

Free!

Lerne deinen PC kennen

Was ist denn DOS?

```
C:\>FANG AN  
Befehl oder  
Dateiname  
nicht gefunden
```

Acrobat Reader: Wie ...

F5/F6 öffnet/schließt die Ansicht **Lesezeichen**

Strg+F sucht

Im Menü Ansicht stellst du ein, wie die Datei gezeigt wird

STRG+0 = Ganze Seite **STRG+1** = Originalgrösse **STRG+2** = Fensterbreite

Im selben Menü kannst du folgendes einstellen:: **Einzelne Seite**, **Fortlaufend** oder **Fortlaufend - Doppelseiten** .. Probiere es aus, um die Unterschiede zu sehen.

Navigation

Pfeil Links/Rechts: eine Seite vor/zurück

Alt+ Pfeil Links/Rechts: Wie im Browser: Vorwärts/Zurück

Strg++ vergrößert und **Strg+-** verkleinert

Bestellung und Vertrieb für den Buchhandel

Bonner Pressevertrieb, Postfach 3920, D-49029 Osnabrück

Tel.: +49 (0)541 33145-20

Fax: +49 (0)541 33145-33

bestellung@knowware.de

www.knowware.de/bestellen

Autoren gesucht

Der KnowWare-Verlag sucht ständig neue Autoren. Hast du ein Thema, daß dir unter den Fingern brennt? - ein Thema, das du anderen Leuten leicht verständlich erklären kannst?

Schicke uns einfach ein paar Beispielseiten und ein vorläufiges Inhaltsverzeichnis an folgende Adresse:

lektorat@knowware.de

Wir werden uns deinen Vorschlag ansehen und dir so schnell wie möglich eine Antwort senden.

Mein PC und ich	4	Eine ganze Diskette kopieren.....	41
Sprache und Schreibweise	4	Formatierung.....	41
Lies mit den Fingern	4	Unformat.....	42
Was tut DOS eigentlich?.....	5	Systemdiskette	42
Hardware.....	5	Die Festplatte	43
Das Gehäuse	5	Hilfe	44
Der Bildschirm.....	7	Zeichen.....	45
Die Tastatur.....	7	Die Startdateien.....	46
Die Maus.....	8	Default	46
Der Drucker	8	Sei vorsichtig	46
Der Boot-Prozess	9	Panik?	47
Klar	9	BAT-Dateien.....	48
Shell	10	Wozu BAT-Dateien?	48
Befehle nicht vergessen	11	AUTOEXEC.BAT.....	49
Seite 46. Version.....	11	Einfache BAT-Dateien.....	49
Dateien	12	Installierung eines Programmes.....	50
Programme und Dokumente	12	Freeware und Shareware.....	50
Dateinamen	13	Versuch's einfach	51
Verzeichnisse	14	Automatischer Start eines Programms.....	52
Tree	16	Komprimierte Dateien.....	52
Die Platte wechseln.....	18	Selbstentpacken	52
Das Verzeichnis wechseln	18	Komprimierungsprogramme.....	54
Ein Verzeichnis erstellen	20	Ein PAK-Verzeichnis erstellen	54
Ein Verzeichnis löschen.....	20	PKZIP	55
DIR.....	21	Ein PKZIP-Archiv entpacken	55
Kurzübersicht.....	21	Ein ZIP-Archiv erstellen	56
Wild Cards	21	ARJ	56
Suche auf der gesamten Festplatte.....	22	Ein ARJ-Archiv entpacken	56
Versteckte Dateien	22	Ein ARJ-Archiv erstellen	57
Namen und Pfade von Dateien	23	LHA	57
Suchpfad	23	Ein LHA-Archiv entpacken	57
Dateitypen.....	24	Ein LHA-Archiv erstellen.....	57
Den Inhalt einer Datei ansehen	25	Virus	58
Eine Datei drucken.....	26	Leere Batterie.....	59
Das Schirmbild drucken.....	26		
Kontinuierlicher Druck	26		
Kopie einer Datei	27		
Empfangsmethode.....	28		
Überschreiben	28		
Kopie mehrerer Dateien	28		
Der Fingerabdruck von DOS	29		
Eine Datei umbenennen	29		
Eine Datei verschieben	30		
Den Prozeß abbrechen	30		
Interne und externe Befehle	31		
Sicherheitskopie	31		
Attribute	32		
Der Letzte macht das Licht aus.....	33		
Anwendung einer Sicherheitskopie	34		
Bearbeitung einer Textdatei	35		
Eine Datei löschen	37		
Undelete.....	37		
Doppelte Sicherung.....	38		
Ein Verzeichnis mit Dateien löschen	39		
Disketten	40		

Mein PC und ich

Vor etwa fünf Jahren suchte ich eine gebrauchte elektronische Schreibmaschine. Ich hatte einige Manuskripte in der Mache und war meine alte Olivetti leid. Ein Bekannter hatte damals einen PC, den er nicht brauchte – den kaufte ich mir.

So machte ich Bekanntschaft mit DOS – und hatte das Gefühl, auf einem anderen Stern zu sein.

Nach einem Jahr ärgerte ich mich, daß ich nicht schlicht eine Schreibmaschine gekauft hatte. Mein Nadeldrucker ratterte gotterbärmlich, verschlang meterweise Farbbänder und schrieb unvorhersehbare Hieroglyphen anstelle von ä, ö und ü. Mein Bestand an Handbüchern und Anweisungen wuchs und wuchs – selber schrieb ich weniger und weniger.

Allmählich wurde mir allerdings auch klar, daß der Fehler nicht unbedingt bei mir lag. Ich entdeckte, daß ich – genau wie tausend andere verwirrte PC-Benutzer – genug davon hatte, unverständliche Gebrauchsanweisungen in unbegreiflichen Sprachen zu lesen.

Später überwand ich dann meinen Widerstand und kaufte einen größeren Computer, mit dem ich mich allmählich vertraut gemacht habe; aber es erstaunt mich immer wieder, wie selten brauchbare Gebrauchsanweisungen für einen PC sind.

Deshalb war es eine positive Überraschung, als mir eines Tages Michael Maardts Heft *Nutze Deinen PC optimal* in die Hand fiel. Das war etwas, was ich brauchen konnte – und es kostete nur 5 Mark. Also schrieb ich ihm. Das führte zu einer Zusammenarbeit, deren Ergebnis unter anderem dieses Anfängerheft zu DOS ist.

Ich bin 49 Jahre alt und Architekt. Meinen PC verwende ich vor allem zum Schreiben und Zeichnen. Meine Kinder, 6 und 9 Jahre alt, und ihre Freunde benutzen ihn für Spiele.

Im Lauf der letzten Jahre haben sich etliche meiner Freunde ebenfalls einen PC angeschafft. Ein Blick auf mein Bücherregal zeigt ihnen, daß ich mich mit diesem Thema befaßt habe – also fragen sie mich regelmäßig um Rat. Und ihre Fragen machen mir klar, wo die meisten Schwierigkeiten liegen.

In diesem Heft habe ich mich meinem Thema von zwei Seiten genähert. Einerseits befaße ich mich mit einigen praktischen Problemen, denen die meisten Leute begegnen.

Andererseits untersuche ich nur einige typische Aspekte von DOS gründlich und versuche nicht, alles mitzubekommen.

Dieser zweite Punkt – die Gründlichkeit – hat mir die größten Schwierigkeiten bereitet. Dabei stellte ich mir oft vor, daß der Leser die unangenehme, aber ziemlich relevante Frage stellte: *Und was kann ich damit anfangen?*

Es ist mir wichtig, daß du regelmäßig ein Aha-Erlebnis hast – aber leider ist DOS nicht gerade etwas für Anfänger.

Du kannst viel dazu beitragen, die kommenden Ausgaben dieses Heftes zu verbessern, indem du mir schreibst. Klagen, Jubel, Kritik, Vorschläge – alles ist herzlich willkommen. Verbesserungen lassen sich eben immer anbringen!

Schreib mir via die Adresse des Verlags.

Sprache und Schreibweise

Der Text enthält mehr Englisch, als mir lieb ist. Das hängt damit zusammen, daß DOS in den USA entwickelt wurde.

Hast du eine deutsche Ausgabe von DOS, werden bestimmte Mitteilungen auf deinem Bildschirm auf Deutsch erscheinen – die Befehle werden aber immer noch auf den englischen Wörtern aufbauen.

Ich habe alle Befehle und Dateinamen mit COURIER und Großbuchstaben geschrieben, um sie vom übrigen Text abzuheben. In deinen Befehlen kannst du frei zwischen Groß- und Kleinbuchstaben wählen.

Lies mit den Fingern

Du solltest dieses Heft nicht auf einen Satz verschlingen und annehmen, nun wüßtest du genauer Bescheid. Setz dich in aller Ruhe mit einer Tasse Kaffee vor deine Maschine, während du liest. Etliche Punkte sollten ausprobiert werden, damit sie "sitzen".

Viel Spaß!

Steen Juhler

Was tut DOS eigentlich?

Soll ein Computer funktionieren, muß er ein Betriebssystem haben, mit dessen Hilfe seine einzelnen Teile zusammenarbeiten. Dieses Betriebssystem sorgt dafür, daß die Informationen, die durch den Computer wandern, am richtigen Ziel ankommen. Es ist nicht in den Computer eingebaut, sondern besteht aus einer Reihe von Anweisungen, die jedesmal, wenn die Maschine startet, in den Speicher gelesen werden.

Das Betriebssystem DOS (Disk Operative System) wurde für einen Heimcomputer hergestellt, einen PC (Personal Computer), den IBM 1981 herausbrachte. IBM benötigte für die neue Maschine ein maßgeschneidertes Betriebssystem und suchte einen geeigneten Produzenten.

Die Entwicklung des Systems wurde Microsoft anvertraut. Das war damals eine bescheidene Garagenfirma. Bill Gates und seine Kollegen vollendeten DOS; mit über 100 Millionen verkaufter Exemplare wurde dieses System allmählich zum weltweit vorherrschenden Betriebssystem für PCs.

Der erste PC machte Schule und bekam viele Nachfolger. Heute werden alle DOS-basierten Computer, die IBM-kompatibel sind, PCs genannt.

Bill Gates hatte die Absicht, ein Betriebssystem herzustellen, das sogar seine Mutter benutzen konnte. Nach vielen Jahren Erfahrung steht wohl fest, daß DOS die Benutzerfreundlichkeit noch nie zu seinen Pluspunkten rechnen konnte.

Dieses Betriebssystem wurde von Intellektuellen erdacht, die sich nur schwer vorstellen konnten, wie unlesbare Kodierungen auf normale Leute wirken. Sein Erfolg liegt vermutlich an der Verbindung von technologischer Entwicklung und der simplen Logik, die es trotz allem darstellt.

Zu den sympathischsten Zügen von DOS gehört, daß das System "rückwärts-kompatibel" ist. Das bedeutet, daß jede neue Ausgabe oder Version von DOS auf einer älteren Maschine laufen kann. DOS hat die Entwicklung zu schnelleren und größeren Computern mitgemacht, ohne die älteren Modelle zu vernachlässigen. Während das System mit immer neuen Fähigkeiten ausgestattet wurde, änderte sich sein Kern im Grunde nicht. Das hat allerdings seinen Preis. Zum Beispiel ist die Speichergröße, die DOS handhaben kann, ziemlich bescheiden im Vergleich zu den Möglichkeiten moderner Maschinen. Natürlich hat man das Problem gelöst; aber die Kunst, die alten, viel zu schmalen Kleider an die neuen Megakörper anzupassen, wird immer komplizierter. Unter anderem darum ist die Glanzzeit

von DOS vorbei. Längst wurden neue und effektivere Betriebssysteme entwickelt, die für den Benutzer unsichtbar im Hintergrund arbeiten. Nur sollte man daran denken, daß auch diese Systeme von Intellektuellen erstellt wurden, die nun behaupten, sie kennen die Bedürfnisse der Benutzer im voraus. Die Systeme werden *benutzerfreundlicher*, aber weniger *benutzergesteuert*. DOS dagegen liegt immer noch an einer passenden Stelle zwischen dem Unbeholfenen und dem Praktischen.

Bevor wir uns genauer mit diesem System auseinandersetzen, wollen wir uns kurz die einzelnen Teile eines Computers ansehen.

Hardware

Die verschiedenen Typen kleiner Computer bestehen grundsätzlich aus ein und derselben Art Apparatur oder Hardware; hier sehen wir uns einen PC an.

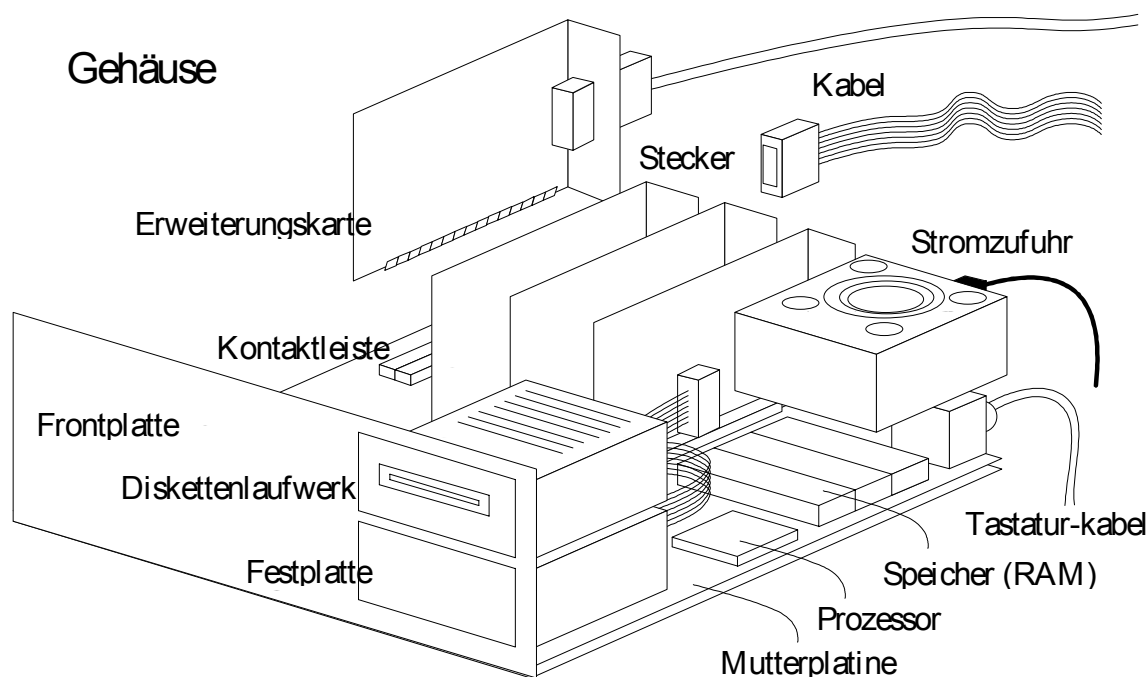
Das Gehäuse

Alle Teile eines PCs sind über Leitungen und Kabel mit dem Gehäuse verbunden. Hier sammeln sich die Fäden.

Die Stromzufuhr wird in eine normale Steckdose gesteckt werden. Du solltest dafür sorgen, daß der Kontakt nicht zufällig unterbrochen werden kann. Viele der neueren Programme können zwar abstürzen, ohne daß gleich alles verschwindet – aber so ein Abbruch macht nie Spaß.

Viele PCs haben auf der Rückseite des Gehäuses einen Stromausgang. Den kannst du für die Stromversorgung des Bildschirms verwenden, so daß Monitor und Gehäuse gleichzeitig an- oder ausgeschaltet werden.

Wenn ein PC über den Stromschalter eingeschaltet wird, erfolgt ein sogenannter Kaltstart. Das hörst du unter anderem daran, daß die Festplatte auf Touren kommt. Zwischen einem Stromabbruch und einem Kaltstart müssen mindestens 10 Sekunden vergehen, da die Festplatte erst einmal zur Ruhe kommen muß. Aktivierst du dagegen den Reset-Knopf oder drückst **Strg+Alt+Entf** (siehe Seite 43), während der PC läuft, dann erfolgt ein sogenannter Warmstart. Hier rotiert die Festplatte während des Starts weiter. Bei beiden Startformen wird der gesamte Inhalt des Speichers gelöscht, und das System muß erneut eingelesen werden.



Außer dem transportablen Modell, das wir hier nicht beschreiben, gibt es zwei Haupttypen von Gehäusen: das Desktop- oder Tischmodell steht auf seiner breiten Fläche; der Bildschirm wird dann darauf gestellt. Das Tower- oder Standmodell steht senkrecht neben dem Bildschirm oder auf dem Boden. Der Vorteil eines Towers ist mehr freier Platz: auf dem Schreibtisch sowie im Innern des Towers – für zusätzliche Teile wie etwa Festplatten oder Erweiterungskarten. Um Platz zu sparen, kannst du ein Desktop-Gehäuse ohne weiteres auf die Seite stellen. Nur mußt du dafür sorgen, daß die Festplatte nicht auf dem Kopf steht.

Die Stromzufuhr des PCs sitzt am hinteren Ende des Gehäuses, neben einem Ventilator. Diskettenlaufwerk und Festplatte sind in ihrem jeweiligen Metallkasten hinter der Vorderseite des Gehäuses angebracht. Im allgemeinen hat jedes Laufwerk ein Lämpchen, das leuchtet, wenn Daten von der Platte geholt oder auf ihr gespeichert werden.

Die wichtigsten elektronischen Bestandteile des Computers sind auf der Mutterplatine, auf englisch: Motherboard, montiert; im Desktop-Modell sitzt diese unten im Gehäuse, im Tower-Modell steht sie senkrecht an der Seite. Hier finden wir den Prozessor, auch die CPU=Central Processing Unit genannt, und den Speicher, das RAM = Random Access Memory).

Der Prozessor, der mit seinen hunderttausenden von Transistoren für die gesamte Datenbearbeitung sorgt, entspricht in seiner Größe ungefähr einem

Stück After Eight-Schokolade; die RAM-Bausteine, die durch Einzelteile ergänzt werden können, einer normalen Tafel Schokolade.

Andere Komponenten sind auf kleinere Karten montiert, die senkrecht zur Mutterplatine stehen. Zum Beispiel hat die Festplatte eine spezielle Karte, einen sogenannten Controller, wie auch der Bildschirm eine Bildschirmparte. Wird die Festplatte durch eine größere oder schnellere ersetzt, kann die Controllerkarte ebenfalls ausgetauscht werden. Entsprechend gilt das für die Bildschirmparte, die unterschiedliche Eigenschaften haben kann. Diese Karte hat ihren eigenen Speicher; seine Größe spielt eine Rolle für die Fähigkeit des Bildschirms, komplizierte Bildverläufe zu steuern. Die kleineren Karten sind mit länglichen vielbeinigen Steckern an der Mutterplatine befestigt, die auf Englisch *Slots* heißen. Auf diese Weise kann die Elektronik der Mutterplatine mit Soundcards, CD-ROM-Laufwerken, Faxmodems und anderem erweitert werden, soweit im Gehäuse Platz dafür ist.

Die kleineren Karten haben auf der Rückseite des Gehäuses Anschlüsse für externe Kabel.

Der Bildschirm

Der Bildschirm oder Monitor eines PCs muß der Bildschirmkarte entsprechen, die im Gehäuse sitzt.

Während die technischen Daten des Monitors für die Größe und Stabilität des Bildes Bedeutung haben, bestimmt die Bildschirmkarte, wie feinkörnig und schnell sich das Bild verhält.

Früher wurde der Bildschirm vor allem für Text verwendet; heute dagegen enthalten die meisten Programme viele graphische Effekte und sind reich an Farben.

Das stellt große Anforderungen an die Auflösung des Bildschirms – wie feinkörnig er ist, wie viele Punkte, sogenannte Pixels, auf ihm erscheinen. Die meisten Farbmonitore verfügen heute über eine Auflösung nach VGA oder Super VGA (SVGA). Ein Farbmonitor mit VGA-Auflösung kann 640 x 480 Punkte in 256 Farben zeigen – ein SVGA-Monitor 1024 x 768 Pixel in bis zu 16 Millionen Farben. Je mehr Farben und Pixel gezeigt werden, desto zäher wird die Arbeit an graphischen Darstellungen. Der Engpaß ist hier unter anderem die Bildschirmkarte. Ein Standard-Monitor ist heute normalerweise 14 Zoll groß.

Wenn du mit DOS-Befehlen arbeitest, läuft das in der Regel in reinem Text ohne Bilder ab. Hier kannst du dich also mit geringer Bildauflösung begnügen.

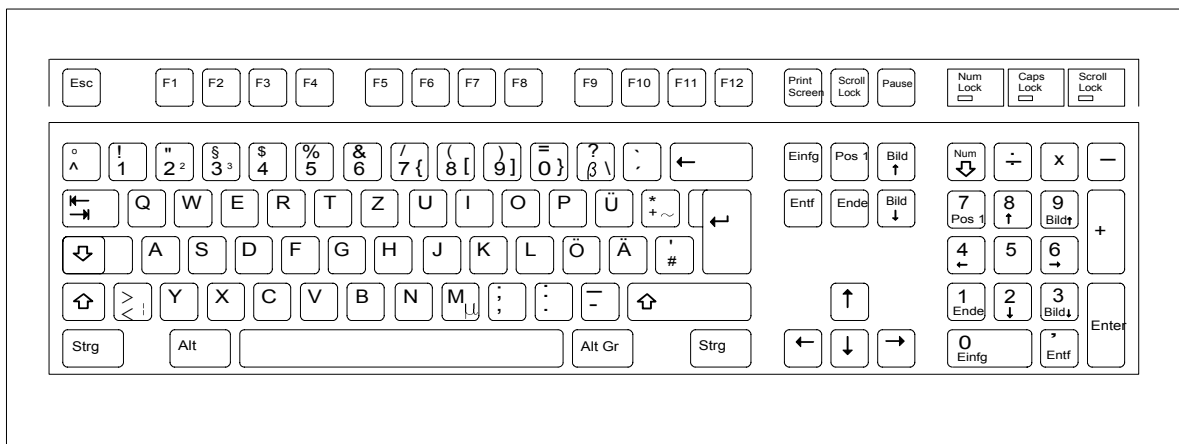
Die Tastatur

Die Tastaturen der ersten PCs erkennt man daran, daß auf ihnen die Funktionstasten F1-F10 links von den übrigen Tasten sitzen. Heute verwendet man in der Regel eine sogenannte erweiterte Tastatur mit 102 Tasten, bei der die Funktionstasten am oberen Rand sitzen.

Bestimmte Tasten gibt es in zwei Ausgaben, zum Beispiel die wichtige Eingabe-Taste, die Pfeiltasten und die Zahlen. Die Zahlentasten an der rechten Seite der Tastatur sind zum Beispiel für schnelle Eingaben von Kalkulationen berechnet und ergeben ausschließlich Zahlen, wenn die NumLock-Funktion aktiviert ist – in diesem Falle leuchtet das entsprechende Licht. Ist NumLock nicht aktiviert, haben diese Tasten die Bedeutung, die unten auf ihnen angegeben ist, zum Beispiel Ende statt 1.

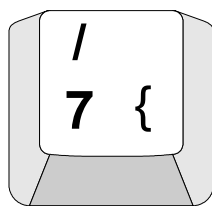
Die meisten übrigen Tasten haben je nach Aktivierung der Umschalttaste (mit dem breiten Aufwärts-Pfeil) eine von zwei Bedeutungen – zum Beispiel Groß- oder Kleinbuchstaben. Die Taste mit dem großen Abwärts-pfeil, die *permanente Umschalttaste (CapsLock)*, setzt alle Buchstaben als Großbuchstaben, übt diese Wirkung aber nicht auf die anderen Tasten aus. Willst du zum Beispiel ein Prozentzeichen schreiben, das auf der Tastatur über 5 sitzt, mußt du gleichzeitig die Umschalttaste drücken, auch wenn CapsLock aktiviert ist.

Die Taste, die als Alt Gr (Alternative Graphics) bezeichnet wird – sie sitzt rechts von der Leertaste – veranlaßt bestimmte Tasten, ein besonderes Zeichen einzugeben, das unten rechts auf der jeweiligen Taste steht.



Eine normale deutsche Tastatur mit 102 Tasten

Die hier gezeigte Taste kann drei verschiedene Zeichen eingeben. Wird sie allein gedrückt, erscheint die Zahl 7. Wird gleichzeitig die Umschalttaste gedrückt, erscheint ein Schrägstrich; ist es die Alt Gr, erscheint das Zeichen {.



Zu den zahlreichen Spezialtasten läßt sich ganz allgemein nicht viel sagen, da sie je nach Programm völlig verschiedene Bedeutungen haben. Später werden wir uns einige Bedeutungen ansehen, die in DOS mit ihnen verbunden sind.

Auf der deutschen Tastatur gibt es die Buchstaben Ä, Ö und Ü; das bedeutet aber leider nicht, daß sie in allen Programmen diese Bedeutung haben. Wird dein PC grundsätzlich darauf eingestellt, eine deutsche Tastatur zu benutzen, haben die Tasten diese Bedeutung, so lange du in DOS arbeitest. Startest du aber ein englischsprachiges Programm, das diese Buchstaben nicht in der Tastaturbesetzung hat, kann es problematisch werden.

Für alle Tasten gilt, daß ihre Bedeutung je nach Bedarf neu definiert werden kann.

Die Maus

Die Maus gehörte nicht immer zur Standardausstattung eines PCs; das ändert sich aber allmählich. Mehr und mehr Programme sind mit ihrer Hilfe leichter zu bedienen. Wenn du sie über eine spezielle Unterlage bewegst, bewegt der Zeiger (Cursor) sich entsprechend auf dem Bildschirm. Das macht sie sehr praktisch, wenn es darum geht, Objekte in Menüs, Spielen und graphischen Programmen auszuwählen und zu verschieben.

Eine Maus hat zwei oder drei Schalter; der linke hat normalerweise die gleiche Bedeutung wie die Eingabetaste. In bestimmten Programmen kann die Funktion der Schalter auf häufig angewandte Funktionen des Programms umdefiniert werden. Arbeitet man zum Beispiel linkshändig mit der Maus, so empfiehlt es sich, die Bedeutungen des rechten und des linken Schalters auszutauschen.

Arbeitest du mit DOS so, wie ich es in diesem Heft beschreibe, benötigst du keine Maus.

Der Drucker

Ein PC läßt sich über einen parallelen oder seriellen Ausgang mit einem Drucker verbinden.

Bei einer parallelen Verbindung hat das Kabel mehrere Leiter, über die mehrere Signale gleichzeitig gesendet werden können.

Über die serielle Verbindung wird jeweils nur ein Signal gesendet.

Die drei gebräuchlichsten Druckertypen sind der Nadeldrucker, der Tintenstrahldrucker und der Laserdrucker.

