ist.
Nur in Verbindung mit dem s-Parameter zu nutzen

Beispiele

Wir kopieren ein Verzeichnis a = c:\temp in das Verzeichnis d:\temp mit allen Unterverzeichnissen
xcopy c:\temp d:\temp /s/e

Wir kopieren alle Dateien des Formats TXT im Laufwerk c nach d:\temp.
Wildcards werden uns in diesem Fall helfen
xcopy c:*.txt d:\temp /s/e

DATEIBEFEHLE

Erklärung der Abkürzung „d“

d = kann für einen Dateinamen (wintotal.txt) stehen, aber auch für sog. Wildcards bzw. Platzhalter.
Will man z.B. alle Dateien des Formats TXT kopieren, gibt man einfach *.txt ein, anstatt jede einzelne Datei aufzuzählen. Dieses „*“ wird als Wildcard bezeichnet.

Der normale Windows-User braucht für diese Aktion nur im Windows Explorer auf die Spalten „Typ“ bzw. „Dateinamen“ zu klicken.

...

Hier weitere Beispiele für Wildcards:

*.txt alle Dateien mit der Dateiendung TXT sind ausgewählt
wintotal.* alle Dateien mit den Dateinamen wintotal sind ausgewählt. Es werden Dateien mit allen Dateinennungen angezeigt.

Für eure Fantasie noch einige andere Beispiele:

Win*ws.* ; *win.avi

del d del (delete)
Löschbefehl für eine oder mehrere Dateien

z.B.
del *.* löscht alle Dateien im aktuellen Verzeichnis
del *.txt löscht alle Dateien mit den Endungen TXT im akt.
Verzeichnis
del readme.txt löscht die Datei readme.txt im aktuellen Verzeichnis

copy d1 d2 Kopiert eine Datei in ein anderes Verzeichnis oder in dasselbe Verzeichnis
aber unter einem anderen Namen.
D1 = Quelldatei, Wildcards
D2 = Zieldatei mit oder ohne Wildcards, Ziellaufwerk.
Hier wieder einige Beispiele für eure Fantasie:
a) copy wintotal.txt a:\ (kopiert die wintotal.txt auf das Laufwerk a)
b) copy wintotal.* c:\temp (kopiert alle Dateien mit dem Namen wintotal
in das Verzeichnis c:\temp)
c) copy c:\temp*.txt d:\windows (kopiert alle Dateien der Endung TXT im
c:\temp-Verzeichnis in das Windows-Verzeichnis auf dem Laufwerk d

Wem der copy-Befehl nicht reicht, der möge sich den „xcopy“-Befehl (weiter
oben) näher anschauen ;-)

move d1 d2 Der Windows-User würde zu diesem Befehl „Ausschneiden sagen“. Prinzipiell
ist dieser Befehl mit dem copy-Befehl identisch. Nur dass die Quelldatei
VERSCHOBEN und nicht kopiert wird.

Beispiele:
move *.* a:\ alle Dateien werden zum Laufwerk a verschoben
Move c:\readme.txt c:\temp
Die Datei readme.txt wird vom Hauptverzeichnis c ins Unterverzeichnis
c:\temp verschoben.

edit d DOS-Editor (d = Dateiname (8.3 Format))
Der DOS-Editor ist im Grunde der Vorgänger zu Notepad.
Einmal aufgerufen, wird eine grafische Benutzeroberfläche für den Editor
gestartet, mit der man fortan alle möglichen Dateien (TXT, bat, sys etc.)
bearbeiten kann.

Beispiele:
edit c:\config.sys
edit a:\autoexec.bat
edit c:\temp\readme.txt

ren d1 d2 (ren = rename; d1 = Quelldatei; d2 = Zieldatei)
Umbenennen-Befehl

...

Um die Datei readme.txt in wintotal.txt umzubenennen:

```
ren readme.txt wintotal.txt
```

Man kann auch hier Wildcards einsetzen.

Um z.B. alle TXT-Dateien im Verzeichnis c:\temp in INI-Dateien umzuwandeln:

```
ren c:\temp\*.txt c:\temp\*.ini
```

attrib [Parameter] d Datei-Eigenschaften ändern

Jede Datei hat wie in Windows verschiedene Dateieigenschaften.