☞ Der *Nadeldrucker*, der auch Matrixdrucker genannt wird, schreibt, indem er wie eine Schreibmaschine kleine Stifte in das Farbband drückt. Der Druckkopf saust über das Papier; je mehr Nadeln der Druckkopf enthält, desto detaillierter wird das Ergebnis. Für Farbdrucke wird ein mehrfarbiges Band verwendet. Der Nadeldrucker hat den Vorteil, daß er bei Formularen und ähnlichem Durchschläge ermöglicht. Außerdem ist er sehr preiswert. Mit Graphik kann er nicht besonders umgehen; außerdem macht er während der Arbeit ziemlichen Lärm.

☞ Der Druckkopf eines *Tintenstrahldruckers* unterscheidet sich von dem des Nadeldruckers dadurch, daß er durch kleine Düsen Tinte aufs Papier sprüht. Dieser Druckertyp wird auch Inkjet genannt. Seine Düsen sitzen dicht nebeneinander und arbeiten sehr genau. 300 dpi (Dots per Inch) werden problemlos geliefert. Mehrere Modelle können mit Hilfe mehrfarbiger Tintenpatronen Farbdrucke liefern. Ein Tintenstrahldrucker arbeitet recht leise und ist außerdem relativ billig.

☞ Ein *Laserdrucker* liefert die besten Ergebnisse. Er arbeitet nach dem Prinzip des Fotokopierens, bei dem durch statische Elektrizität Kohlenstaub aufs Papier gedrückt wird. Auflösungen von 600 dpi oder mehr sind nicht ungewöhnlich, und Farben werden in guter Qualität wiedergegeben. Die meisten Laserdrucker werden mit eingebautem Speicher geliefert, der als Puffer zwischen PC und Drucker arbeitet.

Außerdem existiert ein Typ dieses Druckers, der mit Postscript arbeitet – einer graphischen Programmiersprache, die die Fertigstellung einer Graphik in den Drucker verlegt. Manche Laserdrucker können durch ihre hohe Spannung Ozon erzeugen; dazu gehört allerdings nicht der LED-Typ.

Laserdrucker arbeiten schnell und leise und werden immer billiger.

Der Boot-Prozess

Wenn du deinen PC anschaltest, führt er einige vorbereitende Aufgaben aus, um für die eigentliche Arbeit bereit zu sein. So gut wie alle Programme, die in ihm angewandt werden, müssen von einer Diskette oder der Festplatte in den Speicher transportiert werden. Einzige Ausnahme sind die vorbereitenden Aufgaben, die durch eingebaute Programme erledigt werden. Diese Programme nennt man BIOS (Basic Input Output System); sie sind als Teil der Hardware des Computers in einem Microchip auf der Mutterplatine untergebracht. Die erste Aufgabe ist eine Prüfung der übrigen Hardware. Arbeitet der Speicher? Rotiert die Festplatte? Sind Monitor und Tastatur angeschlossen? Erst wenn dieser Power On Self Test (POST) beendet ist, beginnt das eingebaute Programm seine Suche nach einem Betriebssystem, mit dem es den Start fortsetzen kann.

Das BIOS sieht zunächst nach, ob sich auf einer Diskette im Diskettenlaufwerk ein System findet. Ist das nicht der Fall, wird die Festplatte untersucht. Wenn das Betriebssystem gefunden ist, muß es in den Speicher eingelesen werden. Das nennt man: die Maschine booten. Das hat nichts mit Schiffen zu tun; Boot (engl.) bedeutet Stiefel und wird als Bild dafür gebraucht, daß der Computer sich am Stiefelriemen aufwärts zieht. Ohne ein Betriebssystem ist ein Computer so hilflos wie Baron Münchhausen, der sich an seinen Haaren aus dem Wasser ziehen wollte. In der amerikanischen Version der Geschichte sind es allerdings nicht die Haare, sondern die Stiefelriemen ...

Nach dem Booten ist die Maschine bereit, die gestellten Aufgaben durchzuführen.

Du kannst DOS auf zwei verschiedene Weisen anwenden: entweder als Betriebssystem im Hintergrund für die aktiven Programme – oder direkt mit Hilfe von Befehlen. Oft wird der PC so eingestellt sein, daß er automatisch ein Programm startet, wenn die Maschine bootet. Es ist zum Beispiel gar nicht ungewöhnlich, mit einem Menü oder mit Windows zu beginnen.

Diese Programme arbeiten mit DOS zusammen, das laufend im Hintergrund als Betriebssystem funktioniert.

Willst du direkt mit DOS in Verbindung treten, mußst du zunächst die Programme beenden und Platz freimachen für die Klarmeldung von DOS.

Klar

Bevor du direkt mit DOS arbeiten kannst, mußt du folgende Mitteilung auf deinem Bildschirm sehen:

```
C:\>
```

Das ist die Klarmeldung von DOS.

Die Zeichen `C:\>` werden Prompt genannt; sie machen darauf aufmerksam, daß die Maschine bereit ist, DOS-Befehle entgegenzunehmen. Prompt bedeutet auf Englisch Stichwort.

Hinter dem letzten Zeichen des Prompt `>` siehst du einen blinkenden Strich-Cursor, der anzeigt, wo du schreiben kannst. Dieser Cursor bewegt sich im Rhythmus deiner Tastenanschläge nach rechts.

Schreibe nun das Wort `DATE` nach dem Prompt und taste Eingabe.

```
C:\>DATE
```

DOS reagiert augenblicklich auf den Befehl, indem es das aktuelle Datum auf dem Bildschirm angibt. In der deutschen Ausgabe wäre das etwa

```
Gegenwärtiges Datum: Di, 30-05-1995
Neues Datum (TT-MM-JJ):
```

In diesem Heft werden der Prompt und andere Mitteilungen von DOS mit `MAGERER` Schrift angegeben – die Befehle, die du selbst eintastest, werden **FETT** geschrieben.

Der Cursor steht im Augenblick hinter der zweiten Zeile; hier kannst du das Datum berichtigen, falls es verkehrt sein sollte. Die Klammer sagt dir, daß du das Datum als Tag-Monat-Jahr mit jeweils zwei Zeichen schreiben solltest. Soll es auf den 29. Mai 1995 geändert werden, schreibst du

```
29-05-95
```

und tastest Eingabe. DOS setzt selber den richtigen Wochentag ein, sobald das Datum gezeigt wird.

Wiederhole nun den Befehl

C:\>**DATE**

Gegenwärtiges Datum: Mo, 29-05-1995
Neues Datum (TT-MM-JJ):

Ist das Datum korrekt, tastest du schlicht Eingabe;
hierdurch erscheint erneut der Prompt

C:\>

Ein entsprechender Befehl ist **TIME** – er zeigt die
augenblickliche Uhrzeit

C:\>**TIME**

Gegenwärtige Uhrzeit: 12:25:36.70p
Neue Uhrzeit:

Beim Cursor kannst du eine genaue Zeitangabe
schreiben und dann im richtigen Augenblick
Eingabe tasten.

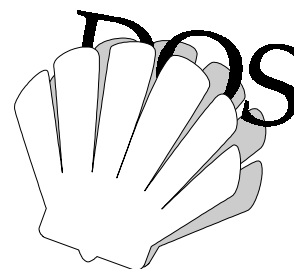
Ist die angegebene Zeit korrekt, tastest du
Eingabe, ohne etwas zu schreiben.

Die Werte für Datum und Uhrzeit werden in einem Mikrochip auf der Mutterplatine aktualisiert, dem sogenannten CMOS – zusammen mit den übrigen Einstellungen für das eingebaute BIOS-Programm. Um diese Angaben zu bewahren, wird Strom von einer kleinen Batterie genommen, die durch den Strom des angelegten Computers laufend wieder aufgeladen wird. Mehr dazu auf Seite 59.

DATE und **TIME** sind Beispiele für Befehle, die du direkt im DOS schreiben kannst. Die Zeile auf dem Bildschirm, die mit dem Prompt beginnt, wird die Befehlszeile genannt.

Shell

DOS-Befehle haben den großen Nachteil, daß man sie sich nur schwer merken kann. Die wenigsten haben Lust, in einem Handbuch nachzuschlagen, wenn sie einen Prompt beantworten müssen. Du kannst nun aber ein besonderes Programm benutzen, das sich “um DOS legt” und die Befehlszeile ersetzt.



Ein Programm, das DOS-Befehle in ein Menü übersetzt, wird Shell genannt – das englische Wort für Schale.

Ein Beispiel für eine solche Shell ist **DOS-SHELL**, das mit einigen Versionen von MS-DOS ausgeliefert wird, sich aber nicht gerade durch Benutzerfreundlichkeit auszeichnet. Besser gelungen sind zum Beispiel **PC-TOOLS** und **NORTON COMMANDER**.

Der letztere ist mein eigener Favorit – ich könnte dieses Programm nur schwer entbehren.

Ich kann die Anwendung einer solchen Shell nur empfehlen – sie macht DOS wesentlich einfacher und zugänglicher. Außerdem lernst du es durch den Gebrauch einer Shell besser kennen. Wenn du obendrein die in diesem Heft beschriebenen Möglichkeiten im DOS kennst, so wird das ganze zu einem Spiel!

Eine kurze Beschreibung von Norton Commander findest du im Heft
Nutze deinen PC optimal.

Befehle nicht vergessen

Irgendwann wirst du mit Sicherheit einen Befehl wiederholen, den du gerade erst eingegeben hast. Ein Druck auf die F3-Taste wiederholt den letzten Befehl, den du gegebenenfalls berichtigen kannst.

Schreibst du viele DOS-Befehle, wird dir das Programm DOSKEY helfen, das die letzten Eingaben der Befehlszeile speichert. Es wird aktiviert, indem du schreibst

`C:\>DOSKEY` ..und Eingabe tastest.

Du kannst nicht sehen, daß das Programm läuft – auf dem Bildschirm ändert sich nichts. DOSKEY arbeitet im Hintergrund und merkt sich still und leise die Befehle, mit denen du den Prompt beantwortest. Ein Druck auf eine Pfeiltaste holt einen früheren Befehl hinter den Prompt; drückst du mehrmals auf den Aufwärtspfeil, blätterst du rückwärts durch deine gegebenen Befehle. Der Abwärtspfeil bringt dich wieder zurück.

Erreichst du einen Befehl, den du wiederverwenden willst, tastest du Eingabe, um ihn auszuführen.

Du hast auch die Möglichkeit, einen früheren Befehl zu berichtigen. Hast du den entsprechenden Befehl über den Aufwärts- oder Abwärtspfeil erreicht, kannst du den Cursor seitwärts durch den Text bewegen. Drückst du auf die Einfg-Taste, kannst du deine neuen Zeichen entweder zwischen die existierenden setzen oder sie überschreiben. Einfg ist eine Abkürzung für Einfügen. Der berichtigte Befehl wird durch die Eingabetaste aktiviert und erscheint in der Liste deiner geschriebenen Befehle.

Normalerweise kann DOSKEY 512 Zeichen speichern, was zwanzig bis dreißig Befehlen entspricht. Drückst du F7, erscheint auf dem Bildschirm eine Liste der Befehle, die du nach der Aktivierung von DOSKEY geschrieben hast.

DOSKEY ist ein nützliches kleines Programm; mit seiner Hilfe kannst du deinen Computer bei jedem Anschalten automatisch hochfahren. Mehr dazu auf Seite 46.

Version

Schreibe nun den Befehl `VER` nach dem Prompt

`C:\>VER`

und taste Eingabe.

Auf dem Bildschirm wird dir mitgeteilt, mit welcher DOS-Version du augenblicklich arbeitest – zum Beispiel

MS-DOS Version 6.2

Das Betriebssystem DOS wird heute von mehreren Firmen hergestellt, die ihr eigenes Zeichen an den Anfang des Namens setzen. PC-DOS stammt von IBM, DR-DOS von Digital Research (später Novell). Das meistverbreitete Produkt stellt Microsoft unter dem Namen MS-DOS her. Dies ist die Ausgabe, mit der wir uns in diesem Heft befassen.

Die Zahlen nach dem Namen geben die Versionsnummer an. Wir beschäftigen uns mit den Versionsnummern, die mit 5 oder 6 beginnen. Hinter dem Punkt können weitere Ziffern erscheinen, etwa 6.0, 6.2 oder 6.21.

Das Betriebssystem DOS arbeitet als Muttersprache und Werkzeugkasten in deinem PC.

Während sich die fundamentale Sprache kaum von einer Version zur nächsten verändert, werden beständig neue Hilfsmittel hinzugefügt. Eine neue Version bedeutet also nicht unbedingt, daß deine Maschine besser oder schneller wird – sie gibt dir vermutlich eher neues Zubehör.

Zu den Dingen, auf die DOS-Befehle sich wirklich verstehen, gehört die Verteilung deiner Dateien auf Disketten und Festplatten. Nun wollen wir uns ansehen, was eine Datei ist; zunächst solltest du aber deinen Bildschirm freimachen, indem du schreibst

`C:\>CLS` ..und Eingabe tastest.

Nachfolgend werde ich nicht jedesmal erwähnen, daß du Eingabe tasten muß, um einen Befehl auszuführen.

CLS bedeutet CLear Screen; es gibt dir ein neues Bild auf dem Schirm mit dem Prompt in der oberen Hälfte.

Dateien

Das Wort Datei steht für den langweiligen Begriff Aktenstück. Zum Glück sind die Dateien selbst keineswegs langweilig!

Eine Datei entspricht einem Blatt Papier mit Inhalt. Das kann ein Drudel sein, ein Gedicht, eine Zeichnung; eine Notiz oder eine Kalkulation; ein Strickmuster, ein Musikstück oder ein Brief. Statt Papier verwendet ein Computer Dateien.

Eine Datei besteht aus Zeichen; aus Buchstaben, Zahlen oder Kodierungen. In ihrer Grundform, der Textdatei, besteht sie ausschließlich aus den Zeichen, die du auf einer Schreibmaschine verwendest. Dateien lassen sich leichter bearbeiten als Papier – sie können elektronisch gespeichert und verändert werden. Es ist einfach, ihren Inhalt zu verändern, und sie lassen sich schnell verschieben. Außerdem nehmen sie nicht soviel Raum ein wie Papier.

Außer Text können Dateien auch Kodierungen enthalten. Diese nehmen eine Sonderstellung ein, da sie komplizierte Prozesse im Computer starten können.

Programme und Dokumente

Es gibt zwei verschiedene Arten von Dateien: Programme und Dokumente.

Ein *Programm* enthält die Anweisungen, die es dir ermöglichen, bestimmte Aufgaben auf deinem PC durchzuführen – zum Beispiel einen Brief zu schreiben. Der Inhalt eines Programmes ist jedesmal, wenn du es verwendest, derselbe.

Ein *Dokument* ist das Ergebnis deiner Arbeit im Programm – zum Beispiel eine Zeichnung oder ein Brief.

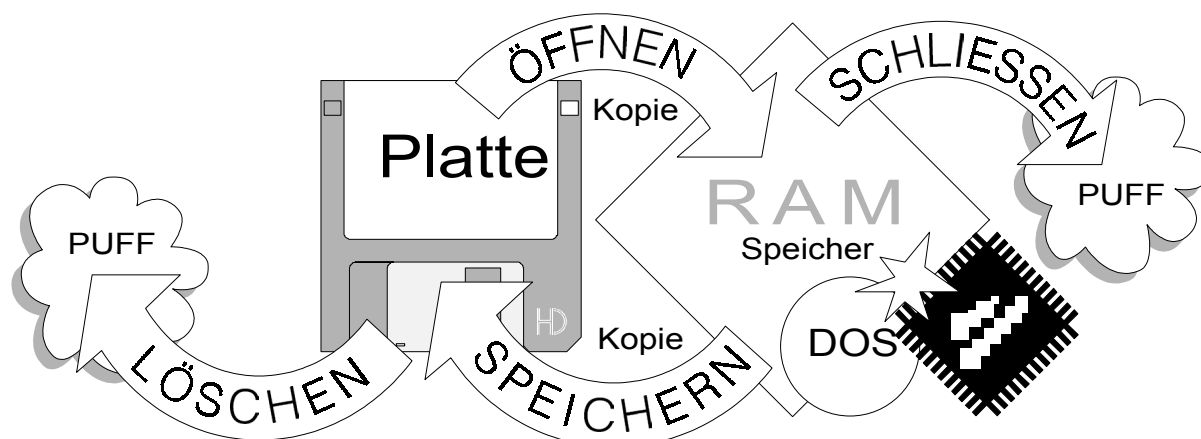
Der Inhalt eines Dokuments ändert sich in dem Umfang, wie du an ihm arbeitest.

Deine Arbeit am PC ist also eine Wechselwirkung zwischen Programm und Dokument. Darum müssen beide Dateientypen gleichzeitig zur Verfügung stehen. Der Ort im PC, wo sich die Dateien zur Zusammenarbeit treffen, wird *Speicher* genannt.

Ursprünglich werden die Dateien in einem Lager aufbewahrt, von wo sie bei Bedarf geholt werden. Dieses Lager kann die Festplatte sein oder eine Diskette.

Prinzipiell gibt es keinen Unterschied zwischen Festplatte und Diskette. In der Praxis ist die eingebaute Festplatte schneller und geräumiger als eine Diskette; beide dienen aber als Lager für Dateien. Im Folgenden verwende ich die gemeinsame Bezeichnung *Platte* für Festplatte und Diskette.

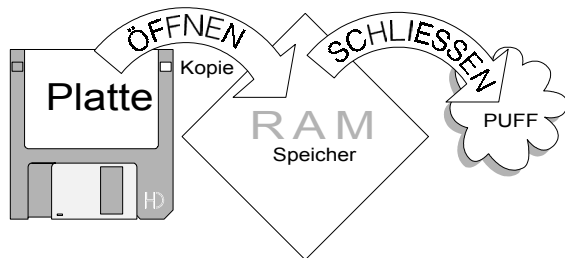
Soll ein Dokument mit Hilfe eines Programms bearbeitet werden, werden beide Dateien von der *Platte* in den Speicher geholt. Das geschieht, indem dort eine Kopie der auf der Platte befindlichen Dateien erstellt wird. Nun gibt es beide Dateien zweimal: in der *geschlossenen* Ausgabe auf der Platte und der *offenen* im Speicher. Während die geschlossene Ausgabe passiv auf Lager liegt, ist ihre jeweilige offene Ausgabe im Speicher überaus aktiv. *Der Speicher* wird auch RAM (Random Access Memory) genannt, was *Gedächtnis mit willkürlichem Zugang* bedeutet – das heißt, sein Inhalt kann frei ummöbliert werden.



Kopien der Dateien werden zwischen Platte und Speicher ausgetauscht

Die schöpferische Arbeit besteht nun darin, Teilstücke der offenen Dateien im Speicher auseinanderzureißen und zusammenzufügen. Das Programm liefert das Rezept dafür, wie die verschiedenen eingelaufenen Daten im Prozessor des Computers zu behandeln sind. "Die Taste mit dem Buchstaben A wurde aktiviert. Dieser Buchstabe ist als erstes Zeichen der Zeile 14 im Brief an Tante Sophie einzusetzen" und so weiter.

Hast du dich zu einem Ergebnis kombiniert, mit dem du zufrieden bist, kannst du deine Aufgabe beenden. In der Regel ist es nicht notwendig, ein Programm zu speichern, da es vermutlich das nächste Mal, wenn du schreiben willst, in seiner ursprünglichen Form verwendet wird. Und es liegt ohnehin als geschlossene Datei auf deiner *Platte*. Also beendest du das Programm, indem du es schlicht aus dem Speicher entfernst!



Nach dem Gebrauch wird das Programm aus dem Speicher entfernt

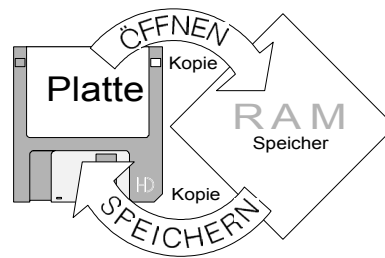
Das geänderte Dokument dagegen muß für den späteren Gebrauch *gespeichert werden*.

Leider enthält die deutsche Sprache eine Falle, was den Begriff "eine Datei speichern" betrifft.

Unmittelbar würde man annehmen, daß eine Datei *gespeichert* wäre, wenn sie sich im *Speicher* befindet. Das ist aber nicht so – ganz im Gegenteil ist der Speicher der gefährlichste Ort, wo sich eine Datei aufhalten kann!

Wird der Strom im PC nur eine Sekunde unterbrochen, ist der gesamte Inhalt des Speichers auf immer verloren! Also solltest du deine Arbeit so schnell wie möglich an einen sicheren Ort kopieren: auf eine *Platte*.

Eine Datei speichern bedeutet, sie vom *Speicher* auf eine *Platte* zu überführen..



Die Kopie des geänderten Dokuments kehrt auf die Platte zurück

Erst wenn sich die Datei auf einer *Platte* befindet, kann sie in Frieden ruhen – unabhängig von den Launen der Elektronik.

Auch bei mir hat es lange gedauert, bis ich mich an die Arbeitsteilung zwischen Speicher und Platte gewöhnte.

Es ist nicht immer ganz einfach, genau zu überblicken, was sich zu einem bestimmten Zeitpunkt im Speicher des Computers befindet. Grundsätzlich kannst du damit rechnen, daß das, was du auf dem Bildschirm siehst, zum Inhalt des Speichers gehört.

Dateinamen

Jede Datei hat einen Namen, durch den sie von den anderen unterschieden wird.

Ein Dateiname besteht aus einem Vornamen und einem Nachnamen, auf Deutsch auch *Dateiname* und *Dateierweiterung*. Der Vorname kann aus höchstens acht Zeichen bestehen, der Nachname aus bis zu drei. Meistens handelt es sich um Buchstaben; aber auch andere Zeichen können benutzt werden. Vorname und Nachname werden durch einen Punkt getrennt.

Hier folgen die Namen von drei Dateien, die sich vermutlich auf deiner Festplatte befinden. Leider sind das Namen, die selbst einem Reiseführer Sprachprobleme geben können:

```
AUTOEXEC . BAT
CONFIG . SYS
COMMAND . COM
```

Auch wenn der Vorname nicht acht Buchstaben beansprucht, wird der Punkt ohne Zwischenraum zwischen Vor- und Nachnamen plaziert.

Der Nachname deutet in der Regel an, um welche Art von Datei es sich handelt. Siehe Seite 24.



WARNUNG

Laß die Finger von diesen drei Dateien, bis du mehr über sie weißt (Seite 31 und 46).

Verzeichnisse

Auf einer Festplatte können mehrere tausend Dateien mit unterschiedlichem Inhalt gelagert sein. Liegen sie in wildem Durcheinander, wird es schwer sein, jeweils die rechte zu finden.

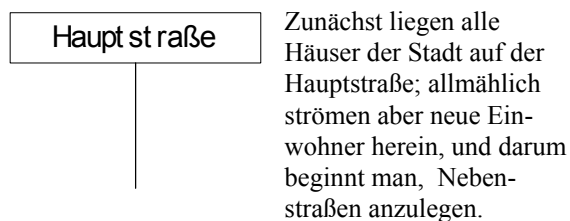
Darum wird die Festplatte in kleinere Abschnitte mit jeweils eigenen Dateien aufgeteilt. Es geht darum, die Dateien auseinanderzuhalten, so daß nur miteinander verwandte Dateien beieinander liegen. Ein solcher Abschnitt wird ein Verzeichnis genannt – auf Englisch Directory.

Der Name hängt damit zusammen, daß man sich eine Übersicht über die Dateinamen jedes Verzeichnisses verschaffen kann – ähnlich wie man in einem Adressverzeichnis nachschlagen kann, wer in der Strandallee wohnt. Ein Verzeichnis ist die Heimat einer Gruppe von Dateien.

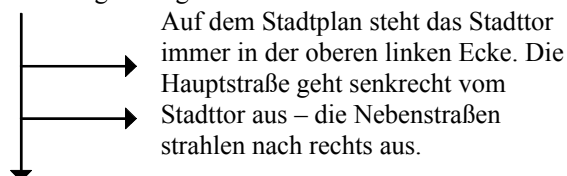
Anfänglich besteht die Festplatte nur aus einem Verzeichnis, dem Stammverzeichnis. Allmählich füllt sich die Platte mit vielen Dateien; du kannst mit Hilfe von DOS neue Verzeichnisse für sie erstellen.

DOS erstellt neue Verzeichnisse durch Verzweigungen. Das ist praktisch, aber nicht ganz einfach zu verstehen. Also heißt es nun genau zuhören.

Stell dir die Festplatte als offene Landschaft vor, in der du eine Stadt anlegst. Das einzige, was gegeben ist, ist das Stadttor, das direkt zur Hauptstraße der Stadt führt.



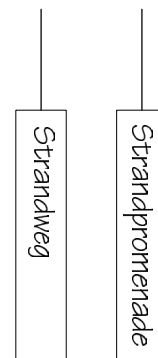
Die Stadt ist recht eigenartig – sie kann nur in zwei Richtungen ausgebaut werden.



Auf dem Stadtplan sieht eine Nebenstraße so aus:



Nehmen wir nun an, daß die Strandallee später zwei kleinere Nebenstraßen erhält. Werden sie senkrecht an die Strandallee angelegt, könnte das so aussehen:



Im Technischen Amt hat man nun aber beschlossen, daß alle Stadtpläne *ohne senkrecht stehenden Text* abgefaßt werden sollen. Das soll die Erstellung der Pläne mit Hilfe einer Textverarbeitung erleichtern – dafür wird es den Benutzern schwerer fallen, sich in den Straßen zu orientieren.

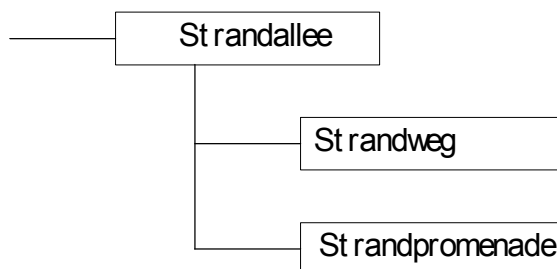
Auf der Karte werden die Striche, die zu den kleineren Nebenstraßen führen, abgelenkt:



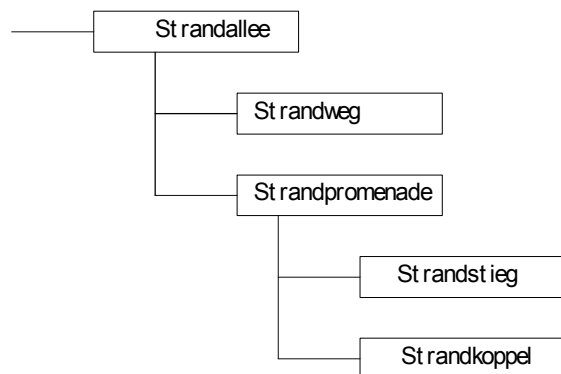
..und..



..so daß die Nebenstraßen der Strandallee *waagrecht* erscheinen können. Also:



Hat die Strandpromenade zum Beispiel zwei Nebenstraßen, so wird der Abzweig folgendermaßen aussehen:



Hier ist zu beachten, daß die Felder der Straßennamen jeweils nach rechts rücken, wenn eine neue Verzweigung entsteht. Je weiter rechts du dich auf der Karte befindest, desto mehr Straßen mußt du passieren, um zur Hauptstraße zurückzukehren.

Dieser schematisierte Stadtplan entspricht so ziemlich der Weise, in der DOS Verzeichnisse aufzeichnet.

Entsprechend ist die Stadt, die durch Verzweigung der Wege ausgebaut wird, ein gutes Bild für die Erstellung von Verzeichnissen.

Anfänglich gibt es nur eine Hauptstraße, die dem Stammverzeichnis entspricht. Unterabschnitte könnten als Verzeichnisse erstellt werden, die den Nebenstraßen entsprechen, die weitere Nebenstraßen (also Verzeichnisse) haben.

Das Wort "Nebenstraße" besagt, daß eine Straße von einer anderen, größeren ausgeht. In DOS wird das Wort Unterverzeichnis für den Sachverhalt verwendet, daß ein Verzeichnis Unterabteilung eines anderen Verzeichnisses ist. Ein Unterverzeichnis ist ebenfalls ein Verzeichnis; die ersten Silben verraten, daß es einem anderen Verzeichnis untergeordnet ist.

Die Dateien liegen in Verzeichnissen wie die Häuser an den Straßen der Stadt. Was Dateien in einem Verzeichnis miteinander verknüpft, ist ihre gemeinsame Adresse. Diese ist eine genaue Angabe ihrer Plazierung auf der Festplatte. Will DOS eine bestimmte Datei finden, muß es ihre Adresse kennen.

Stell dir nun Herrn DOS als Tourist in unserer Stadt vor. Er hat viele Dinge zu erledigen und bewegt sich unbeschwert durch die Straßen.

Plötzlich klingelt sein Mobiltelefon: er soll ein bestimmtes Haus namens Bella Vista aufsuchen. Das ist die einzige Angabe, die er erhält; also sieht er sich die Häuser der Straße an, auf der er sich befindet. Als es sich zeigt, daß es hier kein Haus namens Bella Vista gibt, nimmt er resolut das Telefon und teilt mit, daß das Haus nicht aufzutreiben ist – und damit Basta.

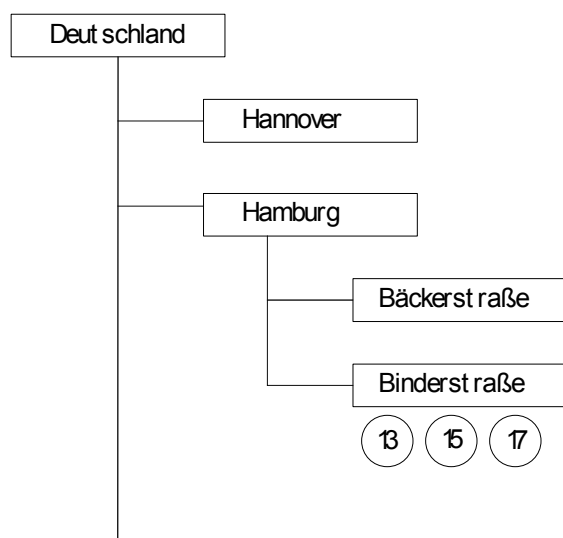
Hätte er die richtige Adresse erhalten, sähe die Sache ganz anders aus. Was er benötigt, ist eine genaue Beschreibung des Weges, der zur Straße führt, an der Bella Vista liegt. Wäre Herr DOS ein Taxifahrer, hätte ihm der Straßename vermutlich gereicht – er hätte selber den schnellsten Weg gefunden. Als Tourist zieht er aber eine idiotensichere Anweisung des genauen Weges vor.

Der nächste Bescheid übers Telefon hat mehr Erfolg. Nun erhält er die genaue Wegbeschreibung vom Stadttor bis zur richtigen Straße. Bei seinem angeborenen Orientierungsvermögen bereitet es ihm keine Schwierigkeit, sich den Weg zurück zur Hauptstraße einzuprägen; nun geht alles wie geschmiert. Am Ende der beschriebenen Route liegt der Weg mit dem stolzen Haus Bella Vista.

Das Bild des leichtfüßigen Herrn DOS auf seiner ewigen Suche durch die Winkel der Festplatte zeigt die Bedeutung der Wege, der sogenannten **Pfade** von deinem Stammverzeichnis zu den Dateien.

Das Bild der Stadt mag aber fehlerhaft erscheinen – ein Briefträger würde sich weigern, so zu arbeiten. Wer schriebe wohl auch Adressen in dieser Weise?

Sehen wir uns also ein Beispiel an, das für einen Briefträger ein Genuß wäre. Wir vergleichen die Festplatte mit dem ganzen Land. Die ersten Unterverzeichnisse sind Städte, die wiederum in Straßen unterteilt sind.



Hier entspricht der Pfad mehr einer Adresse, die du auf einen Brief schreiben würdest – nur nicht ganz in dieser Reihenfolge.

In diesem Bild entsprechen die Dateien den Häusern, die an den angegebenen Nummern liegen.

Verzeichnisse sind ein Grundbegriff für das Verständnis der jeweiligen Platzierung von Dateien. Um dies zu verdeutlichen, wollen wir einige DOS-Befehle ausprobieren, in denen Verzeichnisse eine entscheidende Rolle spielen.

Bei einigen der folgenden Befehle wirst du sehen, daß eine Menge Text schnell über den Bildschirm läuft.

Willst du das stoppen, drückst du die Pausetaste oben rechts auf der Tastatur. Drückst du anschließend auf eine andere Taste, rollt der Text weiter.

Tree

Möchtest du sehen, wie deine Festplatte in Verzeichnisse aufgeteilt ist, schreibst du folgenden Befehl:

```
C:\>TREE
```

Du siehst nun auf dem Bildschirm eine schematische Auflistung deiner Verzeichnispfade, also der Zweige, die deine Verzeichnisse verbinden, zum Beispiel:

```
C: .
├── DOS
```

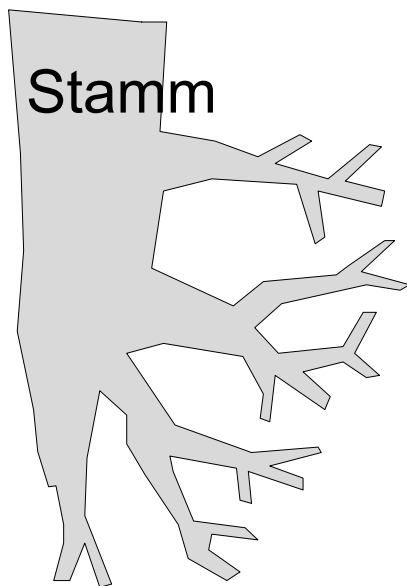
Diese recht simple Aufteilung gilt für eine Festplatte, auf der DOS eben installiert wurde.

Hier gibt es nur ein Hauptverzeichnis C:\ und ein Unterverzeichnis DOS mit speziellen DOS-Programmen.

Wenn später weitere Dateien auf der Platte gelagert wurden, sieht die Unterteilung vielleicht folgendermaßen aus:

```
C: .
├── BAT
├── DATEN
│   ├── BRIEFE
│   ├── GRAPHIK
│   ├── NOTIZEN
│   └── KALKULUS
│       ├── BUDGET
│       └── MWS
├── DOS
├── SPIELE
│   ├── EURO
│   ├── FRAC
│   ├── KEEN1
│   └── YATZY
├── TMP
└── UTIL
    ├── BAN
    └── DCF
```

Diese Auflistung erinnert mit ihren vielen Verzweigungen an einen Baum; zwar einen etwas schematischen Baum mit der Wurzel am oberen (!) Ende und dem Stamm an der linken Seite. Das erklärt den Namen des Befehls: TREE.



Das oberste Hauptverzeichnis wird auch Stammverzeichnis genannt

Hast du dir nun die Plazierung deiner Verzeichnisse mit Hilfe des Befehls `TREE` angesehen, machst du dem Bildschirm wieder frei mit dem Befehl `CLS`.

Schaffe dir nun einen Überblick über den Inhalt deines Hauptverzeichnisses, indem du schreibst:

```
C:\>DIR
```

Es erscheint eine Übersicht, die ungefähr so aussieht:

```
Datenträger in Laufwerk C: ist
STEEN
Verzeichnis von C:\
DOS          <DIR>
AUTOEXEC BAT      503
COMMAND COM    54.619
CONFIG  SYS      504

4 Datei(en)   55,626 Byte
93,528,064 Byte frei
```

Erstellst du über `DIR` einen Überblick, so siehst du rechts zwei Spalten mit Datum und Uhrzeit. Hier wird der Zeitpunkt der letzten Änderung für die jeweilige Datei angegeben. Diese Spalten habe ich um der Übersichtlichkeit willen nicht in meine Beispiele aufgenommen.

Im übrigen gibt dir eine solche Übersicht etliche nützliche Informationen. Ganz oben siehst du, daß die Festplatte den Namen `STEEN` trägt. Dies ist ihr Volume oder Label – ein Name, den du frei wählen kannst.

Hierauf folgt der Name eines Unterverzeichnisses, das mit der Bezeichnung `<DIR>` angezeigt wird.

Darunter siehst du Namen und Größe von drei Dateien, die im allgemeinen im Hauptverzeichnis liegen.

Die Größe deiner Dateien und Platten wird in Bytes gemessen. 1 Byte entspricht einem Zeichen.

1 Kilobyte (KB oder K) besteht aus 1.024 Bytes. Daß die Zahl 1.024 verwendet wird statt 1.000, liegt an der Vorliebe der Computertechniker für das Zweizahlen- oder binäre System.

1.024 ist die zehnte Potenz von 2.

1 MegaByte (MB) sind 1.024 KB.

Das entspricht $1.024 \times 1.024 =$

1.048.576 Bytes.

Weiter unten wird angegeben, daß es insgesamt vier Dateien gibt. Das liegt daran, daß das Unterverzeichnis `DOS` auch als Datei gerechnet wird. Ganz unten siehst du, wie viele Bytes an freiem Platz sich auf deiner Platte befinden.

Es mag etwas verwirren, daß ein Unterverzeichnis als Datei in der Übersicht angegeben wird.

Aber es IST eine Datei !

Die einzige, dafür aber wichtige, Funktion dieser Datei ist, darauf aufmerksam zu machen, daß es ein Unterverzeichnis namens `DOS` gibt.

Die meisten Dateien haben sowohl Vor- als auch Nachnamen, während Verzeichnisse in der Regel nur einen Vornamen haben.

Schreibst du:

```
C:\>DIR...
```

erhältst du eine Übersicht, die nur Dateien ohne Nachnamen enthält, das heißt, vor allem Unterverzeichnisse.

Die Platte wechseln

Bisher sah der Prompt so aus:

```
C:\>
```

Das bedeutet, daß die von dir geschriebenen Befehle sich auf das Stammverzeichnis deiner Festplatte beziehen. DOS kann nun aber seinen Ort wechseln und dorthin gehen, wo du etwas erledigen willst. Das kann ein anderes Verzeichnis oder eine andere Platte sein.

DOS faßt eine Platte als Definition eines eigenen *Tree*, eines Verzeichnisbaums, mit eigenen Verzeichnissen auf. Eine Festplatte funktioniert normalerweise als ein einziges Laufwerk – sie läßt sich aber ohne weiteres in kleinere Laufwerke unterteilen.

Der erste Buchstabe des Prompt gibt an, auf welchem Laufwerk DOS zur Zeit arbeitet. Ein Laufwerk wird mit einem Buchstaben mit nachfolgendem Doppelpunkt bezeichnet. Die Namen **A:** und **B:** werden für Diskettenlaufwerke verwendet – manchmal kann eine Maschine nämlich zwei von ihnen enthalten. Das eingebaute Laufwerk, die Festplatte, heißt im allgemeinen **C:**

Die Zeichen nach der Laufwerksbezeichnung geben an, in welchen Verzeichnis des jeweiligen Laufwerks DOS deinen Befehl ausführen soll. Der umgekehrte Schrägstrich, auf Englisch ein *Backslash*, nach dem **C:** wird als Trennzeichen vor und nach jedem Verzeichnis verwendet. Steht vor dem Schlußzeichen nur **C:**, bedeutet das, daß du im Stammverzeichnis der Festplatte arbeitest.

Wechsle nun die Platte von **C:** nach **A:**. Bevor du DOS aufforderst, sich für die A-Platte zu interessieren, mußt du eine Diskette ins Laufwerk stecken – du solltest ein Klicken hören.

Nun schreibst du:

```
C:\>A:
```

Der Prompt wird zu

```
A:\>
```

Willst du zur C-Platte zurück, schreibst du:

```
A:\>C:
```

Das Verzeichnis wechseln

Auf deiner Festplatte kannst du aus dem Hauptverzeichnis **C:** den Ausgangspunkt für deine Befehle ändern, indem du in ein anderes Verzeich-

nis überwechselst. Das tust du mit dem Befehl *Change Directory*, abgekürzt **CD**.

Gehst du ein Stockwerk nach unten in ein Unterverzeichnis, mußt du seinen Namen angeben.

Du schreibst beispielsweise:

```
C:\>CD DOS
```

Der Prompt gibt sogleich zu verstehen, daß du dich im neuen Verzeichnis befindest:

```
C:\DOS>
```

Hier kannst du dir seinen aktuellen Inhalt ansehen, indem du **DIR** schreibst:

```
C:\DOS>DIR
```

Auf meiner Festplatte sieht die Übersicht so aus:

```
Datenträger im Laufwerk C: ist
STEEN
Verzeichnis von C:\DOS
.                <DIR>
..               <DIR>
MONOUMB  386          8.783
VFINTD   386          5.295
DBLSPACE BIN        64.246
CHOICE   COM          1.754
COMMAND  COM          54.619
DISKCOMP COM        10.748
DISKCOPY COM        13.335
DOSKEY   COM          5.861
EDIT     COM          413
```

... (und so weiter)

```
133 Datei(en)
4.834.491 Byte
68.542.464 Byte frei
```

Am Ende der Übersicht liest du, daß es in **C:\DOS** auf meiner Festplatte 133 Dateien gibt.

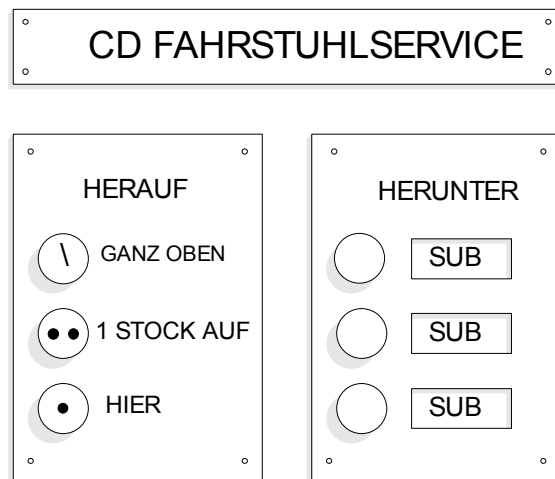
Aus Gründen der Übersichtlichkeit zeige ich nur die ersten neun.

Im Gegensatz zur vorigen Übersicht haben die hier durch **<DIR>** angezeigten Verzeichnisse keine Namen, sondern zeigen sich als Punkte. Das liegt daran, daß es in meinem **C:\DOS** noch keine Unterverzeichnisse gibt.

Das erste Verzeichnis (mit *einem* Punkt) ist das aktuelle Verzeichnis, in dem wir uns befinden, nämlich `C:\DOS`

Das nächste Verzeichnis (mit zwei Punkten) ist die Bezeichnung für das darüber liegende Verzeichnis – ein Stockwerk nach oben, also `C:\`

Stell dir einen Aufzug vor, der zwischen diesen Verzeichnissen auf und ab fährt. In ihm hat DOS Knöpfe mit einem respektive zwei Punkten angebracht. Drückst du auf den Knopf mit einem Punkt, bleibt der Aufzug, wo er ist, und öffnet nur die Türe des augenblicklichen Stockwerks. Drückst du auf den Knopf mit zwei Punkten, fährt der Aufzug ins obere Stockwerk und setzt dich dort ab. Gibt es Unterverzeichnisse, kannst du einen Stock tiefer in ein Unterverzeichnis fahren, indem du den Kopf mit seinem Namen drückst.



Die Zeilen, die `<DIR>` enthalten, sagen dir also, zu welchen Türen du aus dem aktuellen Verzeichnis Zugang hast, einschließlich seiner eigenen Tür. In unserem Beispiel kannst du nicht tiefer kommen, weil es keine weiteren Unterverzeichnisse gibt; dagegen kannst du aufwärts fahren, indem du den Knopf mit zwei Punkten drückst. Das ist dasselbe wie der Befehl

```
C:\DOS>CD..
C:\>
```

Hier kommst du ins Stammverzeichnis zurück, indem du einen Stock aufwärts fährst.

In anderen Situationen befindest du dich vielleicht tief in irgendeinem Unterverzeichnis – etwa `C:\WP51\BRIEFE\PRIVAT` – und möchtest schnell ins Stammverzeichnis zurück. Für diesen Zweck hast DOS einen Expreßknopf gesetzt, der mit `\` markiert ist und dich unmittelbar ins Stammverzeichnis zurückbringt.

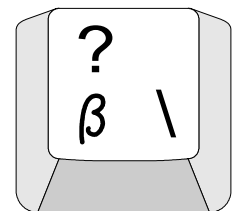
Das entspricht dem Befehl

```
C:\WP51\BRIEFE\PRIVAT>CD \
C:\>
```

Der Expreßbefehl `CD \` ist eine Abkürzung für `CD C:\` – nur läßt du die Laufwerksbezeichnung der Festplatte `C:` aus und bezeichnest den Stamm nur mit einem umgekehrten Schrägstrich (englisch *Backslash*).

Der Zwischenraum hinter `CD` kann ausgelassen werden.

Der Backslash gehört zu den meistverwendeten Zeichen in DOS. Leider hat er auf der deutschen Tastatur keine eigene Taste, sondern erscheint als Alternativbesetzung der Taste `β`.



Du erhältst das alternative Zeichen, wenn du die Taste `Alt Gr` (Alternative Graphic) hältst, während du die jeweilige Taste drückst. `Alt Gr` findest du rechts von der Leertaste.

Stehst du in einem Verzeichnis und möchtest in ein anderes, das sich in einer anderen Verzweigung deines Verzeichnisbaums befindet, kannst du nicht einfach einen Stock nach oben oder nach unten gehen. In diesem Fall gehst du über den Stamm, um anschließend ins neue Verzeichnis einzutauchen – das machst du, indem du einen Backslash vor den Namen des neuen Verzeichnisses setzt, etwa

```
C:\BAT>CD \DOS
C:\DOS>
```

Ein Verzeichnis erstellen

Willst du ein neues Verzeichnis erstellen, verwendest du den Befehl Make Directory, abgekürzt MD. Das neue Verzeichnis erscheint als Unterverzeichnis im aktuellen Verzeichnis.

Bist du zum Beispiel im Stammverzeichnis und möchtest das neue Verzeichnis `C:\PAK` erstellen, schreibst du

```
C:\>MD PAK
```

Willst du nun unter `C:\PAK` ein neues Unterverzeichnis `ZIP` erstellen, wechselst du zunächst in `C:\PAK` über, indem du schreibst

```
C:\>CD PAK
C:\PAK>
```

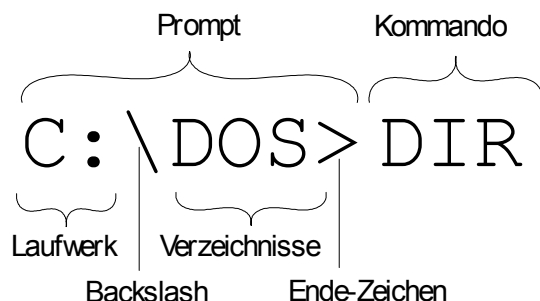
– dann erstellst du das neue Unterverzeichnis:

```
C:\PAK>MD ZIP
```

Im Verzeichnisbaum sieht das so aus:



Erstellst du ein neues Verzeichnis, entspricht das der Anlage einer neuen Straße. Häuser sind noch nicht gebaut (es gibt keine Dateien); die werden aber kommen.



Der Backslash wird als Trennzeichen zwischen Laufwerk und Verzeichnis verwendet.

Ein Verzeichnis löschen

Ein Verzeichnis wird gelöscht mit dem Befehl RD (Remove Directory) samt dem Namen des zu löschenden Verzeichnisses.

Du kannst nur ein Verzeichnis löschen, das weder Unterverzeichnisse noch Dateien enthält. Bevor du es löschst, mußt du es also leeren.

Nun willst du das leere Verzeichnis `ZIP` löschen, das ein Unterverzeichnis deines aktuellen Verzeichnisses `C:\PAK` ist. Du schreibst:

```
C:\PAK>RD ZIP
```

Nun ist `ZIP` gelöscht.

Versuchst du ein Verzeichnis zu löschen, das nicht als Unterverzeichnis des aktuellen Verzeichnisses existiert oder nicht leer ist, erscheint folgende Mitteilung:

```
Ungültiger Pfad, kein
Verzeichnis oder Verzeichnis
nicht leer
```

Das ist eine sogenannte Fehlermeldung. Sie besagt, daß du etwas geschrieben hast, das DOS nicht versteht oder akzeptiert.

Das Löschen eines Verzeichnisses entspricht in etwa der Schließung einer Straße, die nur erfolgen kann, wenn die Häuser, die Dateien, verschoben oder abgerissen, das heißt gelöscht, wurden – siehe auch Seite 37. Außerdem müssen alle Nebenstraßen dieser Straße ebenfalls geleert und geschlossen sein.

Es gibt allerdings einen Befehl, der ein Verzeichnis völlig entfernen kann, ohne daß es zuvor geleert wurde – siehe Seite 39.

DIR

Du hast bereits gesehen, daß du über den Befehl `DIR` erfahren kannst, welche Dateien in deinen Verzeichnissen liegen. Enthält dein aktuelles Verzeichnis sehr viele Dateien, füllt die gesammelte Übersicht mehr Platz, als dir dein Monitor bietet. Erstellst du zum Beispiel eine Übersicht über `C:\DOS` durch die Eingaben

```
C:\>CD DOS
C:\DOS>DIR
```

so wird diese Übersicht mit ihren zahlreichen Dateien blitzschnell vorbeigekollert und erst dann stillstehen, wenn die letzte Datei erreicht ist.

Du kannst, wie schon erwähnt, das Bild mit der Pausentaste anhalten; der Befehl `DIR` bietet aber auch eine automatische Pausenfunktion, die du aktivierst, indem du schreibst

```
C:\DOS>DIR /P
```

Nun erhältst du dieselbe Übersicht in jeweils bildschirmgroßen Portionen. Sie geht erst dann ein Bild weiter, wenn du eine Taste drückst:

Eine beliebige Taste drücken,
um fortzusetzen

`/P` steht für Pause.

Schalter

Ein Schrägstrich mit einem nachfolgenden Buchstaben in einem Befehl wird *Schalter* genannt.

Die meisten Befehle können mit einem oder mehreren Schaltern versehen werden, die seine Grundfunktion ergänzen.

Schalter werden in der Regel am Ende eines Befehls angebracht.

Wünschst du eine mehrspaltige Übersicht, kannst du schreiben

```
C:\DOS>DIR /W
```

Nun werden die Dateinamen in vier Spalten gezeigt. `/W` steht für Wide (breit).

In einer breiten Übersicht fehlen die Spalten mit der Dateigrößen und dem Datum der letzten Änderung.

Kurzübersicht

Willst du untersuchen, ob es in deinem aktuellen Verzeichnis eine Datei mit einem bestimmten Namen gibt, schreibst du `DIR` und anschließend den Namen der gesuchten Datei. Du suchst eine Datei namens `ANSI.SYS` in `C:\DOS`, indem du schreibst

```
C:\DOS>DIR ANSI.SYS
```

Nun erscheint eine Kurzübersicht, die nur die gesuchte Datei enthält:

```
Verzeichnis von C:\DOS
ANSI  SYS          9.065
1 Datei(en)      9.065 bytes
```

Wild Cards

In den meisten DOS-Befehlen, unter anderem `DIR`, kannst du ein oder mehrere Zeichen durch sogenannte WildCards ersetzen. Die üblichste WildCard ist das Stern-Zeichen `*`.

DOS faßt diesen Stern in einem Befehl als 'alle möglichen Zeichen' auf.

Willst du zum Beispiel eine Kurzübersicht über alle Dateien des aktuellen Verzeichnisses erstellen, deren Vorname mit dem Buchstaben `A` beginnt, schreibst du

```
C:\DOS>DIR A*.*
```

```
Verzeichnis von C:\DOS
APPEND  EXE      10.774
ATTRIB  EXE      11.208
ANSI    SYS       9.065
3 Datei(en) 31.047 Byte
```

Statt `A*.*` kannst du auch `A*` schreiben.

Fehlt der Punkt, nimmt DOS also an, daß du den Vornamen geschrieben hast, während der Nachname aus allen möglichen Zeichen bestehen kann.

Willst du Dateien angeben, deren Nachname mit `CO` beginnt, schreibst du `*.CO*`

Die Bezeichnung `*.*` bedeutet 'alle Dateien'.

Suche auf der gesamten Festplatte

Fügst du nach dem `DIR` ein `/S` ein, wenn du eine Kurzübersicht zu einer bestimmten Datei bestellst, kannst du DOS dazu bringen, alle Unterverzeichnisse des aktuellen Verzeichnisses nach dem betreffenden Dateinamen zu durchsuchen.

Beginnst du den Befehl `DIR /S` an der Spitze des Stammverzeichnisses, werden alle Verzeichnisse einer Platte durchsucht.

Du brauchst also nicht unbedingt zu wissen, in welchem Verzeichnis sich eine Datei befindet.

Willst du untersuchen, ob es irgendwo auf deiner Festplatte eine Datei namens `RASMUS.DOC` gibt, schreibst du

```
C:\>DIR /S RASMUS.DOC
```

und erhältst eine Kurzübersicht der Verzeichnisse, in denen DOS diese Datei findet. Dateien gleichen Namens können problemlos auf derselben Platte liegen, solange sie nicht im selben Verzeichnis stehen.

Wie du siehst, ist der Schalter `/S` vor den Namen der Datei gerückt. Ob du einen Schalter unmittelbar nach dem Namen eines Befehls schreiben willst, wie hier, oder ans Ende der Zeile, ist Geschmackssache. Beide Versionen funktionieren.

Bist du dir nicht sicher, ob die Datei `RASMUS.DOC` heißt, kannst dich aber erinnern, daß sie mit `RAS` beginnt, schreibst du

```
C:\>DIR /S RAS*
```

Es erscheint eine Kurzübersicht aller Dateien auf der Platte, die mit `RAS` beginnen.

Willst du eine gründliche Suche mit `DIR /S` durchführen, mußt du im Stammverzeichnis beginnen.

Die Befehle 'Gehe zum Stammverzeichnis' und 'Schreibe `DIR /S`' habe ich auf Seite 48 in einer sogenannten BAT-Datei kombiniert, um sie leichter merkbar zu machen.