Es gibt insgesamt 4 verschiedene Dateieigenschaften.

- a) **read-only** = Datei ist schreibgeschützt
- b) **hidden** = Datei ist versteckt (nur mit dem Befehl dir, sichtbar)
- c) **archiv** = Datei verfügt über ein Archivflag
- d) **system** = Datei ist eine Systemdatei (elementar wichtige Datei wie command.com, d.h. sie kann nur nach Abfrage gelöscht, verschoben oder umbenannt werden).

Der jeweilige Parametername wurde dabei **fett** gekennzeichnet (r für schreibgeschützt, h für versteckt etc.).

Um jede dieser einzelnen Eigenschaften zu ändern, wird der attrib-Befehl angewandt.

Im Nachfolgenden werden die ganzen Parameter erklärt.

[Parameter]

Um eine Eigenschaft hinzuzufügen wird ein + verwendet.

Um eine Eigenschaft zu entfernen wird ein – verwendet.

±r	Schreibschutz ändern
±h	Verstecken-Eigenschaft ändern
±a	Archivflag ändern
±s	Systemflag ändern

Man muss dabei nur das jeweilige Attribut in der Kommandozeile angeben, das man ändern will.

Hier einige Beispiele, um uns das Ganze zu erleichtern:

Alle Eigenschaften für die Datei config.sys aktivieren:

Config.sys ist nun schreibgeschützt, versteckt, eine Systemdatei und hat ein Archivflag.

```
attrib +r +h +a +s config.sys
```

Nun sollen alle Eigenschaften bis auf die Systemeigenschaft wieder entfernt werden:

```
attrib -r -h -a +s config.sys
```

Config.sys soll schreibgeschützt, aber keine Systemdatei mehr sein:

```
attrib +r -s config.sys
```

Alle Files mit den Endungen TXT sollen versteckt werden:

```
attrib -h *.txt
```

...

Systembefehle

<i>label L</i>	Datenträgernamen ändern (L = Name des Laufwerks/Datenträgers) Im darauffolgenden Dialog gibt man dann einfach den neuen Namen ein.
<i>format L</i>	Formatiert das Laufwerk (L = Name des Laufwerks) <i>z.B.</i> <i>format a:</i> <i>format c:</i>
<i>diskcopy L: L:</i>	Mit diesem Befehl kann man Disketten kopieren. (L = Name des Laufwerks). Will man eine Diskette kopieren, gibt man <i>diskcopy a:</i> <i>a:</i> ein. Der erste Buchstabe gibt den Namen des Quelllaufwerks an, der zweite den Namen des Ziellaufwerks.
<i>time</i>	Zeigt die aktuelle Zeit an. Mit diesem Befehl kann man auch die Uhrzeit verstellen, indem man, nachdem man den Time-Befehl aufgerufen hat, im Time-Dialog die aktuelle Zeit eingibt.
<i>date</i>	Zeigt das aktuelle Datum an. Auch hier hat man die Option, wie beim Time-Befehl das Datum in einem Dialog zu verändern.
<i>mem</i>	Zeigt die momentane Speicherbenutzung an. Ich empfehle den Befehl in dieser Form anzugeben: <i>mem/c/p</i> Dadurch werden mehr Informationen angezeigt.
<i>ver</i>	zeigt die aktuelle DOS-Version an.
<i>fdisk</i>	Partitionierungsbefehl. Einmal ausgeführt, gelangen wir in eine grafische Benutzeroberfläche. Es arbeitet menüorientiert, der Windows-User wird sich freuen. Es gibt insgesamt 3 relevante Menüoptionen. Am besten man folgt den Bildschirmdialogen und liest sie sich gut durch. Ansonsten empfehle ich, beim Partitionieren ein anderes Programm als Fdisk zu verwenden. Auf der Wintotal-Seite gibt es einige sehr gute Alternativen zu fdisk (z.B. der Ranish Partition Manager). Menü 1 Partition erstellen Menü 2 Aktive Partition festlegen (Bootlaufwerk) Menü 3 Partition löschen
<i>fdisk /mbr</i>	Undokumentierter Befehl Dieser Befehl erlaubt es, den Master Boot Record Sektor der Festplatte neu zu erstellen. Sehr sehr nützlich, wenn ein Bootvirus (Parity Boot) im MBR festsetzt. Mit dieser Methode wird der Virus ratzefatz gekillt.
<i>sys L1 L2</i>	Systemdateien werden vom Quell-Laufwerk L1 zum Ziel-Laufwerk L2 kopiert. Systemdateien sind Dateien, die zum Starten von DOS ABSOLUT notwendig sind. Dazu gehören z.B. die <i>io.sys</i> , <i>msdos.sys</i> und die <i>command.com</i> . Bsp.: Um die Systemdateien vom Laufwerk <i>c</i> zum Laufwerk <i>a</i> zu kopieren, um so eine bootfähige Diskette zu bekommen, gibt man Folgendes ein: ...