Versteckte Dateien

Erstellst du mit `DIR` eine Übersicht über beispielsweise das Stammverzeichnis deiner Festplatte, indem du schreibst

```
C:\>DIR
```

so werden bestimmte Dateien nicht in diese Übersicht aufgenommen. Dies sind die sogenannten *versteckten Dateien*.

Sie sind versteckt, weil sie für DOS wichtig sind, aber nie in Befehlen verwendet werden. Daß sie versteckt werden, geschieht aus Sicherheitsgründen – damit keine Änderung an ihrem Inhalt oder ihrer Plazierung vorgenommen werden kann.

Willst du versteckte Dateien in deine Übersicht aufnehmen, schreibst du

```
C:\>DIR,
```

Das Komma nach `DIR` ohne Zwischenraum macht den gesamten Unterschied aus. Diese Ausgabe ist ein sogenannter undokumentierter Befehl, der im DOS-Handbuch nicht beschrieben wird.

Vergleiche nun eine Übersicht mit und eine ohne Komma. Im Stammverzeichnis liegen normalerweise mindestens zwei versteckte Dateien:

```
IO.SYS und MSDOS.SYS
```

Diese Systemdateien sind die Grundlage für die Anwendung von DOS auf deinem PC.

Rühre sie nicht an!

Namen und Pfade von Dateien

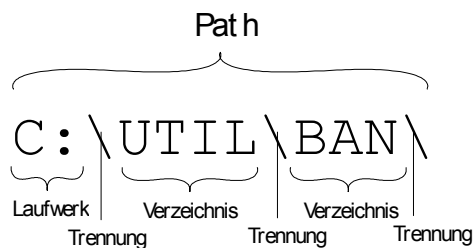
Gehört zu einem Befehl der Name einer Datei, die sich nicht im augenblicklichen Verzeichnis befindet, muß du ihren Namen und ihre Plazierung angeben. Sehen wir uns ein Beispiel an:

Auf meiner Festplatte gibt es ein Programm, das große Fahnen zeichnen kann. Alle Dateien dieses Programmes habe ich gespeichert in einem Verzeichnis namens `BAN` als Unterverzeichnis in `UTIL`, in dem eine Reihe von Utilities, von Werkzeugen, enthalten ist. Eine dieser Dateien ist ein Schrifttyp, ein sogenannter Zeichensatz, mit dem ich Texte für ein solches Banner erstellen kann. Der Name dieser Datei ist `HOLYWOOD.FNT`.

Will ich diese Datei genau bezeichnen, muß auch ihre Plazierung in meinen Verzeichnisse angegeben werden:

```
C:\UTIL\BAN\HOLYWOOD.FNT
```

Das, was vor dem Namen der Datei steht, wird ihr *Path* genannt. Dieses englische Wort bezeichnet eine Route, einen Pfad.



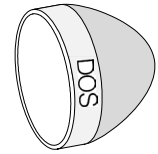
Der Path gibt die Plazierung einer Datei an

Der Path, der durch einen Backslash (\) vom Dateinamen getrennt wird, gibt den Weg an, der vom Stammverzeichnis zur Datei führt. Beginnst du im Stammverzeichnis `C:\`, müßt du zunächst ins Verzeichnis `UTIL`.

Von hier gehst du ins Unterverzeichnis `BAN`, wo du die Datei findest.

Suchpfad

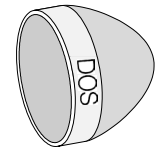
Die Mehrzahl der Befehle wird einen Dateinamen enthalten. Ist dieser Name mit einem Path versehen, wird DOS das Verzeichnis durchleuchten, zu dem der Path führt.



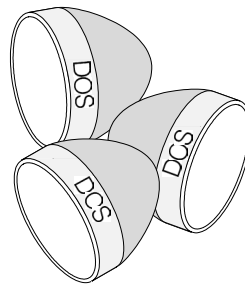
Schreibst du den Namen einer Datei ohne ihren Path, wird DOS sie nur im aktuellen Verzeichnis suchen.

Liegt die Datei nicht hier, weiß DOS nicht, wo es suchen soll, und teilt mit:

Dateiname nicht gefunden



Bestimmte Dateien werden so oft in Befehlen verwendet, daß es angenehm wäre, ihren Path nicht schreiben zu müssen – wenn man sich gerade nicht in ihrem Verzeichnis befindet.



Darum gibt es einen Befehl `PATH`, der die Suchleuchte von DOS in ausgewählten Verzeichnissen aktivieren kann. Die Dateien in diesen Verzeichnissen können jederzeit ohne Path angegeben werden, wo du dich auch befindest.

Der Suchpfad ist eine Liste von diesen ausgewählten Verzeichnissen.

In DOS 6 ist ein Verzeichnis namens `C:\DOS` von vornherein in den Suchpfad aufgenommen. Hier findest du eine Reihe viel verwendeter Programme; es wäre praktisch, diese durch Angabe ihres bloßen Namens aktivieren zu können – egal in welchem Verzeichnis man sich befindet. Und der Pfad, der zu den Dateien in `C:\DOS` führt, also der Path `C:\DOS`, versteht sich von selbst auf deinen sämtlichen Platten. Willst du sehen, welche Verzeichnisse zu deinem Suchpfad gehören, schreibst du

```
C:\>PATH
```

Nun zeigt DOS dir den augenblicklichen Suchpfad:

```
PATH=C:\;C:\DOS
```

In diesem Fall besteht er aus `C:\` und `C:\DOS`. Die einzelnen Verzeichnisse auf dem Suchpfad werden durch ein Semikolon voneinander getrennt.

Du kannst selbst den Suchpfad deiner meistverwendeten Verzeichnissen zusammenstellen. Möchtest du `C:\BAT` einbeziehen, schreibst du

```
C:\>PATH C:\;C:\DOS;C:\BAT
```

Nun besteht der Suchpfad aus den drei gezeigten Verzeichnissen. Wie du siehst, mußt du den gesamten Suchpfad schreiben, auch wenn du nur etwas hinzufügen willst. Der einzige Zwischenraum des Befehls kommt nach dem Wort `PATH`.

Sieh dir den Suchpfad noch einmal an:

```
C:\>PATH
PATH=C:\;C:\DOS;C:\BAT
```

Das Wort `PATH` kann also in einem Befehl auf zweierlei Weise verwendet werden:

- Einerseits `PATH` alleine, wodurch du den entsprechenden Suchpfad siehst.
- Andererseits `PATH` plus eine Liste der Verzeichnisse, die von nun an einbezogen werden sollen.

Die Reihenfolge der Verzeichnisse im Suchpfad ist nicht ganz einerlei. DOS nimmt seine Suche in der Reihenfolge der aufgeführten Verzeichnisse vor, bis die Datei gefunden ist. Enthalten zum Beispiel die ersten Verzeichnisse auf dem Suchpfad viele Dateien, während die gesuchte Datei erst im letzten Verzeichnis steht, wird die Suche länger dauern.

Um bei jedem Maschinenstart ein und denselben Suchpfad zu sichern, kann der `PATH`-Befehl in die `AUTOEXEC.BAT` (siehe Seite 46) aufgenommen werden.

Dateitypen

Der Name einer Datei läßt in der Regel erkennen, um welchen Dateityp es sich handelt. Es gibt bestimmte Typen von Dateien mit verschiedenen Eigenschaften.

In `C:\DOS` befindet sich zum Beispiel die Datei `PRINT.EXE`.

Der *Vorname* (oder Dateiname) sagt in der Regel etwas über die spezifische Datei aus – was sie enthält oder tun kann. Der Vorname `PRINT` sagt deutlich, daß es sich um etwas aufs Drucken bezogenes handelt.

Leider läßt sich nicht immer so leicht durchschauen, was ein Vorname eigentlich andeutet.

Der *Nachname* (oder die Dateierweiterung) gibt den *Typ* einer Datei an – ob sie ein Programm, ein Dokument oder etwas anderes ist.

Es folgen einige der üblichsten Dateitypen:

.COM	Programme durchschnittlicher Größe.
.EXE	große Programme
.BAT	BAT-Dateien, Miniprogramme.
.TXT	Textdateien – Informationen für Benutzer
.DOC	nicht ganz so eindeutig – dieser Nachname gibt Dokumente von Microsofts Textverarbeitung Word, aber auch Textdateien mit Gebrauchsanweisungen für Programme an.

Dateien mit den Nachnamen `COM`, `EXE` oder `BAT` sind Programmdateien, die bestimmte Aufgaben ausführen, wenn sie aktiviert werden.

`COM`- und `EXE`-Dateien sind die meistverbreiteten Programmdateien. Die kleinsten (`COM`) haben den Charakter von Befehlen mit einfachen Funktionen; die großen (`EXE`) können komplizierte Aufgaben durchführen.

`BAT`-Dateien (`BAT`) sind Miniprogramme in Form von Textdateien, die vom Benutzer erstellt werden (siehe Seite 48).

`COM`-, `EXE`- und `BAT`-Dateien zeichnen sich dadurch aus, daß sie aktiviert werden, wenn du ihren *Vornamen* schreibst.

Willst du etwa `PRINT.EXE` aktivieren, schreibst du einfach `PRINT` nach dem Prompt.

Die Datei `AUTOEXEC.BAT` läßt sich aktivieren, indem du `AUTOEXEC` schreibst; außerdem hat sie die Besonderheit, daß **sie unmittelbar nach jedem Maschinenstart automatisch aktiviert wird.**

AUTOMATIC **EXE**CUTION = automatische Ausführung.

Auf deiner eigenen Festplatte werden dir Dateien mit speziellen Nachnamen begegnen, die bei der Anwendung bestimmter Programme entstehen. Der Nachname wird so eine Art Warenzeichen für ein bestimmtes Programm. Ein Beispiel ist das Programm `Works`, dessen Dokumentnamen mit `.WKS` enden.

Oft werden dir aber Nachnamen begegnen, die dir nichts sagen, weil nicht zu erkennen ist, welches Programm gerade diese Bezeichnung anwendet.

Den Inhalt einer Datei ansehen

Besteht eine Datei aus Text, kannst du sie unmittelbar lesen. Ihr Inhalt wird durch den Befehl `TYPE` und den Namen der Datei auf den Bildschirm gesetzt. Willst du den Inhalt der `AUTOEXEC.BAT` sehen, schreibst du

```
C:\>TYPE AUTOEXEC.BAT
```

Normalerweise kann diese Datei problemlos auf dem Bildschirm Platz finden. Willst du dir dagegen die Datei `C:\DOS\README.TXT` ansehen, so füllt sie wesentlich mehr als ein Schirmbild:

```
C:\DOS>TYPE README.TXT
```

Dieser Text wird in rasendem Tempo über den Bildschirm rollen und erst zur Ruhe kommen, wenn das letzte Schirmbild erreicht ist. Du kannst den Ablauf ruckweise stoppen, indem du wechselweise die Pausetaste und eine andere Taste drückst – aber das ist eine unbequeme Leseweise für einen größeren Text.

Willst du eine Textdatei Seite für Seite durchlesen, kannst du den Befehl `TYPE` mit einem anderen Befehl namens `MORE` verwenden. Du schreibst

```
C:\DOS>TYPE README.TXT | MORE
```

Das sonderbare Zeichen vor `MORE` ist ein Spezialzeichen, das die beiden Befehle trennt. Es gibt nur wenige Befehle, die derart nach einem anderen Befehl gesetzt werden können – `MORE` ist einer von ihnen.

Dieses Zeichen erscheint, wenn du die **Alt Gr** hältst, während du die Taste links vom Buchstaben **Y** drückst.

Du kannst das Zeichen auch setzen, indem du die Alt-Taste und den ASCII-Wert des Zeichens eingibst, der 124 ist. Du hältst die Alt-Taste, während du auf den Zahlentasten rechts auf der Tastatur **124** tastest. **Alt** ist eine Abkürzung für alternativ (wie bei **Alt Gr**) und gibt bestimmten Tasten eine neue Bedeutung.

Weiteres zu ASCII-Werten findest du auf Seite 45.

Mit dem Befehl `TYPE | MORE` kannst du den Text schirmbildweise durchblättern. Jedesmal, wenn du eine Taste drückst, kommt ein neues Schirmbild.

Ich halte es für ziemlich idiotisch, daß etwas so banales wie das Lesen einer Textdatei einen so komplizierten Befehl erfordert.

`TYPE` sollte natürlich über einen Schalter `/P` verfügen, wie `DIR` – aber das ist leider nun einmal nicht so.

Später (Seite 49) gebe ich ein Beispiel, wie sich das Problem mit etwas Vorarbeit lösen läßt.

Der Befehl `TYPE` ist für Textdateien berechnet, kann aber auch verwendet werden, um den Inhalt der Kodierungen bestimmter Programme zu zeigen. Programmdateien mit den Nachnamen `COM` und `EXE` bestehen aus Zeichen, die auf dem Schirm mehr oder weniger als Kauderwelsch erscheinen.

Die meisten Zeichen werden im allgemeinen unverständliche Codes sein, die gar nicht als Text gelesen werden sollten. Sie enthalten Anweisungen für den Computer nach Art eines Strickmusters, das ja auch nicht gerade Bettlektüre für jedermann ist.

Ab und zu tauchen allerdings lesbare Brocken mit Copyright, Fehlermeldungen und ähnlichem auf.

Sieh dir nun den Inhalt verschiedener Dateien auf deiner Festplatte durch den Befehl `TYPE` an.

Handelt es sich um eine große Datei, wird es einige Zeit dauern, ihren Inhalt zu erfassen. Du kannst stoppen, indem du `Strg+C` drückst. Du hältst die Taste namens `Strg`, während du auf `C` drückst. `Strg` ist eine Abkürzung für Steuerung – eine Art Kontrolltaste.

Gibt dein Computer sonderbare Piepser von sich, während du Programmdateien ansiehst, ist das kein Grund, nervös zu werden. Das liegt an einem besonderen Zeichen, das dem eingebauten Lautsprecher der Maschine Bescheid zum Piepsen gibt.

Eine Datei drucken

Hast du einen Drucker an deinen PC angeschlossen, kannst du durch den Befehl `PRINT` gefolgt vom Dateinamen eine Datei drucken.

Willst du etwa die `AUTOEXEC.BAT` drucken, sorgst du dafür, daß der Drucker angeschlossen und eingeschaltet ist, und schreibst

```
C:\>PRINT AUTOEXEC.BAT
```

Wenn du den Befehl `PRINT` zum erstenmal anwendest, fragt DOS, wo der Drucker angeschlossen ist:

```
Name des Ausgabegeräts [PRN]:
```

`PRN` ist DOS' Bezeichnung für die normale Verbindung zum Drucker. Du kannst entweder `PRN` schreiben oder einfach Eingabe drücken, da DOS selbst diese Möglichkeit in viereckigen Klammern vorschlägt. Nun erscheint folgende Mitteilung auf dem Bildschirm:

```
Residenter Teil von PRINT geladen
C:\AUTOEXEC.BAT wird gerade
gedruckt
```

– und der Drucker beginnt seine Arbeit.

Taucht der Prompt wieder auf, während der Drucker arbeitet, kannst du während des Druckvorgangs weitere Befehle eingeben. `PRINT` arbeitet nämlich 'im Hintergrund' - DOS kann also ohne weiteres gleichzeitig andere Dinge erledigen.

Das Schirmbild drucken

Arbeitest du in DOS und möchtest dein augenblickliches Schirmbild ausdrucken, kannst du die Taste `Print Scrn` drücken, während der Drucker eingeschaltet ist. Diese Taste sitzt rechts von `F12` am oberen Rand der Tastatur. Diese Art, das Schirmbild einzufangen, nennt man einen *Screen Dump* oder eine *Hardcopy*.

Kontinuierlicher Druck

Experimentierst du mit verschiedenen Befehlen, kann ein kontinuierlicher Druck der Vorgänge auf dem Bildschirm nützlich sein. Ein sogenanntes *Printer Echo* startest du, indem du die Kombination `Strg+PrintScreen` drückst und weiterarbeitest, bis du erneut diese Kombination tastest. Probiere auch das aus.

Kopie einer Datei

Eine der wichtigsten Funktionen von DOS ist die Fähigkeit, Dateien zu kopieren. Der offensichtlichste Zweck des Kopierens ist, Dateien zwischen der Festplatte und einer Diskette zu transportieren. Werden Programme installiert, werden sie von einer Diskette auf die Festplatte kopiert. Wenn die Daten der Festplatte vor dem Löschen geschützt werden sollen, erstellt man eine Sicherheitskopie in der umgekehrten Richtung. Aber auch das Kopieren von Dateien zwischen verschiedenen Verzeichnissen einer Festplatte kann nützlich sein. Soll eine Datei aus einem Verzeichnis in ein anderes überführt werden, wird sie zunächst an den neuen Ort kopiert, um dann an ihrem ursprünglichen Platz gelöscht zu werden. Schließlich spielt das Kopieren eine zentrale Rolle, wenn der Inhalt von Dateien geändert werden soll. Der neue Inhalt ersetzt den alten, indem er auf ihn aufkopiert wird. Man sagt, daß der neue Inhalt den alten überschreibt.

Du kopierst eine Datei mit dem Befehl `COPY` – im allgemeinen folgen Path und Name der Originaldatei sowie der Kopie.

Das Original ist die Datei, aus der du kopierst – die Kopie ist die neue Datei, die du erstellst.

`COPY`

Path	Datei
------	-------

Path	Datei
------	-------

Original Kopie

Erfolgt das Kopieren innerhalb des aktuellen Verzeichnisses, kannst du den Pathnamen auslassen.

`COPY`

Datei

Datei

Original Kopie

Willst du zum Beispiel eine Kopie der Datei `AUTOEXEC.BAT` im aktuellen Verzeichnis erstellen, in diesem Fall im Stamm, und die Kopie als `AUTOEXEC.BAK` ins gleiche Verzeichnis setzen, reicht es, wenn du schreibst

```
C:\>COPY AUTOEXEC.BAT *.BAK
1 Datei(en) kopiert
```

Nun ist die `AUTOEXEC.BAT` als `AUTOEXEC.BAK` im gleichen Verzeichnis kopiert.

Willst du die `AUTOEXEC.BAT` in ein anderes Verzeichnis kopieren, brauchst du ihr keinen neuen Namen zu geben, da sich Dateien gleichen Namens in verschiedenen Verzeichnissen befinden können. Du mußt aber einen neuen Path als Heimat der Kopie angeben, der sich vom aktuellen Verzeichnis unterscheidet.

`COPY`

Datei

Path

Original Kopie

Läßt du den Dateinamen auf der Kopie aus, geht DOS davon aus, daß ihr Name im neuen Verzeichnis der gleiche sein soll wie im Original.

```
C:\>COPY AUTOEXEC.BAT C:\DOS
1 Datei(en) kopiert
```

Eine andere Möglichkeit ist, `C:\DOS` als aktuelles Verzeichnis zu wählen und hierhin zu kopieren:

```
C:\DOS>COPY C:\AUTOEXEC.BAT
1 Datei(en) kopiert
```

Da du hier kein Verzeichnis für deine Kopie angegeben hast, geht DOS davon aus, daß die Originaldatei ins aktuelle Verzeichnis kopiert werden soll:

`COPY`

Path	Datei
------	-------

--	--

Original Kopie

In den letzten beiden Beispielen führte das Kopieren dazu, daß es nun zwei identische Dateien namens `AUTOEXEC.BAT` in `C:\` und in `C:\DOS` gibt.

Wenn du kopierst, gibt DOS Bescheid, wie viele Dateien kopiert wurden. In den folgenden Beispielen werde ich diese Mitteilung nicht jedesmal zeigen.

Empfangsmethode

Alle gezeigten Verwendungsweisen für `COPY` haben ihre unterschiedslose Berechtigung. Die nachfolgend gezeigte Methode sollte man sich aber merken:

- Wechsle das Verzeichnis, bis du dort stehst, wo deine Kopie landen soll – also im Empfangsverzeichnis.
- Nun schreibst du `COPY` und Path und Namen der Originaldatei.

Das hat den Vorteil, daß du dich beim Befehl nur auf den Path des Originals konzentrieren mußt, also das Verzeichnis, aus dem du kopierst.

Außerdem befindest du dich nach dem Kopieren dort, wo auch die Kopie ist – also im Empfangsverzeichnis – und kannst dich beispielsweise vergewissern, daß die Kopie gut im aktuellen Verzeichnis angekommen ist, indem du `DIR` schreibst.

Überschreiben

Wie du siehst, teilt `COPY` netterweise immer mit, wie viele Dateien kopiert wurden.

`COPY` hat allerdings auch eine weniger nette Eigenschaft: besteht bereits eine Datei mit dem Namen der Originaldatei, so wird sie durch die neue Kopie ersetzt. Ohne weitere Warnung! Erst seit MS-DOS 6.2 ist eine solche Warnung eingebaut.

Man sagt, daß die ursprüngliche Datei von der neuen *überschrieben* wird.

Gab es zum Beispiel in `C:\DOS` bereits eine Datei namens `AUTOEXEC.BAT`, vielleicht mit anderem Inhalt, so würde diese Datei ohne Warnung von der neuen Kopie überschrieben. Du solltest grundsätzlich untersuchen, ob bereits eine Datei mit dem Namen des Originals existiert. Noch ein Grund, die Empfangsmethode zu verwenden.

Kopie mehrerer Dateien

`COPY` kann auch mehrere Dateien im gleichen Befehlszug kopieren. Das läßt sich machen, indem man WildCards in den Namen der Dateien verwendet.

Die Bezeichnung `*.*` bedeutet: ‘Alle Dateien’ (Dateien mit jedem beliebigen Vor- und Nachnamen).

Die Bezeichnung `*.COM` bedeutet: ‘Alle Dateien mit dem Nachnamen `COM`’.

Willst du zum Beispiel alle Dateien mit dem Nachnamen `COM` aus `C:\DOS` in das A-Laufwerk kopieren, schreibst du

```
C:\>A:
A:\>COPY C:\DOS\*.*COM
```

Da du hier keine Namen für deine Kopien angibst, erhalten sie den gleichen wie ihr Original.

Willst du nur Dateien des Verzeichnisses `C:\DOS` mit dem Vornamen `HELP` ins A-Laufwerk kopieren, schreibst du

```
A:\>COPY C:\DOS\HELP.*
```

Die WildCard `*` läßt sich in den meisten DOS-Befehlen verwenden.

Der Fingerabdruck von DOS

Wenn du das erste Mal in ein Laufwerk überwechselst, landest du in seinem Stammverzeichnis.

Wechselst du das nächste Mal in dieses Laufwerk, erinnert sich DOS, in welchem Verzeichnis du das letzte Mal warst. Also landest du dort.

DOS setzt sozusagen seinen Fingerabdruck auf jedem Laufwerk im zuletzt besuchten Verzeichnis.

DOS faßt also jeden Hinweis auf ein Laufwerk als einen Hinweis auf das zuletzt besuchte Verzeichnis auf.

Dank dieses Fingerabdrucks kannst du den Path für Dateien, die auf einem anderen Laufwerk stehen, weglassen.

Nehmen wir an, du legst eine Diskette ins A-Laufwerk, um dir die Dateien in `A:\TOOLS` anzusehen:

```
C:\>A:
A:\>CD TOOLS
A:\TOOLS>DIR
```

Nachdem du die Übersicht gesehen hast, willst du nun alle Dateien aus `A:\TOOLS` auf die Festplatte kopieren. Du wechselst das Laufwerk auf C:

```
A:\TOOLS>C:
C:\>
```

Hier erstellst du ein neues Verzeichnis, zum Beispiel `C:\TOOLS`, und wechselst darauf über:

```
C:\>MD TOOLS
C:\>CD TOOLS
C:\TOOLS>
```

Nun kannst du die Dateien aus `A:\TOOLS` in das aktuelle Verzeichnis `C:\TOOLS` setzen – du schreibst schlicht

```
C:\TOOLS>COPY A:*.*
```

DOS hat nämlich seinen Fingerabdruck auf `A:\TOOLS` gesetzt und faßt `COPY A:*.*` als einen Bescheid auf, alle Dateien dieses Verzeichnisses zu kopieren.

Du kannst sogar die WildCards auslassen und einfach schreiben

```
C:\TOOLS>COPY A:.
```

Eine Datei umbenennen

Manchmal wirst du eine Datei umbenennen wollen. Der Befehl hierzu heißt `RENAME`. Dieser Befehl wird recht häufig angewendet; darum akzeptiert DOS die Abkürzung `REN`. Du mußt den alten sowie den neuen Namen der Datei angeben.

Nehmen wir an, du hast eine Notiz namens `MEMO.TXT` geschrieben, die im Stammverzeichnis liegt. Willst du diesen Namen in `REFERAT.TXT` ändern, schreibst du

```
C:\>REN MEMO.TXT REFERAT.TXT
```

Also `REN` und danach den alten sowie den neuen Namen.

Gibt es bereits eine Datei mit dem neuen Namen oder existiert der alte Name nicht, wird DOS antworten:

```
Doppelt vorhandener Dateiname oder
Datei nicht gefunden
```

Doppelt vorhandener Dateiname bedeutet, daß du versuchst, zwei gleichnamige Dateien zu erstellen. Das wird zum Glück nicht akzeptiert.

Sei vorsichtig beim Umbenennen von Dateien. Viele Programme rechnen damit, Dateien bestimmten Namens zu finden. Dokumente müssen oft ihre eigene Dateierweiterung behalten, damit das Programm, das sie herstellte, sie finden kann.

Du kannst eine Datei nicht von einem Verzeichnis in ein anderes verschieben, indem du sie umbenennst. Hierzu mußt du den anschließend beschriebenen Befehl `MOVE` verwenden.

Eine Datei verschieben

Manchmal speicherst du eine Datei, etwa einen Brief, in einem Verzeichnis, wo sie deiner Meinung nach nichts verloren hat. Glücklicherweise kannst du sie in ein anderes Verzeichnis verschieben. In DOS 6 kannst du eine Datei verschieben mit Hilfe des Befehls `MOVE`.

Hinter `MOVE` schreibst du den Namen der zu verschiebenden Datei und ihr neues Verzeichnis. Willst du zum Beispiel die Datei `REFERAT.TXT` aus dem Stammverzeichnis nach `C:\DATA` verschieben, schreibst du

```
C:\>MOVE REFERAT.TXT \DATA
```

Die Laufwerksbezeichnung `C:` kann im Namen des neuen Verzeichnisses ausgelassen werden, da es auf diesem Laufwerk liegt.

In DOS 5, wo es `MOVE` nicht gibt, mußt du eine Verschiebung vornehmen, indem du die Datei zunächst ins neue Verzeichnis kopierst und anschließend das Original löschst.

```
C:\>COPY REFERAT.TXT \DATA
C:\>DEL REFERAT.TXT
```

Den Prozeß abbrechen

In den meisten Programmen bedeutet eine Aktivierung der `Esc`-Taste ein Abbrechen des Prozesses. Das gilt auch für einen DOS-Befehl, den du geschrieben, aber noch nicht ausgeführt hast.