sys c: a:

Der „Affengriff“ Neustart von DOS.
Durch Drücken der Tasten Strg+Alt+Entf wird DOS neu gestartet.
Dieses Kommando wird von vielen eingefleischten DOS-Usern als „Affengriff“ bezeichnet !!

Ausgabe von Dateien

type d Die Datei d wird auf dem Bildschirm angezeigt.
Um die Datei readme.txt anzuzeigen, geben wir Folgendes ein:
type readme.txt
Falls die Datei zu lang sein sollte, hilft der | more-Befehl weiter
type readme.txt | more

print d Die Datei d wird gedruckt.
Folgende Parameter sind wichtig.

/d:Gerät An welchem Gerät soll ausgedruckt werden.
Am COM1-Anschluss (/d:com1)
Am COM2-Anschluss (/d:com2)
oder z.B. am LPT1-Anschluss (/d:lpt1)

Um die Datei readme.txt am Drucker, welcher am LPT1-Anschluss hängt, zu drucken gibt man ein:
print readme.txt /d:lpt1

Interne Programme

Je nach DOS-Version kann es passieren, dass einige Programme nicht zur Verfügung stehen.

Die meisten dieser internen Programme verfügen über eine grafische Benutzeroberfläche, die mit einem Menüsystem ausgestattet ist. Sie sind folglich auf Benutzerfreundlichkeit optimiert. Am besten man liest sich von Menü zu Menü ;-).

So werden Windows9x-User mit ihrer DOS-Eingabeaufforderung z.B. nicht mehr in den Genuss von Memmaker kommen. Einfach mal ausprobieren, ob der ein oder andere Befehl geht oder nicht.

memmaker Speicheroptimierungstool, um den konventionellen Speicher unter DOS zu erhöhen. Absolut notwendig, um einige DOS-Spiele zum Laufen zu bringen. Memmaker verfügt über eine grafische Benutzeroberfläche, welche menüorientiert ist. Der Windows-User wird wohl keine Probleme haben, sich hier zurechtzufinden.

scandisk DOS-Pendant zum Windows Scandisk. Dieses Programm prüft die Oberfläche der Festplatte oder Diskette auf Fehler und versucht diese zu reparieren.

defrag Defragmentierungsprogramm, das ab DOS 6.2 mit zur Distribution gehört. Es unterscheidet sich nicht viel von der Windows-Version. Defrag ordnet die Dateien auf der Festplatte neu an, um das Leseverhalten der Festplatte zu optimieren. In günstigen Fällen wird der PC dadurch schneller!!

...

- dosshell Der grafische Dateimanager unter DOS.
Fast schon eine Legende. ;-)
- msd System-Diagnose-Programm (msd = MicroSoft Diagnostic)
DOS-Pendant zur Windows-Systeminformation.
- msav Das Antivirenprogramm von Microsoft.
Ich würde vom Benutzen dieses Programms abraten. Es gibt genügend
andere Freeware-DOS-Alternativen, welche aktueller und schneller sind.

Ich hoffe, der eine oder andere Tipp, den ihr noch nicht kanntet, hat euch geholfen besser mit DOS umzugehen. ;-)

Gruß PCDParker