Entschließt du dich, einen Befehl rückgängig zu machen, bevor du Eingabe getastet hast, kannst du mit einem Druck auf die `Esc`-Taste oben links auf der Tastatur den Text der Befehlszeile löschen.

Ist DOS allerdings bereits dabei, einen Befehl durchzuführen, den du bereust, kannst du eine drastischere Methode zum Abbrechen verwenden, indem du `Strg+C` tastest.

Sobald du diese Kombination `Strg+C` drückst, siehst du das Zeichen `^C` auf dem Bildschirm. Das ist ein Bescheid an DOS, eine aktuelle Aufgabe so schnell wie möglich abzubrechen und zum Prompt zurückzukehren.

Ist DOS voll mit einem Teilprozeß beschäftigt, kann es einige Sekunden dauern, bis die Notbremse registriert wird; in der Regel erfolgt das aber augenblicklich.

Es lohnt sich, `Strg+C` in den Fingerspitzen zu haben, wenn du eine plötzliche Unterbrechung benötigst. Bist du zum Beispiel dabei, eine größere Anzahl an Dateien zu kopieren, und entdeckst, daß du einen verkehrten Zielort angegeben hast, kannst du durch eine schnelle Unterbrechung viel Arbeit sparen. DOS hat vielleicht vor der Unterbrechung nur einen Teil des verkehrten Befehls durchgeführt.

Du zerstörst keineswegs deine Dateien, indem du einen Befehl durch `Strg+C` unterbrichst.

Nach einer unterbrochenen Kopierung kann es zum Beispiel vorkommen, daß ein paar Kopien am verkehrten Ort liegen; sie lassen sich aber je nachdem finden (siehe Seite 22) und löschen.

Interne und externe Befehle

Wenn ein Computer bootet und DOS von der Festplatte oder einer Diskette in den Speicher gelesen wird, betrifft das zunächst nur die unbedingt notwendigen Dateien. Müßten alle DOS-Befehle von Anfang an eingelesen werden, so beanspruchte das zu viel Zeit und zuviel Speicherplatz. Neben den Systemdateien (siehe Seite 42) wird daher auch ein Programm eingelesen, das nur einen kleinen Teil aller DOS-Befehle enthält.

Dieses Programm nennt man den Kommandoprocessor; es trägt den Namen `COMMAND.COM`.

Die Befehle, die verwendungsbereit im Speicher liegen, wenn die Maschine mit dem Booten fertig ist, heißen *interne* Befehle.

`VER`, `DATE`, `TIME`, `CLS`, `REN` und `DIR` gehören zu den internen Befehlen, die ohne erneutes Einlesen anwendbar sind. Auch `COPY` ist ein interner Befehl.

Als DOS allmählich größer wurde und neue oder verbesserte Befehle erhielt, wurden diese zu selbständigen Programmen, die nur dann Platz im Speicher beanspruchen, wenn sie angewendet werden. Programme, die solche externen Befehle ausführen, liegen im allgemeinen auf der Festplatte in einem Verzeichnis namens `C:\DOS`.

`COPY` war von Anfang an ein häufig verwendeter interner Befehl. Mit der Zeit wollte man ihn mit weiteren Möglichkeiten ausstatten – das war aber unvereinbar mit der Forderung nach geringer Größe der Datei `COMMAND.COM`.

Darum hat man statt dessen eine intelligentere Ausgabe von `COPY` als externe Parallele des ursprünglichen Befehls entwickelt. Dieser fortschrittliche Vetter erhielt den Namen `XCOPY` – das X steht hier für **eX**ternal.

Sicherheitskopie

`XCOPY` verfügt über einige nützliche Fähigkeiten, die man durch Zufügen eines Schalters nach dem Befehl benutzen kann.

Ein oft benutzter Schalter für `XCOPY` ist `/S` – er bedeutet: 'Unterverzeichnisse eingeschlossen'.

`/S` ist ein Beispiel für einen Schalter, der in den meisten Befehlen dasselbe bedeutet. Beachte aber Seite 41.

Schreibst du

```
A:\>XCOPY C:\*.*
```

oder kürzer – ohne Sterne

```
A:\>XCOPY C:\.
```

bedeutet das, daß alle Dateien in deinem Verzeichnis `C:\` (dem Stammverzeichnis deiner Festplatte) auf das A-Laufwerk zu kopieren sind.

Fügst du `/S` an diesen Befehl:

```
A:\>XCOPY /S C:\.
```

heißt das, daß die Kopierung für alle Dateien des Stammverzeichnisses wie auch seiner sämtlichen Unterverzeichnisse gilt – also für alle Dateien auf der Festplatte; und das wird vermutlich auf der Diskette zu Platzproblemen führen ...

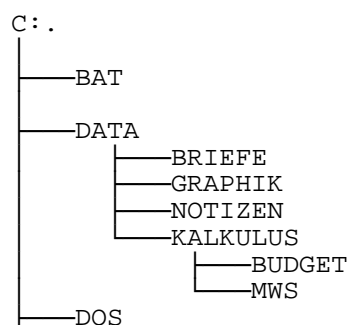
`/S` schließt alle Dateien aller Verzeichnisse ein, die unter dem aktuellen liegen.

Liegen alle deine Dokumente in `C:\DATA` samt seinen Unterverzeichnissen, kannst du sie alle ins A-Laufwerk kopieren, indem du schreibst

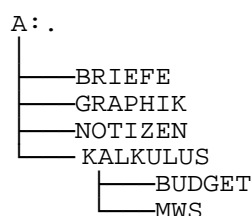
```
A:>XCOPY /S C:\DATA\.
```

Der Befehl erstellt eine getreue Kopie deiner Dateien in ihren jeweiligen Unterverzeichnissen. Das heißt, es wird eine Kopie eines ganzen Zweiges des kompletten Verzeichnisbaums erstellt.

Hat der Verzeichnisbaum der Festplatte das folgende Aussehen, wird nur die Verzweigung unterhalb DATA kopiert:



Auf der Diskette hat der Baum folgendes Aussehen::



Findet der gezeigte Zweig mit seinen Dateien nicht genug Platz auf der Diskette, muß das Kopieren auf andere Weise vorgenommen werden (siehe Seite 33).

XCOPY /S erstellt am Zielort automatisch neue Unterverzeichnisse – du brauchst das also nicht selbst zu tun.

Sollen größere Dateigruppen gespeichert werden, so ist die Methode, ganze Zweige zu kopieren, sehr praktisch.

In unserem Beispiel enthält dein Verzeichnis DATA sämtliche Dateien, die aus Sicherheitsgründen regelmäßig auf Disketten kopiert werden sollten.

Dokumente können nicht ohne weiteres ersetzt werden, wenn sie durch einen Fehler gelöscht wurden oder wenn die Festplatte einen Kollaps hat.

Programme dagegen lassen sich jederzeit von den Originaldisketten wieder auf die Festplatte installieren.

Attribute

Kopierst du regelmäßig alle Dokumente deiner Festplatte auf Disketten, hast du dich gut gegen den Verlust deiner Daten gesichert. Andererseits wirst du vermutlich bald zu viele Sicherheitskopien ein und derselben Datei haben.

Dieses Problem läßt sich lösen, indem du nur Kopien von Dateien erstellst, die seit dem letzten Mal neu erstellt oder geändert wurden.

Willst du kontrollieren, welche Dateien kopiert wurden und welche nicht, kannst du sie mit einer kleinen Marke versehen, einem Archiv-Attribut.

Ein Archiv-Attribut – ein scheußlich langer Name, den ich schleunigst in A-Attribut abkürze – kann ein- oder ausgeschaltet sein.

Auf Englisch heißt das *Set* und *Cleared*.

Ist das A-Attribut eingeschaltet, bedeutet das im allgemeinen, daß die Datei nicht kopiert ist.

Anfänglich ist das A-Attribut eingeschaltet. Das gilt für neue wie auch für geänderte Dateien.

Ob das A-Attribut ausgeschaltet ist, stellst du fest mit dem Befehl **ATTRIB**, gefolgt vom Namen der Datei.

Ist das Attribut eingeschaltet, steht vor seinem Namen ein A – andernfalls steht dort nichts.

Du kannst das A-Attribut einschalten mit dem Befehl **ATTRIB +A** und dem Namen der Datei.

Entsprechend kannst du es mit dem Befehl

ATTRIB -A ausschalten.

Du kannst den Bindestrich ohne weiteres als Minus verwenden.

Eigentlich gibt es für jede Datei vier Attribute:

A	Archive	(Datei nicht gespeichert)
S	System	(Systemdatei)
H	Hidden	(versteckte Datei)
R	Read-only	(schreibgeschützte Datei)

Wie das A-Attribut kannst du auch die anderen Attribute mit **ATTRIB** ein- oder ausschalten.

Der Letzte macht das Licht aus

XCOPY hat einen Schalter **/M** der dafür sorgt, daß nur Dateien mit eingeschaltetem A-Attribut kopiert werden.

/M ist ausgefuchst genug, selbst das A-Attribut nach dem Kopieren auszuschalten.

Wenn du den Befehl **XCOPY /M** zum ersten Mal verwendest, solltest du sicherstellen, daß das A-Attribut für alle deine Dokumente eingeschaltet ist. Es ist durchaus möglich, daß es bei einigen ausgeschaltet wurde und das deiner Aufmerksamkeit entging. Du schaltest das A-Attribut für sämtliche Dateien ein, indem du schreibst **ATTRIB /S +A**. Der Schalter **/S** bezieht alle Unterverzeichnisse mit ein.

Führst du keine Dateinamen an, geht DOS davon aus, daß du sämtliche Dateien meinst.

```
C:\>CD DATA
C:\DATA>ATTRIB /S +A
```

Nun bist du bereit, mit **XCOPY** zu kopieren. Du verwendest **/S**, um alle Dateien der Unterverzeichnisse einzubeziehen, und **/M**, wenn du nur Dateien mit eingeschaltetem A-Attribut kopieren willst:

```
C:\DATA>A:
A:\>XCOPY /S/M C:\DATA\.
```

Nun werden alle Dokumente kopiert und anschließend ihre A-Attribute ausgeschaltet.

Sind es viele Dateien, so wird der Platz auf der Diskette bald gefüllt; nun gibt DOS folgende Mitteilung:

```
Zu wenig Speicherplatz auf dem
Datenträger
  38 Datei(en) kopiert
```

Du legst eine formatierte Diskette in das A-Laufwerk (siehe Seite 41) und wiederholst den Befehl. Nun haben die A-Attribute die Funktion eines Lesezeichens. **XCOPY /S/M** geht weiter im Text und kopiert die Dateien, deren A-Attribut noch nicht ausgeschaltet wurde.

Erstellst du nun, zum Beispiel eine Woche später, eine neue Sicherheitskopie mit demselben Befehl

```
A:\>XCOPY /S/M C:\DATA\.
```

wird **XCOPY /S/M** nur die Dateien kopieren, die in der Zwischenzeit mit einem A-Attribut versehen wurden – also neue oder geänderte Dateien.

Dies ist ein Beispiel für einen Befehl, den du häufig verwenden wirst und der ziemlich schwer zu behalten ist.

Es wird nützlich sein, den Befehl in einer sogenannten BAT-Datei zu speichern, die für diesen Zweck wie geschaffen ist.

Siehe Seite 48.

Nennst du die BAT-Datei **SICH.BAT**, brauchst du nur **SICH** zu schreiben, um den Befehl zu wiederholen und damit mit dem Kopieren zu starten.

Diese Art, Sicherheitskopien zu erstellen, kann dazu führen, daß dieselbe Datei in verschiedenen Ausgaben auf mehreren Disketten existiert.

Jedesmal, wenn du eine neue Diskette für deine Sicherheitskopien verwendest, wird eine geänderte Datei erstellt, ohne daß die frühere Ausgabe gelöscht wird. DOS kann nur sehen, was sich auf der aktuellen Diskette befindet – es kennt nicht den Inhalt der Disketten, die im Schrank stehen.

Diese teilweise Verdoppelung kann ein Nachteil sein, wenn du Diskettenplatz sparen willst; sie ist aber ein Vorteil, wenn du zum Beispiel Kopien eines Budgets in verschiedenen Stadien haben möchtest.

Je öfter du Sicherheitskopien erstellst, desto größer werden deine Möglichkeiten, Arbeit zu sparen, indem du eine auf der Festplatte gelöschte Datei rekonstruierst.

Hast du viel Arbeit in deine Dokumente gesteckt, solltest du deine Kopie-Disketten weit vom PC entfernt aufbewahren, um sie vor Brand oder Diebstahl zu schützen.

Anwendung einer Sicherheitskopie

Die Sicherheitskopie ist deine Garantie dafür, daß eine Datei nicht durch einen Fehler der Festplatte verloren geht. So lange alles in Ordnung ist, hast du einfach eine weitere Kopie deiner wichtigsten Dateien auf Disketten und denkst weiter nicht darüber nach. Verschwindet eine Datei aber aus der Festplatte, kannst du dir viel Arbeit ersparen, indem du sie zurückkopierst.

Nehmen wir an, daß du durch einen dummen Zufall alle Dateien in `C:\DATA\BRIEFE` löschst. Nun vermuten wir, daß du sechs Disketten mit Sicherheitskopien aus `C:\DATA` hast, die mit den Nummern 1 bis 6 versehen sind. Du legst Diskette 1 ins A-Laufwerk und wechselst auf der Festplatte ins Verzeichnis `DATA\BRIEFE` über:

```
C:\>CD \DATA\BRIEFE
C:\DATA\BRIEFE>
```

Dann wechselst du ins entsprechende Verzeichnis auf A:

```
C:\DATA\BRIEFE>A:
A:\>CD \DATA\BRIEFE
A:\DATA\BRIEFE>
```

und kehrst zu `C:` zurück:

```
A:\DATA\BRIEFE>C:
C:\DATA\BRIEFE>
```

Nun hat DOS seinen *Fingerabdruck* auf die Verzeichnisse beider Laufwerke gesetzt. Jedesmal, wenn du das Laufwerk wechselst, bist du im richtigen Verzeichnis.

Du kopierst jetzt alle Dateien in diesem Verzeichnis über den Befehl `COPY` oder `XCOPY` von `A:` nach `C:`

```
C:\DATA\BRIEFE>COPY A:.
```

Sind die Dateien kopiert, wiederholst du den Befehl mit der Diskette 2:

```
C:\DATA\BRIEFE>COPY A:.
```

Vergiß nicht, daß du einen Befehl wiederholen kannst, indem du `F3` drückst und anschließend Eingabe. Du kannst auch mit `DOSKEY` den AufwärtsPfeil drücken.

Fehlt dein Unterverzeichnis `BRIEFE` auf einer oder mehreren Disketten fehlt, teilt es mit:

```
Datei nicht gefunden
```

und du gehst zur nächsten Diskette weiter. Das machst du so lange, bis alle Dateien aus `A:\DATA\BRIEFE` auf den sechs Disketten kopiert sind.

Du mußt die Disketten in der richtigen Zahlenreihenfolge kopieren, da es vielleicht zwei Dateien gleichen Namens auf zwei verschiedenen Disketten gibt.

Hast du zum Beispiel den Brief `ULRICH.DOC` im März auf die Diskette 3 kopiert und eine korrigierte Ausgabe dieses Briefs im Juni auf die Diskette 5, möchtest du vermutlich die spätere Ausgabe haben.

Hast du `ULRICH.DOC` von der Diskette 3 kopiert und kopierst später `ULRICH.DOC` von der Diskette 5, wird die neuere Ausgabe die ältere überschreiben.

Benötigst du nur einen einzelnen Brief in deinen Sicherheitskopien, zum Beispiel einen, der so ähnlich lautet wie `KIM`, kannst du eine Kurzübersicht der mit `KIM` beginnenden Dateien auf den einzelnen Disketten erstellen, indem du schreibst

```
C:\DATA\BRIEFE>A:
A:\DATA\BRIEFE>DIR KIM*
```

Enthält die Kurzübersicht den gesuchten Brief, zum Beispiel `KIM2.DOC` auf der Diskette 4, kopierst du diese Datei auf die Festplatte, während die Diskette 4 im A-Laufwerk liegt:

```
A:\DATA\BRIEFE>C:
C:\DATA\BRIEFE>COPY A:KIM2.DOC
```

Bearbeitung einer Textdatei

Eine Textdatei ist eine Datei, die ausschließlich normale, das heißt lesbare, Zeichen enthält.

Ein Dokument einer Textverarbeitung, etwa ein Brief, ist keineswegs eine Textdatei – es besteht nämlich aus Text und aus Kodierungen. Der eigentliche Briefftext findet sich in der Datei in Form von gewöhnlichen Zeichen; außerdem setzt das Programm aber Kodierungen ein, die verschiedene Eigenschaften des Textes angeben. Das kann Schriftart, Schriftgröße, Seitenränder und anderes betreffen. Man sagt, daß die Datei formatierten Text enthält.

Die Datei ist darauf eingerichtet, in dem Programm gelesen und bearbeitet zu werden, in dem sie erstellt wurde – dem Programm, in dem diese Kodierungen eine bestimmte Bedeutung haben. Willst du einen von einem bestimmten Programm formatierten Text laden, mußt du entweder dieses Programm selbst besitzen oder die Kodierungen, die Formate, in deine eigene Textverarbeitung übersetzen können.

Bei reinen Textdateien sieht das anders aus. Im Gegensatz zu Dateien mit formatiertem Text enthalten sie keine Kodierungen. Textdateien werden von DOS für bestimmte Zwecke verwendet, unter anderem für BAT-Dateien (siehe Seite 48) und Startdateien (siehe Seite 46).

Abgesehen davon werden Textdateien vor allem als Gebrauchsanweisungen für Programme verwendet, die ohne gedruckte Anweisung geliefert werden, zum Beispiel Shareware (siehe Seite 50).

Shareware-Programme werden in der Regel mit Informationen enthaltenden Textdateien geliefert, die über die Befehle `TYPE` und `PRINT` lesbar sind, ohne auf eine Textverarbeitung zuzugreifen.

Willst du eine BAT-Datei erstellen oder eine Änderung in einer Startdatei vornehmen, kannst du einen Editor verwenden, also eine Textverarbeitung für Textdateien.

DOS 5 und 6 werden mit einem Editor mit dem Dateinamen `EDIT.COM` ausgeliefert, der normalerweise in `C:\DOS` liegt. Ist `C:\DOS` auf dem Suchpfad, kannst du diesen Editor von überall aus starten, indem du einfach schreibst

```
C:\>EDIT
```

Nun meldet er sich auf dem Bildschirm mit einem Grußfeld in der Mitte.

Zunächst einmal wird dir nun eine kurze Einführung angeboten, die du wählen kannst, indem du Eingabe tatest. Weiterhin enthält der Editor eine Hilfsfunktion, die du während der Arbeit anwenden kannst. Drückst du die `Esc`-Taste, verschwindet der Gruß, und du kannst die Bearbeitung beginnen.

Der Editor besteht aus einer viereckigen Fläche, in der du wie in einer Textverarbeitung schreiben und redigieren kannst. In einem Feld am oberen Rand der Fläche steht der Name der Datei, die gerade bearbeitet wird. Beginnst du eine neue Datei, steht dort **Unbenannt**.

Am oberen und unteren Rand siehst du je einen waagerechten Balken. Der obere Balken enthält die Menüs, die du für die Bearbeitung benötigst. Jedes der folgenden Worte **Datei Bearbeiten Suchen Optionen** und **Hilfe** ist eine Überschrift für ein selbständiges DropDownFeld.

Die Menüs werden aktiviert, indem du die `Alt`-Taste hältst, während du den ersten Buchstaben der gewünschten Überschrift drückst. Das DropDownFeld unter **Datei** fällt wie ein Rouleau vom Balken, wenn du **Alt+D** eingibst.

Willst du eine bestimmte Datei bearbeiten, wählst du **Datei** und anschließend **Öffnen** – das gibt dir die Möglichkeit, die gewünschte Datei zu holen.

Hast du eine Datei in den Editor geholt, siehst du ihren Inhalt in der linken Hälfte der Fläche. Mit den Pfeiltasten kannst du nun den Cursor frei an eine Stelle der Datei setzen. Du kannst Text löschen oder hinzufügen, Linien verschieben und vieles andere mehr.

Text läßt sich problemlos verschieben, da das Programm ausschneiden, kopieren und einfügen kann.

Ausschneiden bedeutet, daß ein Teil des Textes entfernt und im Speicher in eine Zwischenablage gelegt wird – eine kurzfristige Lagerung, die funktioniert, so lange `EDIT` aktiv ist.

Kopieren setzt den Text ebenfalls in die Zwischenablage, bewahrt ihn aber gleichzeitig in der Datei.

Einfügen fügt den Inhalt der Zwischenablage an der Stelle der Datei ein, wo der Cursor steht. Die Zwischenablage enthält jeweils nur den zuletzt in sie gelegten Text.

Ausschneiden, **Kopieren** und **Einfügen** finden sich unter dem Menü **Bearbeiten** und können durch die dort angegebenen Tastenkombinationen aktiviert werden. Text, der in die Zwischenablage gesetzt werden soll, muß zunächst markiert werden.

Hältst du die **Umschalttaste**, während du den Cursor mit den **Pfeiltasten** bewegst, wird der Text im vom Cursor berührten Bereich markiert. Willst du etwa eine ganze Zeile verschieben, markierst du sie zunächst durch **Umschalt+PfeilAb**. Nun kann sie in die Zwischenablage ausgeschnitten werden, indem du **Umschalt+Entf** drückst. Setze nun den Cursor dort, wo du die Zeile einfügen willst. Wenn du anschließend **Umschalt+Einf** drückst, wird die Zeile aus der Zwischenablage geholt.

Willst du die Datei nicht weiter bearbeiten, archivierst du sie. Dafür wählst du zuerst **Datei** und dann **Speichern unter** (**Alt+D** und ein **U**). Willst du der Datei einen neuen Namen geben, schreibst du diesen in die Box und bestätigst das mit **Eingabe**.

Achte sorgfältig auf den Unterschied zwischen **Speichern** und dem nächsten Punkt **Speichern unter**. **Speichern** archiviert die Datei unter ihrem originalen Namen. **Speichern unter** bittet dich dagegen ausdrücklich um einen neuen. Wählst du **Speichern**, wird der ursprüngliche Inhalt der Datei durch dem neuen Inhalt ersetzt, was bedeutet, daß du nicht mehr zu deinem Ausgangspunkt zurückkehren kannst. **Speichern unter** gibt dir dagegen die Möglichkeit, dem Ergebnis deiner Bearbeitung einen neuen Namen zu geben und damit das Original zu bewahren.

Du verläßt den Editor, indem du **Datei** und anschließend **Beenden** wählst (**Alt+D** und dann **B**).

Hast du die bearbeitete Datei noch nicht archiviert, fragt der Editor, ob sie gespeichert werden soll, wenn du **Beenden** wählst.

Aber Vorsicht: Hier kannst du nur **Speichern** wählen, nicht aber **Speichern unter**. Möchtest du die ursprüngliche Datei nicht unter ihrem originalen Namen speichern, mußt du dein **Beenden** mit dem Befehl **Abbrechen** unterbrechen und die Datei durch **Speichern unter** mit einem neuen Namen speichern.

Verwendest du eine Maus, aktivierst du die erwähnten Menüs durch Zeigen und Klicken mit der linken Maustaste. Die Maus macht es möglich, den Cursor schneller durch den Text zu bewegen.

Für die folgenden Abschnitte benötigst du eine Datei, an der du dich üben kannst. Diese Datei muß durch Feuer und Wasser gehen – sie wird unter anderem gelöscht und hinterher wieder gerettet. Möchtest du deine existierenden Dateien vor peinlichen Unglücken bewahren, empfehle ich dir also, mit dem Editor eine Übungsdatei zu erstellen.

Wenn du den Editor startest, kannst du nach dem Wort `EDIT` den Namen der zu bearbeitenden Datei schreiben – auch wenn die Datei noch gar nicht existiert. Wir nennen unsere Übungsdatei `NOTIZ.TXT` und beginnen im Stammverzeichnis, indem wir schreiben

```
C:\>EDIT NOTIZ.TXT
```

Schreibe nun im Editor einen Text von fünf bis sechs Zeilen. Der Inhalt bleibt dir überlassen; unterwegs kannst du aber probeweise die Zeilen verschieben oder andere entsprechende Dinge tun. Brauchst du Hilfe, kannst du jederzeit die **F1**-Taste drücken.

Ist die Datei fertig, archivierst du sie mit **Speichern unter**.

Da du den Editor im Stammverzeichnis aktiviert und deine Datei `NOTIZ.TXT` genannt hat, schlägt der Editor `C:\NOTIZ.TXT` vor.

Du akzeptierst den Vorschlag und beendest `EDIT`. Anschließend erstellst du eine Kurzübersicht über das Stammverzeichnis, um zu sehen, ob die Datei tatsächlich angekommen ist:

```
C:\>DIR NOTIZ.TXT
```

Eine Datei löschen

Du löschst eine Datei mit dem Befehl `DEL` (kurz für Delete, das englische Wort für entfernen).

Willst du die Datei `NOTIZ.TXT` löschen, schreibst du

```
C:\>DEL NOTIZ.TXT
```

Es vergeht kaum der Bruchteil einer Sekunde, bis wieder der Prompt erscheint:

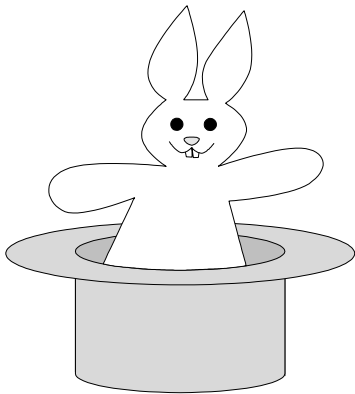
```
C:\>
```

Nun ist die Datei gelöscht!

Möchtest du dich vergewissern, erstellst du über `DIR` eine neue Kurzübersicht.

Der Befehl `DEL` ist ein bißchen gefährlich – er ist nämlich schnell und schonungslos. Es erfolgt keine Warnung und auch keine Bestätigung, daß eine Datei gelöscht wurde.

Das ist natürlich ein Problem; daher stellen MS-DOS 5 und 6 den Befehl `UNDELETE` zur Verfügung, mit dessen Hilfe man eine solche Löschung rückgängig machen und die Datei wiederherstellen kann.



UNDELETE kann eine gelöschte Datei retten

Undelete

`UNDELETE` bedeutet un-löschen.

Das ist ein ziemlich nützliches Werkzeug, das eine gelöschte Datei retten kann.

Das Geheimnis, das es möglich macht, die Löschung einer Datei rückgängig zu machen, ist, daß der Inhalt dieser Datei nach ihrer Löschung für eine Weile unverändert bleibt.

Wenn du ein Musikstück auf einem Tonband löschst, so änderst du durch diesen Prozeß alle "Daten" auf dem Band. Die ursprüngliche Aufnahme wird nullgestellt, so daß sie durch etwas anderes ersetzt werden kann.

Das ist nicht der Fall, wenn eine Datei gelöscht wird. Hier wird nämlich zunächst einmal nur das Inhaltsverzeichnis der Festplatte in Bezug auf die Datei modifiziert. Ihr Inhalt bleibt im selben Bereich der Platte liegen, auch wenn dieser Bereich nun als frei aufgeführt wird. Erst wenn eine neue Datei erscheint und sich auf diesem freien Platz breitmacht, ist der Inhalt der älteren Datei für immer verloren.

Eine entsprechende Situation wäre, daß du dich im Hotel abmeldest, ohne dein Zimmer zu räumen. Die Rezeption erfährt, daß du nicht mehr das Zimmer bewohnst, und bereitet sich darauf vor, es einem neuen Gast zu überlassen. Solange kein neuer Gast erscheint, kannst du deine Sachen stehen lassen und je nachdem deinen Auszug rückgängig machen. Also gilt es, neue Gäste am Betreten des Hotels zu hindern, wenn du dein altes Zimmer behalten willst.

Entsprechend sollte man die Etablierung neuer Dateien auf der Festplatte vermeiden, falls man eine Löschung bereut.

`UNDELETE` funktioniert am besten unmittelbar nach einer Löschung. Kopiere keine Dateien auf die Festplatte, bevor die gelöschte Datei wiederhergestellt wurde.

In DOS 5 gibt es ein, in DOS 6 zwei Hilfsprogramme für `UNDELETE` (siehe Seite 38).

Bevor wir uns genauer mit der erweiterten Anwendung von UNDELETE befassen, sehen wir uns erst einmal die Standardmethode an, die auf dem Bildschirm MS-DOS-Verzeichnis genannt wird.

Diese Methode, die ich für völlig ausreichend halte, benutzt man, indem man UNDELETE mit dem Schalter /DOS ausstattet.

Versuche nun, die Datei NOTIZ.TXT, die du gerade aus dem Stammverzeichnis gelöscht hast, wiederherzustellen.

Du gehst in das Verzeichnis, wo die Datei lag, und schreibst

```
C:\>UNDELETE NOTIZ.TXT /DOS
```

Nun erscheinen einige Mitteilungen auf dem Bildschirm. Zuunterst siehst du

Das MS-DOS-Verzeichnis wird verwendet.

```
?OTIZ TXT 1483 27-05-94 21.43
...A Wiederherstellen (J/N)?
```

Du liest den Namen der Datei, ihre Größe und Datums- und Zeitangabe.

Die Frage Wiederherstellen (J/N)? beantwortest du mit J für Ja.

Wie du siehst, wurde der erste Buchstabe des Vornamens durch ein Fragezeichen ersetzt. Du wirst nun aufgefordert, den ersten Buchstaben des Dateinamens zu schreiben:

Geben Sie den ersten Buchstaben des Dateinamens ein: ?OTIZ.TXT :

Hier schreibst du N und erhältst die erfreuliche Mitteilung:

```
Datei erfolgreich wiederher-
gestellt
```

```
C:\>
```

Die Datei ist wieder da, und der Prompt erscheint erneut.

Daß du den ersten Buchstaben des Dateinamens einfügen mußt, liegt daran, daß dieses Zeichen im Inhaltsverzeichnis der Festplatte tatsächlich mit einem Fragezeichen überschrieben ist, das angibt, daß diese Datei gelöscht ist. Das entspricht einer Streichung deines Namens im Gästebuch des Hotels.

Du kannst auch Wild Cards benutzen, wenn du UNDELETE nach der Standardmethode verwendest:

```
C:\>UNDELETE *.* /DOS
```

Doppelte Sicherung

Es gibt zwei Hilfsprogramme, die du installieren kannst, um gelöschte Dateien leichter retten zu können: DELETE-TRACKING und SENTRY

DELETE-TRACKING wird als Beigabe von MS-DOS Version 5 und 6 mitgeliefert und ist das bescheidenere der beiden Programme. Es sorgt dafür, daß der Platz, den eine gelöschte Datei auf der Festplatte beanspruchte, nicht sofort für neue Dateien zur Verfügung gestellt wird. Mit anderen Worten, eine gelassene Vermietungspolitik, durch die das Hotel immer eine bestimmte Anzahl bisher vermieteter Zimmer leerstehen läßt.

DELETE-TRACKING hat in der Version 5 nicht ganz so hochentwickelte Fähigkeiten wie in der späteren Version.

In dieser, der Version 6, kannst du DELETE-TRACKING für die 100 zuletzt gelöschten Dateien der Festplatte installieren, indem du schreibst

```
C:\>UNDELETE /T 100
```

Das Programm wird in den Speicher eingelesen und hält seine Hand über die 100 zuletzt gelöschten Dateien – es stehen sozusagen jederzeit hundert verlassene Zimmer leer. Allerdings übt das Programm eine Schutzfunktion erst von dem Augenblick an aus, in dem es eingelesen wurde. Willst du diese Funktion verwenden, wann immer dein Computer eingeschaltet ist, mußt du die Zeile

```
UNDELETE /T 100
```

in die AUTOEXEC.BAT einfügen (siehe Seite 46). Die Liste der reservierten Plätze wird in der Datei PCTRACKR.DEL ständig aktualisiert und jeweils auf der Festplatte gespeichert.

Löschst du viele Dateien, wird DELETE-TRACKING manche Operationen auf deinem PC verlangsamen.

SENTRY, das als Teil von MS-DOS 6 mitgeliefert wird, arbeitet mit noch größerer Sicherheit.

SENTRY erstellt ein besonderes Verzeichnis, in das alle Dateien vor dem Löschen kopiert werden.

Allen Gästen des Hotels wird ein kurzfristiger kostenloser Aufenthalt im Nebengebäude angeboten, wenn sie ihr Zimmer verlassen. Niemand wird auf die Straße gesetzt! Das Arrangement wirkt sehr human – aber es wird sicher viel Platz erfordern.

Willst du SENTRY installieren, schreibst du

```
C:\>UNDELETE /S
```

Wenn du dieses Programm zum ersten Mal anwendest, wird es ein neues Verzeichnis `C:\SENTRY` für die Kopien gelöschter Dateien anlegen – sozusagen ein Nebengebäude des Hotels. Das neue Verzeichnis ist versteckt – das heißt, es taucht nicht in einem normalen Verzeichnis auf – und kann bis zu 20% des gesamten Platzes deiner Festplatte füllen! Gleichzeitig wird eine “Verjährungsfrist” für die Aufbewahrung gelöschter Dateien festgelegt, zum Beispiel 7 Tage. Die gewählten Werte für `C:\SENTRY` und die Verjährungsfrist werden in der Datei `UNDELETE.INI` auf der Festplatte gespeichert.

Wie das oben erwähnte Programm muß SENTRY in den Speicher eingelesen werden, um aktiv zu sein – je nachdem über eine Zeile in der `AUTOEXEC.BAT`

Du kannst jederzeit feststellen, ob diese Hilfsprogramme aktiv sind, indem du schreibst

```
C:\>UNDELETE /STATUS
```

Erscheint die Mitteilung

```
UNDELETE nicht geladen
```

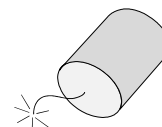
besagt das, daß die Hilfsprogramme nicht installiert sind. Du kannst aber immer noch UNDELETE in seiner reduzierten Ausgabe (mit dem Schalter /DOS) verwenden.

Du kannst ohne diese Hilfsprogramme auskommen, solange du nur daran denkst, eine Löschung baldigst rückgängig zu machen.

Ein Verzeichnis mit Dateien löschen

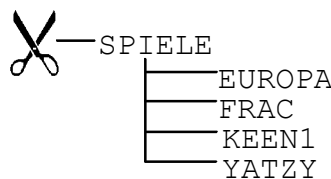
In DOS 6 kannst du durch den Befehl DELTREE ein ganzes Verzeichnis mit Dateien auf einmal löschen.

Der Befehl wird DELTREE genannt, weil er auch alle Unterverzeichnisse samt ihren Dateien löscht.



Das ist dasselbe wie die Entfernung einer ganzen Verzweigung aus dem Verzeichnisbaum. Bevor du deinen Zweig absägst, solltest du dir die Zeit nehmen, mit DIR einen Blick auf die Dateien der Verzeichnisse zu werfen, die du löschen willst. Dazu gibst du den Namen des Verzeichnisses an, wo die Verzweigung ansetzt.

Nehmen wir an, du willst ein Unterverzeichnis im Stamm löschen, das den Namen SPIELE trägt – mit seinem Inhalt an Dateien und weiteren Unterverzeichnissen. Die Verzweigung hat folgendes Aussehen:



Nun schreibst du

```
C:\>DELTREE SPIELE
```

DOS bittet dich, die Löschung zu bestätigen:

```
Löschen des Verzeichnisses SPIELE
und aller Unterverzeichnisse? [jn]
```

Die Buchstaben in eckigen Klammern `jn` bezeichnen deine Wahlmöglichkeiten.

Antwortest du `j` (ja), beginnt die Löschung.

Antwortest du `n` (nein), wird der Befehl abgebrochen.

Die Löschung wird alle Dateien in `C:\SPIELE` und seinen Unterverzeichnissen umfassen:

```
C:\SPIELE\EURO
C:\SPIELE\FRAC
C:\SPIELE\KEEN1
C:\SPIELE\YATZY
```



WARNUNG

Eine Löschung mit DELTREE läßt sich nicht durch UNDELETE rückgängig machen.

Disketten

Das Diskettenlaufwerk nimmt eine Sonderstellung ein, denn sein Inhalt wechselt jedesmal, wenn du eine neue Diskette in den Computer legst.

Willst du zum Beispiel im A-Laufwerk arbeiten, machst du dieses zu deinem aktuellen Laufwerk, indem du schreibst

```
C:\>A:
```

Liegt im A-Laufwerk keine Diskette, erscheint die Fehlermeldung

```
Nicht bereit zum Lesen von
Laufwerk A
(A)bbrechen, (W)iederholen,
(U)ebergehen?
```

Du wählst eine der drei angegebenen Möglichkeiten, indem du ihren Anfangsbuchstaben schreibst:

```
A falls du den Befehl abbrechen willst
W falls du es noch einmal versuchen willst
U falls DOS den Fehler übergehen soll
```

In der Regel wirst du schleunigst eine Diskette einlegen und auf `w` drücken.

Anschließend erscheint der Prompt mit dem Hauptverzeichnis des A-Laufwerks:

```
A:\>
```

Eine Diskette - auch Floppydisk genannt - wird aus weichem (floppy) Material hergestellt und läßt sich in das Laufwerk des Computers einsetzen und wieder entfernen. Früher verwendete man Disketten mit einem Durchmesser von 5 1/4" in einer dünnen Plastikhülle. Heute werden vor allem 3 1/2"-Disketten mit stabiler Hülle verwendet.

Sie werden vor allem in folgenden zwei Qualitäten geliefert:

<p>2DD oder DS, DD Double Side, Double Density für 720 KiloByte</p>

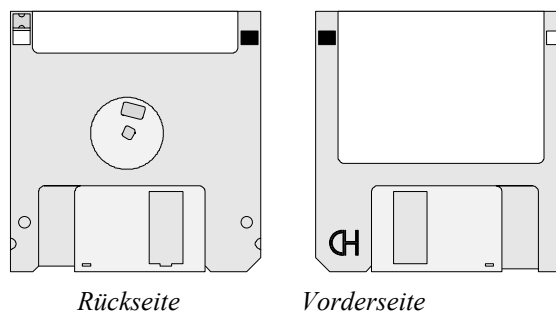
<p>2HD oder DS, HD Double Side, High Density für 1,44 MegaByte</p>
--

Die Abkürzungen 2 oder DS besagen, daß die Diskette beidseitig beschrieben wird, auf Englisch Double Side. Diese Bezeichnung ist heute Standard - und im Grunde etwas verwirrend, da sie mit Double Density verwechselt werden kann.

Der wesentliche Unterschied dieser beiden Formen ist ihre Dichte, engl. Density, die angibt, wie feinkörnig die magnetische Beschichtung ist. Auf einer HD (High Density) liegen die Spuren dichter und können daher doppelt so viele Daten enthalten wie eine DD (Double Density).

Der Preisunterschied zwischen DD- und HD-Disketten ist so gering, daß es sich lohnt, HD zu kaufen. DD wirst du nur dann benötigen, wenn du Kopien existierender DD-Disketten erstellen oder Disketten an andere Leute schicken willst, deren Maschine zu alt ist, um HD-Disketten lesen zu können.

Der Computer erkennt eine HD-Diskette mechanisch durch ein besonderes Loch in ihrer Hülle. Dieses HD-Loch erscheint in der Zeichnung schwarz.



In beiden Diskettentypen gibt es ein weiteres Loch, das mit einem Schieber geöffnet oder verschlossen werden kann; es dient zum Schreibschutz der Diskette. Ist es offen, können keine Daten auf die Diskette geschrieben werden, während sie ohne weiteres lesbar ist.

Die Diskette wird mit ihrem Metallende eingelegt und wendet die Rückseite dem Eject-Knopf zu, mit dessen Hilfe du die Diskette nach Gebrauch wieder auswirfst. Wenn es klickt, sitzt sie richtig.

Auf der Zeichnung siehst du die Diskette mit dem Metallstück nach unten. So pflege ich meine Disketten in eine Box zu stellen - dadurch kann ich sie nämlich wie Karteikarten durchblättern und habe immer das große Etikett oben.

Eine ganze Diskette kopieren

Willst du eine ganze Diskette auf eine andere kopieren, wendest du `DISKCOPY` an:

```
C:\>DISKCOPY A: A:
```

Dieser Befehl kopiert den Inhalt einer Diskette im A-Laufwerk auf eine andere Diskette im A-Laufwerk. Es wird eine getreue Kopie erstellt, was bedeutet, daß alle Dateien auf beiden Disketten an genau demselben Ort erscheinen. Das erfordert weiterhin, daß es sich um zwei Disketten des *gleichen* Typs handelt. Du kannst `DISKCOPY` nicht von einer DD- auf eine HD-Diskette oder umgekehrt anwenden.

Hat DOS einen Teil des Inhaltes deiner Originaldiskette (deiner *Source Disk*) gelesen, wird dieser zwischenzeitlich in der Maschine gelagert. Legst du nun eine neue Diskette (*Target Disk*) ins A-Laufwerk, wird der Inhalt dorthin kopiert. In den meisten Versionen von DOS wird jeweils nur ein kleiner Teil des Inhalts einer Diskette gelagert – also solltest du darauf gefaßt sein, unterwegs die Disketten zu wechseln. Erst seit DOS 6.2 kann der Computer den Inhalt einer kompletten Diskette lagern.

Die Target-Diskette darf nicht schreibgeschützt sein.

Ist die Target-Diskette nicht formatiert, sorgt `DISKCOPY` unterwegs für ihre Formatierung. Mehr zu diesem Thema in der nächsten Spalte.



WARNUNG

Gibt es bereits Dateien auf der Target-Diskette, werden sie alle durch `DISKCOPY` gelöscht.

Formatierung

Auf einer Diskette werden die Daten auf einer Plastikfolie aufgezeichnet, deren Oberfläche ähnlich wie ein Tonband durch eine dünne Metallschicht magnetisch gemacht wurde. Bevor die Diskette Daten aufnehmen kann, muß sie beidseitig mit “Schienen” ausgestattet werden, ähnlich wie die Rillen einer Schallplatte. Diese Schienen werden nicht automatisch erstellt, wenn du Daten auf eine neue Diskette überführst – du mußt sie erst durch eine sogenannte Formatierung schaffen. Du formatierst eine neue Diskette durch den Befehl `FORMAT`.

Du schreibst

```
C:\>FORMAT A:
```

Sogleich erscheint folgende Mitteilung auf dem Bildschirm:

```
Neue Diskette in Laufwerk A:
einlegen und anschliessend die
EINGABETASTE drücken...
```

Hast du deine neue Diskette noch nicht ins A-Laufwerk gelegt, tust du das nun und drückst Eingabe.

Zunächst untersucht DOS, ob die Diskette eine DD- oder eine HD-Diskette ist. Dann beginnt die eigentliche Formatierung. Je nach deiner Ausgabe von DOS erscheinen nun Informationen auf dem Bildschirm – zum Beispiel:

```
Prüfe bestehendes Datenträger-
Format.
Überprüfe 1.44 MB
```

Ist die Formatierung beendet, erscheint folgende Mitteilung:

```
Formatieren beendet.
```

```
Datenträgerbezeichnung(11 Zeichen,
EINGABETASTE für keine)?
```

Der letztere Text gibt dir die Möglichkeit, eine Bezeichnung der Diskette von bis zu 11 von dir bestimmten Zeichen zu schreiben, die als eine Art Überschrift auf der Diskette gelagert wird.

Anders als ein Dateiname kann eine Bezeichnung Zwischenräume und Punkt enthalten, zum Beispiel `BRIEFE NR. 3`

Willst du keine Bezeichnung angeben, tastest du nur Eingabe.

Nun erscheint eine Statistik der Diskette:

```
1,457,664 Byte Speicherplatz auf
dem Datenträger insgesamt.
1457664 Byte auf dem Datenträger
verfügbar.
512 Byte in jeder Zuordnungseinheit.
2847 Zuordnungseinheiten
auf dem Datenträger verfügbar.
Datenträgernummer: 1043-17F1
```

Schließlich wirst du gefragt:

```
Eine weitere Diskette formatieren
(J/N)?
```

was du mit **J** (Ja) oder **N** (Nein) beantwortest.

Heutzutage kannst du auch formatierte Disketten kaufen. Sie sind nur wenig teurer als unformatierte, und meiner Meinung nach ist das Geld dafür gut ausgegeben.



WARNUNG

Wendest du **FORMAT** auf eine bereits verwendete Diskette an, werden alle ihre Daten gelöscht.

Unformat

Ebenso wie **UNDELETE** eine Löschung rückgängig machen kann, gibt es einen Befehl, der eine Formatierung rückgängig macht.

Hast du irrtümlich eine Diskette formatiert, die Dateien enthält, die du noch benötigst, legst du diese Diskette ins A-Laufwerk und schreibst

```
C:\>UNFORMAT A:
```

Du drückst Eingabe und siehst nun eine Mitteilung, die eine Warnung enthält. Der Text macht unter anderem darauf aufmerksam, daß Dateien verschwinden können, falls sie nach der Formatierung geändert oder hinzugefügt wurden.

Endlich wirst du gefragt, ob du den Systembereich der Diskette berichtigen willst. Das bestätigst du, indem du **J** tatest.

Der Prompt kehrt zurück, und die Diskette hat ihren ursprünglichen Inhalt wieder.

Die Befehle **FORMAT** und **UNFORMAT** sind sowohl auf Disketten als auch auf die Festplatte anwendbar. Bist du nicht mit DOS vertraut, solltest du deine Festplatte möglichst nicht formatieren. Willst du unbedingt alle Dateien auf ihr löschen und noch einmal von vorn anfangen, sollte ein erfahrener DOS-Kenner in der Nähe sein.

Systemdiskette

Der Befehl **FORMAT** hat einen Schalter **/S**, mit dessen Hilfe du eine sogenannte Systemdiskette erstellen kannst.

Eine Systemdiskette enthält die Systemdateien, die einen Start des PCs mit dieser Diskette im Laufwerk ermöglichen.

Schaltest du deinen Computer ein, sucht er die Systemdateien zunächst auf **A:** und dann auf **C:**.

Liegt im A-Laufwerk eine Diskette, nimmt der Computer an, das sei eine Systemdiskette. Enthält sie keine Systemdateien, geht die Maschine nicht ohne weiteres auf **C:**, wie sie das sonst tut, sondern gibt die Fehlermeldung

```
Kein System oder Laufwerksfehler
Wechseln und eine Taste drücken
```

Normalerweise entfernst du einfach die Diskette und drückst irgendeine Taste. Daraufhin liest die Maschine die Systemdateien auf **C:**

Es kann aber vorkommen, daß DOS nicht von der Festplatte aus starten kann. Du kannst zum Beispiel eine Änderung in der **CONFIG.SYS** angebracht haben, die der Computer nicht versteht (siehe Seite 46), weswegen er den Dienst verweigert. In diesem Fall kannst du ihn mit einer Systemdiskette im Laufwerk A starten, falls du bereits eine solche hergestellt hast.

Am besten stellst du **JETZT** eine Systemdiskette her !

Du erstellst eine Systemdiskette, indem du eine Diskette ins A-Laufwerk legst und schreibst

```
C:\>FORMAT A: /S
```

Die Zufügung /S sorgt dafür, daß die Systemdateien IO.SYS und MSDOS.SYS mit dem Kommandoprozessor COMMAND.COM nach der Formatierung auf A: überführt werden.

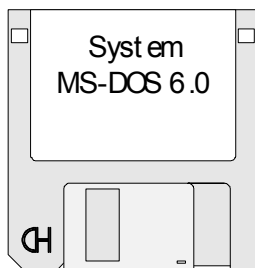
Beachte, daß der Schalter /S hier die Bedeutung **System** hat.

Du kannst auch eine Systemdiskette auf einer *bereits formatierten* Diskette erstellen.

Dazu setzt du die leere Diskette ins A-Laufwerk und schreibst

```
C:\>SYS A:
```

Dieser Befehl überführt ebenfalls die Systemdateien und den Kommandoprozessor.



Du schreibst SYSTEM samt DOS-Versionsnummer auf das Etikett der Diskette und bewahrst sie an einem sicheren Ort auf.

Wie du sahst, werden die Startdateien CONFIG.SYS und AUTOEXEC.BAT nicht automatisch auf die Systemdiskette überführt. Sie sind für den Start des Computers nicht erforderlich, da DOS seine eigenen Werte anwendet, falls diese Dateien fehlen (siehe Seite 46).

Mehr zum Thema Systemdisketten in *Nutze Deinen PC optimal*.

Die Festplatte

Die Festplatte ist sozusagen ein in den Computer eingebauter Stapel von Disketten. Sie besteht nicht aus Plastikfolienscheiben, sondern aus mehreren Metallscheiben übereinander, die sehr schnell um eine gemeinsame Achse rotieren. Das Lesen und Schreiben von Daten erfolgt durch Arme, die zwischen die Scheiben geschoben werden – wie ein Tonarm auf einem Plattenspieler. Dadurch ist eine Festplatte wesentlich schneller als eine Diskette. Außerdem enthält sie die bis zu fünfhundertfache Menge an Daten im Vergleich zu einer Diskette.

Der schwache Punkt einer Festplatte sind die Schreib-/Leseköpfe, die an ihren Armen zwischen den rotierenden Scheiben sitzen.



WARNUNG

Du solltest deinen Computer immer ausschalten, bevor du ihn bewegst.

Erhält die Maschine während der Rotation der Festplatte einen Schlag, können die Schreibköpfe Kratzer in den Belag reißen.

Es gibt spezielle Hilfsprogramme, die die Schreibköpfe der Festplatte während eines Transports beschützen sollen. Programme wie PARK und SHIPDISK setzen die Schreibköpfe über die innerste Spur der Festplatte. Viele neuere Maschinen parken die Schreibköpfe automatisch dort, wenn der Strom abgestellt wird.

Du kannst auch deine Festplatte schonen, indem du wenigstens 10 Sekunden wartest, bevor du den Computer nach einem Abschalten wieder anschaltest. Wegen ihrer schnellen Rotation braucht die Festplatte etwas Zeit, bis sie stillsteht – und es tut ihr nicht besonders gut, wieder eingeschaltet zu werden, während sie sich dreht.

Muß du deinen Computer erneut starten, solltest du den RESET-Knopf an der Vorderseite des Gehäuses drücken oder die Tastenkombination Strg+Alt+Entf verwenden, die das gleiche bewirkt. In diesem Fall machst du einen *Warmstart*, was bedeutet, daß die Festplatte nicht zum Stillstand kommt.

Hilfe

DOS 5 und 6 werden mit dem Programm `HELP` ausgeliefert, das Erläuterungen zu allen DOS-Befehlen gibt.

Weißt du von vornherein, welchen Befehl du genauer untersuchen willst, schreibst du `HELP` und anschließend den Namen des Befehls. Willst du zum Beispiel den Löschbefehl `DEL` untersuchen, schreibst du

```
C:\>HELP DEL
```

Nun stellt dir das Programm einen Text zu diesem Befehl bereit.

In DOS 5 faßt sich der Hilfstext kurz – er füllt nur ein Schirmbild.

DOS 6 startet nun ein etwas umfassendes Hilfsprogramm, dessen Text sich auf drei Ebenen lesen läßt: Syntax, Anmerkungen und Beispiele. Die Syntax zeigt die Regeln, die für die jeweilige Anwendung gelten. Von hier aus kannst du in die anderen Ebenen überwechseln, indem du die Hinweise anklickst, die am oberen Rand der Seite von Dreiecken umrahmt sind. Du kannst auch den Cursor in einen Hinweis stellen, indem du dessen ersten Buchstaben schreibst und dann Eingabe tastest. Du durchblätterst den Text mit den Pfeiltasten oder den Bildtasten. Am unteren Ende der Seite siehst du die Tastenkombinationen, die dir jederzeit zur Verfügung stehen: `Alt+I` (Inhalt) zeigt eine Gesamtübersicht der Befehle, `Alt+W` (Weiter) zeigt den nächsten Punkt im Programm, und `Alt+Z` (Zurück) springt zum letzten Punkt zurück.

Der Text zu `DEL` sagt dir zunächst, daß der Befehl die angegebenen Dateien löscht. Außerdem wird erwähnt, daß der Befehl auch `ERASE` heißt.

Die Syntax ist sehr simpel:

```
DEL [Laufwerk:][Pfad]Dateiname
[/P]
```

Die Glieder, die unmittelbar nach dem Namen eines Befehls geschrieben werden, werden als Parameter bezeichnet.

`COPY` zum Beispiel kann zwei Parameter haben: die Originaldatei und ihre Kopie. `DEL` hat nur einen Parameter: die zu löschende Datei.

Unter dem Punkt Parameter wird erklärt, wie du die zu löschende Datei in drei Teilen beschreiben kannst:

- Laufwerk (Buchstabe plus Doppelpunkt),
- Pfad
- Dateiname (Vor- und Nachname).

Die ersten beiden Teile werden in viereckigen Klammern gezeigt, was bedeutet, daß sie wahlfrei (optional) sind, daß du also selbst bestimmst, ob du sie schreibst. Du mußt also nicht unbedingt Laufwerk und Pfad der zu löschenden Datei erwähnen – in diesem Fall gilt der Befehl aber nur für Dateien im aktuellen Verzeichnis.

Dreht es sich um eine Datei auf einem anderen Laufwerk oder in einem anderen Verzeichnis, mußt du die ersten beiden Parameter anwenden.

Der Schalter `/P` ist ein weiteres Glied nach dem Parameter. Wendest du ihn an, wirst du um eine Bestätigung der Löschung gebeten, bevor sie ausgeführt wird.

Die Anmerkungen zu `DEL` weisen auf Dinge hin, die du beachten solltest, wenn du eine oder mehrere Dateien löschst. Unter anderem findest du den wichtigen Rat, `UNDELETE` so schnell wie möglich anzuwenden, wenn du eine Löschung bereust.

Benötigst du eine Kurzfassung der Hilfe für einen bestimmten Befehl, schreibst du diesen Befehl und dann den Schalter `/?`

```
C:\>DEL /?
```

In diesem Fall erschienen sechs Zeilen mit Erläuterungen zu `DEL` auf dem Bildschirm.

Zeichen

Ein PC arbeitet mit einem Zeichensatz, der aus 256 Zeichen besteht. Wie diese 256 Zeichen auf dem Bildschirm oder dem Drucker aussehen, das hängt von dem verwendeten Standard ab. Textdateien in DOS verwenden den sogenannten ASCII-Zeichensatz. ASCII ist eine Abkürzung für American Standard Code for Information Interchange.

Manche Programme, unter anderem Windows, verwenden einen anderen Standard, nämlich ANSI (American National Standardisation Institute), der im großen und ganzen dieselben Werte benutzt; solange du aber in DOS arbeitest, kannst du dich mit ASCII begnügen. Jedes Zeichen hat eine Zahl von 0 bis 255, die der ASCII-Wert des Zeichens genannt wird. Einige der Zeichen werden für Text verwendet, andere sind Kodierungen für bestimmte Zwecke.

In der Tabelle siehst du einige der meistverwendeten Zeichen für Text. Die ASCII-Werte erscheinen außerhalb des Rahmens. Das Ausrufezeichen mit dem Wert 30+3=33 ist das erste normale Zeichen der Tabelle. An ihrem Ende liegt das Ü mit dem Wert 150+4=154.

	0	1	2	3	4	5	6	7	8	9
30				!	"	#	\$	%	&	'
40	()	*	+	,	-	.	/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	@	A	B	C	D	E
70	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y
90	Z	[\]	^	_	`	a	b	c
100	d	e	f	g	h	i	j	k	l	m
110	n	o	p	q	r	s	t	u	v	w
120	x	y	z	{		}	~			ü
130			ä							
140			Ä						ö	
150			ö	ü						

Textzeichen und ihre ASCII-Werte

Wenn du in DOS arbeitest, kannst du alle Zeichen des ASCII-Zeichensatzes durch ihre Werte schreiben – das geht folgendermaßen vor sich:

Du hältst die Alt-Taste, während du auf den Zahlentasten an der rechten Seite der Tastatur die jeweiligen Werte eingibst – die normalen Zahlentasten kannst du hierfür nicht verwenden. Laß die Alt-Taste erst los, wenn der gesamte Wert eingegeben ist.

Schreibst du Alt+58, erhältst du :

Schreibst du Alt+92, erhältst du \

Die Methode funktioniert unabhängig von der Platzierung der übrigen Tasten auf deiner Tastatur.

Startest du deinen Computer etwa von einer Systemdiskette ohne deutsche Tastatureinstellung, ändern viele Tasten ihre Besetzungen. Das gilt unter anderem für den Doppelpunkt und den Backslash, die in vielen Befehlen verwendet werden. Du kannst jederzeit Alt+58 und Alt+92 eingeben, um diese Zeichen zu schreiben.

PS: Das Zeichen **ß** wird mit Alt+225 erhalten.

Schreibe nun mit Hilfe der ASCII-Werte der Tabelle mehrere Zeichen nach dieser Methode.

Die Startdateien

Zwei Dateien spielen eine besondere Rolle, sobald die Systemdateien und damit DOS in den Computer eingelesen wurden.

Das sind die sogenannten Startdateien, die die Arbeitsweise deines PCs sozusagen feinpolicieren. Diese Dateien, die `CONFIG.SYS` und `AUTOEXEC.BAT` heißen, müssen im Stammverzeichnis dort liegen, wo auch die Systemdateien sind.

`CONFIG.SYS` enthält eine Anzahl von Zeilen, die die Konfiguration des PCs beschreiben – das heißt unter anderem die Steuerung des Speichers und das Aussehen des Bildschirms.

`AUTOEXEC.BAT` enthält DOS-Befehle, die bei jedem Start automatisch durchgeführt werden. Das sind normalerweise Dinge wie die Definition des Prompts, die Festlegung des `PATH` und des Zeichensatzes – hier unter anderem die Fähigkeit, Ä, Ö und Ü zu schreiben – und die Installation der Maus.

Außerdem wird durch die `AUTOEXEC.BAT` oft ein bestimmtes Programm gestartet, zum Beispiel ein Menü, sobald der Computer hochgefahren ist.

Die Startdateien sind also eine Art Rezept für die Ausstattung des Computers beim Start. Sie sind Textdateien, die mit Hilfe eines Editors problemlos geändert werden können.

Default

Anders als man vermuten würde, sollte man die Startdateien besser ganz weglassen, wenn sie Fehler enthalten. Werden sie nicht registriert, während der PC startet, setzt DOS automatisch eine Reihe von Default-Werten ein als Ersatz für die fehlenden Angaben. Default-Werte sind Einstellungen, die von vornherein festgelegt sind und angewendet werden, 'wenn nichts anderes angegeben wird'.

Diese Werte reichen aus, um einen PC zu starten; in diesem Falle werden aber zum Beispiel keine deutschen Tastenbesetzungen verwendet, weil die Default-Werte in den USA festgelegt werden. Das bedeutet, daß etliche Zeichen anders auf deiner Tastatur plaziert werden.

Sei vorsichtig

Wenn du deine Startdateien änderst, ist das ungefähr so, wie wenn du die Motorhaube eines Autos öffnest und an seinen Geheimnissen herumfummelst. Wenn es gut geht, dann wird ein schläfriger Volkswagen zu einem potenten Roadster. Geht es aber daneben, kann es dir passieren, daß dein Computer gar nicht starten will. Ein verkehrtes Komma in einer Startdatei kann den Zugang zur Maschine voll blockieren!

Also solltest du zwei Sicherheitsregeln beachten, bevor du eine Startdatei änderst:

- 1) Erstelle eine Kopie deiner ursprünglichen Startdateien, bevor du sie änderst.

```
C:\>COPY CONFIG.SYS *.BAK
```

```
C:\>COPY AUTOEXEC.BAT *.BAK
```

Hier erhielten beide Kopien den Nachnamen `BAK`

- 2) Vergewissere dich, daß du eine Systemdiskette hast, von der aus der Computer starten kann.

Hältst du diese beiden Regeln ein, kannst du deine Änderungen jederzeit rückgängig machen. Tust du das nicht, riskierst du, daß dein Computer den Dienst verweigert, so daß dir ein erfahrener PC-Benutzer helfen muß, ihn wieder auf Trab zu bringen.

Eine gründliche Anleitung in der Anwendung von Startdateien findest du im Heft *Nutze Deinen PC optimal*.

Panik?

Wenn du deine Startdateien geändert hast, kann es dir passieren, daß der Computer nicht normal starten will. Die Startdateien, vor allem die CONFIG.SYS, sind äußerst empfindlich gegen kleine Fehler. Schlimmstenfalls kommt der Prozeß ohne jede Fehlermeldung zum Erliegen. Und dann sollte man cool bleiben und nicht in Panik geraten.

Du legst in diesem Fall schlicht deine Systemdiskette ins A-Laufwerk und versuchst es noch einmal. Geht alles gut, startet der Computer nun mit den Default-Werten. Die Konfiguration sieht anders aus, als du gewohnt bist, weil die Systemdiskette nicht über deine normalen Startdateien verfügt.

Die neue Konfiguration solltest du nur anwenden, bis du den Fehler behoben hast.

Sollte es tatsächlich einmal passieren, daß du in einer Startdatei einen Fehler gemacht hast und keine Systemdiskette in der Nähe hast, ist noch nicht alle Hoffnung verloren, wenn du nur DOS 6 anwendest. Diese Version verfügt über die angenehme Möglichkeit, während des Boots die Startdateien zu ignorieren. Wenn du den Text Starting MS-DOS siehst, drückst du die Taste **F5** – und schon werden deine Startdateien übersprungen.

Sobald du deinen Computer von einer Systemdiskette oder über die Taste **F5** gestartet hast, solltest du den Fehler in deinen Startdateien ausbessern, so daß deine normale Konfiguration wiederhergestellt wird, wenn du das nächste Mal startest.

Hast du Sicherheitskopien der Startdateien erstellt, zum Beispiel mit dem Nachnamen BAK, solltest du diese als neue Startdateien einsetzen:

```
C:\>COPY CONFIG.BAK *.SYS
```

```
C:\>COPY AUTOEXEC.BAK *.BAT
```

Nun kannst du deinen PC neu starten, indem du **Strg+Alt+Entf** tastest oder den Reset-Knopf drückst.

Die neuen Startdateien werden jetzt die ursprüngliche Konfiguration einsetzen.

Übersicht über die Dateien, die DOS anwendet

kann je nachdem weggelassen werden

KATEGORIE	DATEINAMEN	Path auf der Festplatte	Path auf der Diskette	ANMERKUNGEN
Systemdateien	IO.SYS MSDOS.SYS	C:\	A:\	Sind durch den Befehl SYS oder FORMAT /S auf die Platte zu überführen
Kommando- prozessor	COMMAND.COM	C:\	A:\	Wird automatisch mit den Systemdateien überführt
DOS-Dateien (große Anzahl)	EDIT.COM XCOPY.EXE ANSI.SYS ... und viele mehr	C:\DOS	A:\DOS	Externe DOS-Befehle u.a.m. Sind auf der Systemdiskette entbehrlich
Startdateien	CONFIG.SYS AUTOEXEC.BAT	C:\	A:\	Benutzergesteuerte Textdateien Sind entbehrlich (siehe Seite 46)

Eine volle Installation von DOS überführt alle vier Dateikategorien auf die Festplatte.

Eine Systemdiskette enthält als Minimum Systemdateien und den Kommandoprozessor (siehe Seite 31).

BAT-Dateien

Fällt es dir schwer, dir komplizierte DOS-Befehle zu merken, können dir BAT-Dateien nützlich sein.

BAT-Dateien sind gespeicherte DOS-Befehle.

Der Name `BAT` ist der Nachname derartiger Dateien; er ist eine Abkürzung des englischen Wortes *Batch*, das *'Stapel'* bedeutet. Man kann nämlich in einer BAT-Datei eine Reihe von Befehlen hintereinanderfügen.

Wird die Datei aktiviert, werden die Befehle schubweise einer nach dem anderen ausgeführt.

Eine BAT-Datei wird ebenso wie ein Programm aktiviert, indem du ihren Vornamen hinter den Prompt schreibst und Eingabe drückst.

Willst du zum Beispiel die Befehle speichern, die mit `XCOPY` Sicherheitskopien deiner Dateien erstellen (siehe Seite 31), kannst du die entsprechenden Befehle in eine BAT-Datei schreiben. BAT-Dateien sind reine Textdateien, die du am einfachsten mit dem Editor erstellst.

Der Nachname einer BAT-Datei muß `BAT` lauten; den Vornamen dagegen kannst du frei wählen. Der Einfachheit halber kannst du die nun zu erstellende Datei `SICH.BAT` nennen.

Wähle einen Vornamen, der leicht im Gedächtnis haftet und an die Funktion der BAT-Datei erinnert. Praktischerweise legst du sie in ein Verzeichnis, das nur für BAT-Dateien erstellt wurde, zum Beispiel `C:\BAT`.

Sorge dafür, daß sich dieses Verzeichnis in deinem Suchpfad befindet (siehe Seite 23), so daß du eine BAT-Datei nur durch Schreiben ihres Vornamens aktivieren kannst, egal in welchem Verzeichnis du dich befindest.

Nun erstellst du das neue Verzeichnis, wechselst dorthin über und startest den Editor mit Angabe des Dateinamens:

```
C:\>MD BAT
C:\>CD BAT
C:\BAT>EDIT SICH.BAT
```

Da du den Namen der zu erstellenden Datei angegeben hast, erscheint die Bezeichnung `SICH.BAT` am oberen Rand des Arbeitsfeldes. Nun schreibst du die Befehle, die die BAT-Datei enthalten soll:

```
A:
XCOPY /S/M C:\DATA\
C:
```

Du schreibst also genau das, was du beim Prompt schreiben würdest. Jede Zeile wird mit Eingabe abgeschlossen. Bist du fertig, muß die Datei gespeichert werden.

Da du in `C:\BAT` stehst und den Namen der Datei als `SICH.BAT` angegeben hast, drückst du einfach **Alt+D** und anschließend **B** und bestätigst, daß die Datei gespeichert werden soll.

Der Editor speichert sie auf der Festplatte.

Hast du `C:\BAT` in deinen Suchpfad eingefügt, kannst du jederzeit eine Sicherheitskopierung deiner Dokumente starten, indem du beim Prompt einfach `SICH` schreibst und Eingabe drückst – zum Beispiel:

```
C:\>SICH
```

Augenblicklich wird die `SICH.BAT` aktiviert und zeigt die gespeicherte Befehlszeile auf dem Bildschirm, während sie ausgeführt wird:

```
C:\>A:
A:\>XCOPY /S/M C:\DATA\
Einlesen der Quelldatei(en)...
C:\DATA\BRIEFEB\AASE.DOC
C:\DATA\BRIEFEB\HENRIK.DOC
C:\DATA\BRIEFEB\GEMEINDE.DOC
A:\>C:
C:\>
```

Die letzte Zeile der BAT-Datei hat uns zum Prompt auf der C-Platte zurückgebracht.

Wozu BAT-Dateien?

BAT-Dateien haben viele Vorteile.

- Sie sind leicht anzuwenden.
- Sie können komplizierte Befehle enthalten.
- Sie können mehrere Befehle hintereinander enthalten. Sobald eine Zeile ausgeführt ist, geht DOS automatisch zur nächsten weiter.

DOS hat eine Reihe von Befehlen, die ausschließlich für BAT-Dateien berechnet sind.

Mit Hilfe dieser Werkzeuge kannst du zum Beispiel eine BAT-Datei so einstellen, daß sie während ihres Arbeitsprozesses die Erfüllung bestimmter Bedingungen fordern kann.

AUTOEXEC.BAT

Die meistverwendete BAT-Datei ist AUTOEXEC.BAT.

Sie wird automatisch als letztes Glied des Boot-Prozesses durch DOS aktiviert.

Die AUTOEXEC.BAT ist nicht ganz so zartfühlend wie die CONFIG.SYS – dennoch solltest du die Regeln zur Änderung von Startdateien einhalten (siehe Seite 46).

Willst du zum Beispiel eine neue Zeile in die AUTOEXEC.BAT einfügen, öffnest du den Editor mit Angabe des Dateinamens:

```
C:\>EDIT AUTOEXEC.BAT
```

Nun erscheint der Inhalt der existierenden AUTOEXEC.BAT auf dem Bildschirm. Du kannst zum Beispiel lesen:

```
@ECHO OFF
PROMPT $p$g
PATH C:\;C:\DOS
MODE CON CP PREP=((850)EGA.CPI)
MODE CON CP SEL=850
KEYB GR,850,KEYBOARD.SYS
MOUSE
DOSKEY
```

Die drei Zeilen mit MODE und KEYB sind Befehle, die den deutschen Zeichensatz definieren. Du solltest sie nicht anrühren.

Nehmen wir nun an, daß du ein neues Verzeichnis C:\BAT dauerhaft in deinen Suchpfad einfügen willst.

Also änderst du die Zeile mit PATH, so daß sie folgendes Aussehen erhält:

```
PATH C:\;C:\DOS;C:\BAT
```

Dann speicherst du die AUTOEXEC.BAT unter dem gleichen Namen. Von nun an erscheint C:\BAT im Suchpfad, wenn du deinen PC startest.

Einfache BAT-Dateien

Ich habe einige BAT-Dateien erstellt, die deine tägliche Arbeit mit DOS erleichtern können. Ihr Inhalt mag dir an manchen Stellen geradezu chinesisch vorkommen, aber sie funktionieren. Hast du noch kein BAT-Verzeichnis, würde ich dir empfehlen, es wie oben beschrieben zu erstellen und in deinen Suchpfad einzufügen.

Die Beispiele erscheinen als dreigeteilte Boxen.

Der obere Teil enthält den Namen der jeweiligen BAT-Datei, den du benutzen mußt, wenn du den Editor in C:\BAT öffnest.

Der mittlere Teil der Box erläutert, wie die BAT-Datei funktioniert.

Unten siehst du ihren Inhalt. Diesen Inhalt schreibst du mit Hilfe des Editors genau so, wie du ihn hier liest. Hast du ihn geschrieben, speicherst du die BAT-Datei und beendest den Editor.

ZEIGE.BAT	
Zeigt den Inhalt einer Datei auf dem Bildschirm. mit jeweils einem Schirmbild. Schreibe ZEIGE und den Namen der Datei.	
TYPE %1	MORE

Diese BAT-Datei erspart es dir, jedesmal, wenn du den Inhalt einer Datei sehen willst, TYPE und MORE – getrennt durch den merkwürdigen senkrechten Strich – zu schreiben.

Indem du die BAT-Datei schreibst, schreibst du diesen Befehl ein für allemal – nur mit %1 statt des Dateinamens, den DOS nun selbst in den Befehl einfügt.

Das Zeichen | erscheint, wenn du die Alt Gr hältst, während du die Taste links vom Buchstaben Y drückst.

Um ZEIGE.BAT in dein Verzeichnis C:\BAT auf der Festplatte einzusetzen, schreibst du

```
C:\>CD BAT
C:\BAT>EDIT ZEIGE.BAT
```

Der Editor startet mit dem Dateinamen ZEIGE.BAT am oberen Rand des Bildschirms. Du schreibst nun den Inhalt der Datei in die große Fläche:

```
TYPE %1 | MORE
```

Dann verläßt du den Editor mit **Alt+D** plus einem **B** und bestätigst, daß die Datei gespeichert werden soll.

FF .BAT
Findet eine Datei mit einem gegebenen Namen auf der Platte. Schreibe FF und den Namen der Datei. WildCards können ein Bestandteil des Dateinamens sein.
CD \ DIR /S %1

Der Name **FF** wurde als Abkürzung für 'Find File' gewählt, außerdem hat er den Vorteil, kurz zu sein. Eigentlich läge der Name **FIND** näher – das geht aber nicht, weil er schon für einen DOS-Befehl verwendet wird, der Wörter in einer Textdatei findet.

Du erstellst eine **FF .BAT** im Verzeichnis **C:\BAT** mit dem Editor genau wie die **ZEIGE .BAT**

Diesmal besteht die Datei aus zwei Befehlen auf ihrer jeweiligen Zeile, die durch Eingabe voneinander getrennt werden.

Willst du zum Beispiel die Datei **SYS .COM** auf der Festplatte finden, schreibst du, egal in welchem Verzeichnis der Festplatte du stehst:

```
C:\>FF SYS.COM
```

Die **BAT**-Datei geht nun zunächst zum Stamm der Festplatte und durchsucht dann die gesamte Platte.

Auf dem Bildschirm erscheint eine Kurzübersicht der Verzeichnisse, die eine Datei namens **SYS .COM** enthalten.

Willst du eine Datei finden, kannst dich aber nur erinnern, daß ihr Name mit **KIM** beginnt, schreibst du

```
C:\>FF KIM*
```

und siehst zum Beispiel zwei Dateien, die zur Beschreibung passen:

```
Verzeichnis von C:\DATA\BRIEFE
KIM1  DOC      11,288
KIM2  DOC      13,104
   2 Datei(en)   24,392 Byte
```

Möchtest du Dateien im **A**-Laufwerk finden, deren Namen mit **KIM** beginnt, schreibst du

```
C:\>A:
A:\>FF KIM*
```

Installierung eines Programmes

Installierst du ein Programm, heißt das, daß du alle seine notwendigen Dateien an die richtigen Stellen setzt, so daß das Programm sie finden kann.

Es gibt große Unterschiede bei der Installierung von Programmen auf der Festplatte. Kaufst du ein Programm, wird in der Regel ein ausführliches, gedrucktes Handbuch mitgeliefert, das die nötigen Anweisungen zur Installierung gibt. Im allgemeinen erfolgt sie, sobald du eine Datei namens **INSTALL** oder **SETUP** im **A**-Laufwerk aktivierst:

```
C:\>A:
A:\>INSTALL
```

Es gibt aber viele Programme, die du dir beschaffen kannst, ohne sie wie üblich zu kaufen. Diese Programme werden nur selten mit einem Handbuch geliefert. Installierungs- und Gebrauchsanweisungen für solche Programme erscheinen oft als Textdateien neben den Programmen. Das gilt zum Beispiel für Shareware-Programme.

Freeware und Shareware

Freeware- und Shareware-Programme können frei von jedermann kopiert und weitergegeben werden – einzige Einschränkung ist das Urheberrecht ihres Verfassers. Das heißt, daß niemand durch ihre Kopierung und Weitergabe Geld verdienen kann.

Ein Freeware-Programm kannst du kostenlos benutzen. Ein Shareware-Programm kannst du eine Zeitlang kostenlos ausprobieren. Beschließt du danach, es permanent zu verwenden, mußt du dich registrieren lassen, indem du dem Verfasser einen Geldbetrag schickst.

Diese Vorgangsweise hat den Vorteil, daß die Verfasser von Shareware ihre Produkte ohne Unkosten in Umlauf bringen können. Ein gutes Programm kann sich in kurzer Zeit über die ganze Erde verbreiten. Ob die Verfasser Einnahmen aus ihrer Arbeit erzielen, hängt allerdings ganz und gar von der Zufriedenheit und Ehrlichkeit der Benutzer ab.

Ein besonderer Programmtyp sind Public Domain Programme, für die kein Urheberrecht besteht.

Public Domain-, Freeware- und Shareware-Programme findest du in vielen Bibliotheken, in manchen Computergeschäften und bei Firmen, die solche Programme anbieten. Die Verfasser bedingen sich zwar, daß der Zwischenhandel an ihren Produkten nicht verdienen darf – seine Unkosten darf er aber durch eine geringe Gebühr decken.

Ein Shareware-Programm enthält vielfach folgendes:

- Dateien mit dem eigentlichen Programm
- Dateien zur Installierung des Programmes
- Dateien mit einer Gebrauchsanweisung
- Dateien mit Angaben zur Bezahlung

Versuch's einfach

Hast du eine neue Diskette mit Programmen erworben, legst du sie zunächst ins A-Laufwerk und schreibst

```
C:\>A:
```

Dann siehst du dir ihren Inhalt an:

```
A:\>DIR /P
```

Falls die Diskette mehrere Programme enthält, können sie in verschiedenen Verzeichnissen liegen – von hier aus können sie an verschiedene Stellen der Festplatte kopiert werden.

Zunächst wollen wir aber davon ausgehen, daß alle Dateien im Stammverzeichnis von A: liegen.

Nehmen wir nun an, daß du das Glück hast, eine Datei namens `INSTALL.EXE` oder `SETUP.EXE` zu finden. Das ist ein Programm, das dafür sorgt, daß die Dateien auf die Festplatte kopiert werden. Du startest `INSTALL.EXE`, indem du schreibst

```
A:\>INSTALL
```

Meistens wird dieser Befehl alles übrige von selbst erledigen, auch die Kopierung von A: nach C:

Installationsprogramme werden dich oft um den Namen des Verzeichnisses bitten, in welches das Programm kopiert werden soll.

Gibt es auf der Diskette keine `INSTALL.EXE` oder `SETUP.EXE`, lohnt es sich, auf der Festplatte ein neues Verzeichnis zu erstellen, in das du alle Dateien kopieren kannst.

Ist das neue Programm ein Spiel namens `FRAC`, erstellst du natürlich ein Verzeichnis (siehe Seite 20) namens `C:\SPIELE\FRAC`.

Anschließend machst du es zu deinem aktuellen Verzeichnis, in das du nun nach der Empfangsmethode alle Dateien der Diskette im A-Laufwerk kopieren kannst:

```
C:\SPIELE\FRAC>COPY A:\.
```

Nach dem Kopieren befinden sich die Dateien des Programms auf der Festplatte, und die Diskette kann aus dem A-Laufwerk genommen werden.

Nun siehst du dir die Dateien genauer an, um festzustellen, wo du Hilfe finden kannst:

```
C:\SPIELE\FRAC>DIR /P
```

Zunächst solltest du nach einer Datei namens `README` oder so schauen, die dir sagen wird, was du tun sollst. Findest du zum Beispiel eine Datei namens `README.TXT`, siehst du ihren Inhalt, wenn du schreibst

```
C:\SPIELE\FRAC>ZEIGE README.TXT
```

ZEIGE ist ein Befehl, den du selbst als BAT-Datei gespeichert hast (siehe Seite 49). Er entspricht dem Befehl **TYPE | MORE**, der den Inhalt der Datei Seite für Seite zeigt.

Oder du kannst die Datei drucken, indem du den Drucker einschaltest und schreibst

```
C:\SPIELE\FRAC>PRINT README.TXT
```

Dateien mit dem Namen `README` sind in der Regel kurze Erläuterungen, die als erstes gelesen werden sollten. Findest du keine `README`-Datei, solltest du nach einer Datei mit dem Nachnamen `DOC` sehen – sie wird oft gründlichere Anweisungen enthalten.

Gibt es zum Beispiel eine Datei namens `MANUAL.DOC`, wird sie vermutlich ein komplettes Handbuch für das Programm enthalten.

Du schreibst

```
C:\SPIELE\FRAC>ZEIGE MANUAL.DOC
```

Hast du die ersten fünf der 43 im Handbuch enthaltenen Seiten gelesen und möchtest bis auf weiteres die gründliche Einführung zugute haben, kannst du die `ZEIGE.BAT` unterbrechen, indem du **Strg+C** drückst.

Unterbrichst du eine `BAT`-Datei durch **Strg+C**, stellt dir DOS die Frage
Stapelverarbeitung abbrechen
(`J/N`)?
Antwortest du `J`, kehrst du zum Prompt zurück.

Allmählich sind wir soweit, daß du die Dateien mit dem Nachnamen `BAT`, `COM` oder `EXE` ausprobieren kannst. Eine dieser Dateien wird vermutlich das Programm starten, das du auf die Festplatte kopiert hast.

Du findest zum Beispiel eine Datei namens `FRAC.EXE` und schreibst

```
C:\SPIELE\FRAC>FRAC
```

Nun beginnst das Spiel.

Automatischer Start eines Programms

Du kannst eine `BAT`-Datei erstellen, die `FRAC.EXE` aktiviert, egal in welchem Verzeichnis du stehst. Du startest den Editor in `C:\BAT`

```
C:\>CD BAT  
C:\BAT>EDIT FRAC.BAT
```

und schreibst folgende Zeilen:

```
C:  
CD \SPIELE\FRAC  
FRAC  
CD \
```

Über **Speichern** wird die Datei als `FRAC.BAT` in `C:\BAT`, das im Suchpfad liegt, gespeichert.

Ohne Rücksicht auf dein aktuelles Verzeichnis kannst du nun schreiben

```
C:\>FRAC
```

worauf die `BAT`-Datei das Spiel startet.

Komprimierte Dateien

Wenn du eine Diskette mit Programmen erhältst, sind ihre Dateien oft komprimiert, um Platz zu sparen.

Enthält eine Datei eine Folge von mehreren Zeichen, die oft wiederholt wird, kann diese Folge durch eine Kodierung ersetzt werden, die wesentlich weniger Raum einnimmt. Je nach dem Inhalt der Datei kann sie auf 25 bis 50% ihrer ursprünglichen Größe verringert werden.

Vielfach werden mehrere Dateien in ein gesammeltes Paket komprimiert, ein sogenanntes Archiv. Ein solches Archiv erscheint als einzelne Datei; nach dem Entpacken siehst du die originalen Dateien in ihrer ursprünglichen Größe.

Während dieses Öffnungsprozesses wird eine Kopie der Dateien erstellt, die sich im Archiv befinden. Findet der Prozeß auf der Festplatte statt, muß dort Platz genug sein für das Archiv und für die entpackten Dateien. Diese können zwei- bis dreimal soviel Platz einnehmen wie das Archiv; in der Regel gibt es auf der Diskette, die es enthält, nicht genügend Platz. Am besten kopierst du das Archiv in ein Verzeichnis der Festplatte, das du für diesen Zweck eingerichtet hast, und entpackst es dort.

Willst du dich vergewissern, daß auf deiner Festplatte genug freier Platz ist, erstellst du mit `DIR` eine Übersicht. Ihre letzte Zeile teilt mit, wieviele freie Bytes sich auf der Platte befinden.

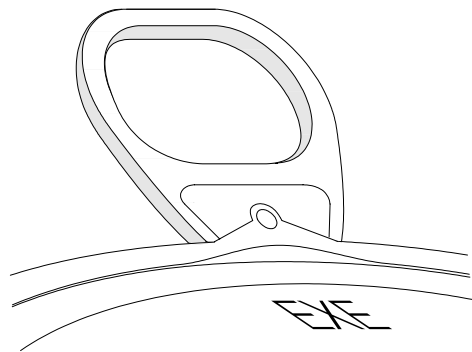
Selbstentpacken

Ein Archiv mit komprimierten Dateien wird durch ein Kompressionsprogramm erstellt, das für das Packen und Entpacken sorgt. Willst du die Dateien entpacken, mußst du also ein Programm benutzen, das dem entspricht, welches die Kompression vornahm. Es ist aber nicht sicher, ob du gerade dieses Programm hast! Darum hat man *selbstentpackende Archive* entwickelt, die zum Entpacken kein besonderes Programm erfordern. Das selbstentpackende Archiv ist in das Dekompressionsprogramm eingebaut. Sobald du den Vornamen des Archivs schreibst, entfaltet sich der Inhalt automatisch. Die selbstentpackenden Archive haben den Nachnamen `EXE`.

Begegnet dir eine Datei mit dem Nachnamen EXE, erkennst du nicht unmittelbar, ob sie ein normales Programm ist, zum Beispiel ein Spiel, oder ein selbstentpackendes Archiv.

Erscheinen keine weiteren Dateien neben der EXE-Datei, handelt es sich höchstwahrscheinlich um ein selbstentpackendes Archiv, da die meisten Programme mehr als eine einzelne Datei benötigen.

In jedem Fall kannst du das Programm untersuchen, indem du den Vornamen der Datei schreibst – allerdings möglichst auf deiner Festplatte, wo es den meisten Platz gibt.



Nehmen wir an, daß du eine Diskette mit einem Spiel bekommst, das aus einer einzelnen Datei namens EUROPAK.EXE besteht.

Du bist dir ziemlich sicher, daß die Datei ein selbstentpackendes Archiv ist, und kopierst sie in ein neues Verzeichnis

```
C:\SPIELE\EURO>COPY A:EUROPAK.EXE
```

Nun aktivierst du die Datei, indem du schreibst

```
C:\SPIELE\EURO>EUROPAK
```

EUROPAK.EXE erweist sich wie erwartet als Archiv und beginnt damit, seinen Inhalt zu entpacken. Auf dem Bildschirm siehst du während der Dekomprimierung den Namen des Programmes samt den Namen der einzelnen Dateien:

```
PKSFX (R) FAST!
Self Extract Utility

Searching EXE:
C:/SPIELE/EURO/EUROPAK.EXE
  Exploding: EURO.001
  Extracting: EURO.BAT
  Exploding: INSTALL.BAT
  Exploding: EURO.DOI
  Extracting: EUROPA.EXE
  Extracting: INSTALL.EXE
```

Nun erstellst du eine Übersicht:

```
C:\SPIELE\EURO>DIR
```

```
Datenträger in Laufwerk C: ist
STEEN
```

```
Verzeichnis von C:\SPIELE\EURO
```

```
.          <DIR>
..         <DIR>
EUROPAK   EXE      582,872
EURO      001      907,684
EURO      BAT        21
INSTALL   BAT      324
EURO      DOI      4,378
EUROPA    EXE      82,052
INSTALL   EXE     11,679
```

```
9 Datei(en)
1,589,010 Byte
90,497,024 Byte frei
```

Wie du siehst, enthält das aktuelle Verzeichnis nun sowohl das Archiv EUROPAK.EXE als auch seinen dekomprimierten Inhalt.

Das Archiv füllt ein gutes halbes MB, während die entpackten Dateien etwas über 1 MB einnehmen. Hättest du die Dekomprimierung auf der Diskette vorgenommen, hätten beide zusammen keinen Platz gefunden.

Die Kopie des Archivs auf deiner Festplatte hat ihren Inhalt abgeliefert und kann nun gelöscht werden:

```
C:\SPIELE\EURO>DEL EUROPAK.EXE
```

Wie du siehst, haben zwei der entpackten Dateien den Vornamen INSTALL.

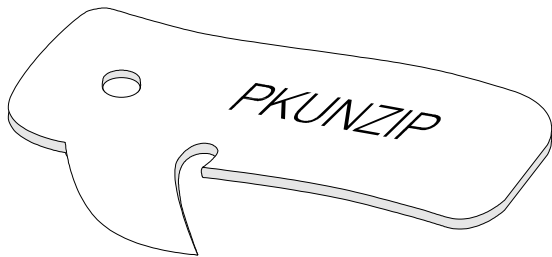
Du schreibst:

```
C:\SPIELE\EURO>INSTALL
```

und erhältst nun die Anweisungen, die dich weiterbringen.

Komprimierungsprogramme

Begegnet dir ein Archiv, das nicht selbstentpackend ist, mußt du das Programm benutzen, das es erstellt hat, um die Dateien zu erhalten. Bisweilen liegt die notwendige Programmdatei neben dem Archiv – das ist aber leider nicht immer der Fall. Zum Glück sind die üblichsten Kompressionsprogramme ziemlich verbreitet; also sollte es dir leicht möglich sein, das entsprechende Programm zu besorgen.



Ich werde drei der meistverwendeten Kompressionsprogramme erwähnen: PKZIP, ARJ und LHA – alle drei sind als Shareware erhältlich. Zunächst solltest du aber ein spezielles Verzeichnis für deine Kompressionsprogramme erstellen.

Ein PAK-Verzeichnis erstellen

Ich kann dir nur empfehlen, ein spezielles Verzeichnis für Kompressionsprogramme einzurichten und es in deinen Suchpfad einzufügen. Auf diese Weise hast du jederzeit freien Zugriff auf diese Programme, egal wo auf der Festplatte du Dateien packst oder entpackst. Erstelle nun `C:\PAK`

```
C:\>MD PAK
```

Wenn du ein Kompressionsprogramm auf einer Diskette erhältst, wird es oft mehrere Typen von Dateien enthalten – der wichtigste hat den Nachnamen EXE.

Die EXE-Dateien erledigen die eigentliche Arbeit, während die übrigen Dateien Gebrauchsanweisungen, Handbücher und andere Ratschläge enthalten.

In die `C:\PAK` solltest du nur die EXE-Dateien kopieren. Die übrigen Dateien können auf der Diskette bleiben, um Platz auf der Festplatte zu sparen.

Fügst du `C:\PAK` in deinen Suchpfad ein, untersuchst du zunächst, welche Verzeichnisse der Suchpfad bereits enthält, zum Beispiel

```
C:\>PATH
```

```
PATH=C:\;C:\DOS
```

Dann fügst du `C:\PAK` ein, indem du schreibst

```
C:\>PATH C:\;C:\DOS;C:\PAK
```

Von nun an ist `C:\PAK` im Suchpfad enthalten – allerdings nur, während der Computer eingeschaltet ist. Willst du diesen Suchpfad dauerhaft machen, so daß er bei jedem Start gilt, muß deine `AUTOEXEC.BAT` folgende Zeile enthalten:

```
PATH C:\;C:\DOS;C:\PAK
```

Du startest den Editor über die `AUTOEXEC.BAT`

```
C:\>EDIT AUTOEXEC.BAT
```

Enthält die `AUTOEXEC.BAT` ohnehin eine Zeile mit dem Wort `PATH`, kannst du dich damit begnügen, `;C:\PAK` hinzuzufügen – andernfalls mußt du die gesamte Zeile schreiben.

Zum Abschluß speicherst du die geänderte `AUTOEXEC.BAT` unter ihrem ursprünglichen Namen.

Vergiß nicht, daß eine Änderung in der `AUTOEXEC.BAT` erst nach einem Neustart funktioniert. Du kannst gegebenenfalls über `Strg+Alt+Entf` neustarten.

Nun ist dein Verzeichnis `C:\PAK` bereit, Kopien der EXE-Dateien aus deinen Kompressionsprogrammen entgegenzunehmen, wenn du sie erhältst. Dank dieser Vorbereitungen kannst du die Befehle des Kompressionsprogrammes anwenden, egal in welchem Verzeichnis auf der Festplatte du dich befindest.

PKZIP

Eines der meistverbreiteten Kompressionsprogramme ist PKZIP. Seine Archive sind an ihrem Nachnamen ZIP zu erkennen.

Die ersten beiden Buchstaben, PK, sind die Anfangsbuchstaben des Namens seines Verfassers, Phil Katz. Zip ist das englische Wort für Reißverschluß.

Die meisten Programme verwenden dieselbe EXE-Datei zum packen und entpacken. So ist es aber nicht mit PKZIP: zum packen benutzt es PKZIP.EXE und zum entpacken PKUNZIP.EXE.

Nehmen wir an, daß du eine Diskette mit diesem Kompressionsprogramm in den Fingern hast.

Du erstellst eine Übersicht über den Inhalt der Diskette:

```
C:\> DIR A:
```

```
Datenträger in Laufwerk A hat  
keine Datenträgerbezeichnung
```

```
Verzeichnis von A:\
```

```
WHATSNEW 204          2,430  
OMBUDSMN ASP          591  
PKUNZJR  COM          2,750  
TREE     COM          414  
ADDENDUM DOC         19,361  
LICENSE  DOC          3,707  
MANUAL   DOC         202,252  
ORDER    DOC          3,304  
README   DOC          741  
SHAREWAR DOC          573  
PKUNZIP  EXE          29,378  
PKZIP    EXE          42,166  
PKZIPFIX EXE          7,687  
ZIP2EXE  EXE          27,319  
AUTHVERI FRM          2,330  
V204G    NEW          10,704
```

```
16 Datei(en) 355,707 Byte  
1,097,216 Byte frei
```

In dieser Ausgabe von PKZIP liegen sechzehn Dateien; vier von ihnen sind Programme mit dem Nachnamen EXE.

Nun wechselst du in das Verzeichnis C:\PAK über und kopierst die EXE-Dateien mit der Empfangsmethode:

```
C:\>CD PAK  
C:\PAK>COPY A:*.EXE
```

Jetzt liegen sie am rechten Ort und können überall auf der Festplatte benutzt werden, wenn du ihren Vornamen schreibst.

Ihre Funktionen sind folgende:

PKZIP.EXE	erstellt ein Archiv
PKUNZIP.EXE	entpackt ein Archiv
ZIP2EXE.EXE	macht ein existierendes Archiv selbstentpackend
PKZIPFIX.EXE	repariert ein zerstörtes Archiv

Ein PKZIP-Archiv entpacken

Nehmen wir an, daß du eine Diskette mit einem neuen Programm hast, einem sehr schnell arbeitenden Nachfolger von DISKCOPY (siehe Seite 41).

Das Programm heißt Disk Copy Fast Version 3.2 und existiert als einzelne Datei A:\DCF32.ZIP.

Du hast bereits ein gemeinsames Verzeichnis für Utilities – das heißt nützliche Werkzeuge – und möchtest Disk Copy Fast in einem Unterverzeichnis namens C:\UTIL\DCF speichern.

Also erstellst du das neue Verzeichnis und wechselst dorthin über (siehe auch Seite 20). Dann kopierst du das Archiv in dein neues Verzeichnis:

```
C:\UTIL\DCF>COPY A:DCF32.ZIP
```

Du beginnst den Prozeß des Entpackens, indem du PKUNZIP schreibst und dann den Vornamen des Archivs.

```
C:\UTIL\DCF>PKUNZIP DCF32
```

In diesem Fall erscheinen im Archiv zwölf Dateien. Das neue Verzeichnis enthält diese Dateien und das Archiv, das du nun aus der Festplatte löschen kannst:

```
C:\UTIL\DCF>DEL DCF32.ZIP
```

Eine der entpackten Dateien heißt READ.ME und enthält vielleicht nützliche Informationen. Du siehst dir ihren Inhalt an:

```
C:\UTIL\DCF>ZEIGE READ.ME
```

Der Text besagt, daß du das Handbuch (eine Datei namens DCF32.DOC) mindestens einmal lesen solltest, bevor du eine wichtige Diskette mit diesem Programm kopierst.

Also drückst du das Handbuch `DCF32.DOC` indem du den Drucker einschaltest und schreibst

```
C:\UTIL\DCF>PRINT DCF32.DOC
```

Name des Ausgabegeräts [PRN]:

Du drückst Eingabe, um PRN zu bestätigen. Nun wird ein 26 Seiten umfassendes Handbuch gedruckt.

Wenn du es gelesen hast, startest du das Programm:

```
C:\UTIL\DCF>DCF
```

Ein ZIP-Archiv erstellen

Nehmen wir an, daß du in `C:\VIKING` ein Spiel hast, das aus vier Dateien besteht und insgesamt 1,7 MB füllt. Dieses Spiel möchtest du gern deinem Vetter schicken, der, wie du weißt, die Datei `PKUNZIP.EXE` hat.

Selbst hast du die `PKZIP.EXE` im Verzeichnis `C:\PAK` – obendrein auf deinem Suchpfad – also beschließt du, alle Dateien in ein neues Archiv `ROLLO.ZIP` auf eine Diskette zu komprimieren.

Du setzt eine neue Diskette ins A-Laufwerk, wechselst ins Verzeichnis `C:\VIKING` über und schreibst

```
C:\VIKING>PKZIP A:ROLLO *.*
```

Das Archiv `ROLLO.ZIP` wird nun auf die Diskette geschrieben, wo es 1,2 MB füllt.

Der Befehl hat drei Abteilungen::

PKZIP	Vorname des Kompressionsprogrammes
A:ROLLO	Laufwerk und Vorname des Archivs
.	Die Dateien, die komprimieren werden sollen

Enthält die letzte Abteilung alle Dateien des aktuellen Verzeichnisses, wie in unserem Beispiel, kann sie weggelassen werden.

Möchtest du eine kurze Übersicht der Befehle in `PKZIP` sehen, schreibst du

```
C:\>PKZIP /?
```

Die Befehle sind nicht einfach zu merken – also lohnt es sich, das Handbuch durchzusehen.

ARJ

Der Name des populären Kompressionsprogrammes `ARJ` setzt sich aus den ersten Buchstaben von `Archive` und `Robert Jung` zusammen, dem Namen des Verfassers.

Der Nachname der `Archive` dieses Programmes ist `ARJ`.

Zum Packen und Entpacken wird die Datei `ARJ.EXE` verwendet, die du ins Verzeichnis `C:\PAK` kopieren solltest.

Ein ARJ-Archiv entpacken

Du hast das Archiv `A:\BANNER.ARJ` erhalten und möchtest es nun entpacken.

Sein Nachname besagt, daß es ein `ARJ`-Archiv ist.

Du hast etwas über das Programm `Banner` gehört und willst es nun als Utility in ein Verzeichnis namens `C:\UTIL\BAN` legen. Also erstellst du dieses Verzeichnis, wie auf Seite 20 gezeigt, und wechselst dorthin über.

Dann kopierst du das Archiv:

```
C:\UTIL\BAN>COPY A:BANNER.ARJ
```

Das Entpacken erfolgt, indem du `ARJ`, einen Buchstaben (einen Befehl) und dann den Vornamen des Archivs schreibst.

```
C:\UTIL\BAN>ARJ E BANNER
```

Der Buchstabe **E** (`Extract`=herausziehen) gibt an, daß Dateien entpackt werden sollen. Der Buchstabe **X** (`eXtract`) bedeutet dasselbe, nur werden hier die Dateien mit ihrem ursprünglichen Pfad entpackt. Letzteres ist manchmal ein Vorteil – es kann aber auch dazu führen, daß die Dateien in einem anderen Verzeichnis landen als dem aktuellen.

In diesem Fall gibt das Archiv vierundzwanzig Dateien von sich, wonach es gelöscht werden kann:

```
C:\UTIL\BAN>DEL BANNER.ARJ
```

Es zeigt sich, daß das Paket weder README- noch DOC-Dateien enthält.

Also mußt du das Programm starten, um zu sehen, wie es funktioniert:

```
C:\UTIL\BAN>BANNER
```

Ein ARJ-Archiv erstellen

Nehmen wir an, daß du ein Virusprogramm auf einer Diskette speichern willst, auf der dir nur 250 KB zur Verfügung stehen. Das Programm besteht aus sechs Dateien in C:\VIR, die insgesamt 361 KB füllen.

Also entscheidest du dich, das neue Archiv SCAN.ARJ mit dem Programm ARJ.EXE zu erstellen, das in deinem Suchpfad liegt.

Du legst die Diskette ins A-Laufwerk, wechselst in C:\VIR über und schreibst

```
C:\VIR>ARJ A A:SCAN
```

und schon wird das Archiv erstellt. Es füllt 212 KB.

Der Buchstabe **A** ist der Befehl zur Archivierung. Außer den Befehlen hat ARJ zahlreiche Schalter, die mit einem Strich vorne geschrieben werden.

Willst du eine kurze Übersicht über die Befehle in ARJ sehen, schreibst du

```
C:\>ARJ /?
```

Auch diese Befehle haften nicht leicht im Gedächtnis. Es lohnt sich, das Handbuch durchzulesen.

LHA

Schließlich wollen wir das Kompressionsprogramm LHA erwähnen, das von dem Japaner Haruyasu Yoshizaki verfaßt wurde. Seine Archive haben einen anderen Nachnamen als das Programm selbst, nämlich LZH.

Zum Packen und Entpacken wird die Datei LHA.EXE verwendet.

Wie bei PKZIP und ARJ kannst du die Programmdatei LHA.EXE ins Verzeichnis C:\PAK kopieren, um sie in den Suchpfad einzufügen.

Ein LHA-Archiv entpacken

Nehmen wir an, du erhältst eine Diskette mit Graphikdateien, die unter dem Namen A:\IMAGES.LZH archiviert wurden.

Diese Dateien möchtest du im Verzeichnis C:\DATA\GRAPHIK speichern; also kopierst du das Archiv dorthin.

Anschließend beginnst du das Entpacken, indem du schreibst

```
C:\DATA\GRAPHIK>LHA E IMAGES
```

Die Befehle in LHA erinnern sehr an die in ARJ. Der Buchstabe **E** steht für Extract.

Sind die Bilddateien entpackt, kannst du das Archiv von der Festplatte löschen:

```
C:\DATA\GRAPHIK>DEL IMAGES.LZH
```

Ein LHA-Archiv erstellen

Das ARJ-Archiv in der voranstehenden Spalte hättest du im großen und ganzen genauso über LHA erstellen können. Der Befehl würde lauten

```
C:\VIR>LHA A A:SCAN
```

Ein Schalter wird in LHA mit einem Schrägstrich voran geschrieben, wie bei DOS-Befehlen.

Willst du eine kurze Übersicht der Befehle in LHA sehen, schreibst du

```
C:\>LHA /?
```

Virus

Ein Computervirus ist ein kleines Programm, das sich in deine Festplatte einschleicht und dort Schlimmes anstellt.

Ein Virus kann sich aus eigener Kraft vermehren. Er ist "ansteckend" und kann sich von einem Computer zum anderen ausbreiten.

Ein Virus schließt sich als blinder Passagier anderen Programmen an. Kopierst du ein infiziertes Programm, folgt der Virus mit. Ein Virus kann aber auch ohne Kopierung überführt werden: startest du ein infiziertes Programm, wird seine Kopie in den Speicher eingelesen, und der Virus kann sich von hier aus verbreiten.

Ein Virus wird entdeckt, wenn in deinem PC unerwartete Dinge vor sich gehen.

Das können zum Beispiel unerklärliche Fehler sein oder eine merkbare Verlangsamung seiner Geschwindigkeit. Die harmloseren Arten von Viren verraten selber ihre Anwesenheit, indem sie zum Beispiel eine Mitteilung auf den Bildschirm schreiben. Die gefährlicheren Typen löschen ohne Warnung Programme und Daten.

Es ist unbedingt empfehlenswert, Virusangriffen durch ein Antivirusprogramm vorzubeugen, das die gesamte Festplatte oder eine Diskette auf Viren untersucht.

Untersuche deine Festplatte ein für allemal mit einem Antivirusprogramm.
Untersuche ebenfalls jede Diskette, wenn du sie das erste Mal ins A-Laufwerk legst.

Die Untersuchung einer Diskette dauert höchstens zwei Minuten. Selbst offizielle Programme können infiziert sein – du solltest aber vor allem bei Disketten mit Programmen unbekannter Herkunft vorsichtig sein.

Entdeckst du einen Virus auf einer Diskette, kannst du ihn ausrotten, indem du sie mit folgendem Befehl formatierst:

```
C:\>FORMAT A: /U
```

/U bedeutet unbedingt – die Formatierung kann nicht rückgängig gemacht werden. Während einer Formatierung kann ein Virus nicht anstecken.

Ist ein Programm auf der Festplatte infiziert, kannst du dich in der Regel damit begnügen, es zu löschen. Ein Virus kann sich aber auch in einem Bereich der Festplatte verstecken, wo er nicht so leicht entfernt werden kann. Hier mußt du dich nach den Anweisungen des Antivirusprogramms richten. Schlimmstenfalls kann es dir passieren, daß du einen Backup aller nicht infizierten Dateien machen und anschließend die Festplatte formatieren mußt. Das dauert lange und kommt zum Glück nur selten vor.

Es gibt viele recht gute Antivirusprogramme, die jeweils ihre eigenen Vorzüge haben.

DOS 6 wird mit zwei Antivirusprogrammen ausgeliefert. Das eine, MWAV . EXE , ist nur unter Windows zu verwenden; das andere, MSAV . EXE, ist dagegen ein DOS-basiertes Programm.

Willst du Microsoft AntiVirus starten, schreibst du

```
C:\>MSAV
```

worauf das Menü des Programms erscheint. Willst du deine gesamte Festplatte auf Viren scannen, drückst du **F4**. Willst du ein anderes Laufwerk scannen, zum Beispiel eine Diskette im A-Laufwerk, drückst du **F2**.

Unter dem Punkt **Options (F8)** kannst du das Programm auf deine Bedürfnisse zuschneiden.

Über die Hilfstaste **F1** erhältst du eine Erklärung der verschiedenen Möglichkeiten.

Ich würde dir empfehlen, dir das Shareware-Programm Viruscan von McAfee zu besorgen. Meine Version dieses Programmes sucht nach 2738 Virustypen und wird laufend aktualisiert.

Leere Batterie

Zu den unangenehmeren Problemen, die du mit deinem PC erleben kannst, gehört, daß die Batterie leerläuft.

Bei einer stationären Maschine kommt das selten vor; transportable Computer erleben das aber öfter. Die eingebaute Batterie liefert Strom für einen Microchip im PC namens *CMOS*, der das eingebaute Programm BIOS (siehe Seite 9) enthält. Während der Computer ausgeschaltet ist, sind seine eigenen technischen Spezifikationen im BIOS gespeichert, das außerdem Datum und Zeit registriert. Darum ist es ein sicheres Zeichen für eine leere Batterie, wenn der PC das Datum als 1. Januar 1980 angibt; frühere Daten begreift DOS nicht. In der Computerwelt ist das Altertum nicht gerade weit weg!

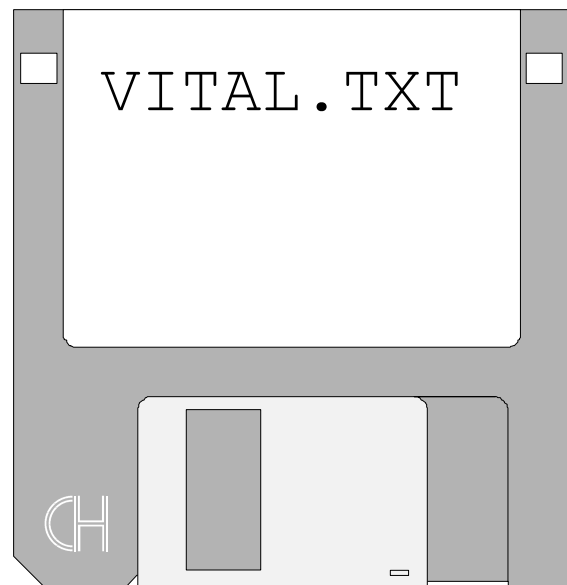
Der Verlust von Datum und Uhrzeit läßt sich verschmerzen. Es dauert nur wenige Sekunden, die richtigen Werte über *DATE* und *TIME* einzugeben (siehe Seite 9).

Schlimmer sieht es aus mit einer Reihe von anderen BIOS-Angaben, die der Computer für seine Arbeit benötigt. Hast du diese Angaben nicht gespeichert, kann ein Neustart des Computers problematisch werden. Unter Bekannten hilft man sich oft, indem man verschiedene Leute anruft, bis man jemanden mit einem ähnlichen Computer findet. Klappt das, versucht man es mit den Werten seiner Maschine.

Das sicherste ist aber, die Daten auf eine Diskette zu kopieren, bevor etwas schief geht.

Es gibt mehrere Werkzeugprogramme, die über diese Funktion verfügen. Eines der bekanntesten ist *Norton Utilities* – es enthält die Möglichkeit, diese lebenswichtigen Angaben auf einer sogenannten *Rescue*-Diskette zu speichern.

Etliche dieser Angaben können aber vom Programm *Microsoft Diagnostics* gehandhabt werden, das mit Windows 3.1 und DOS 6 geliefert wird. Dieses Programm besteht aus den Dateien *MSD.COM* und *MSD.EXE*, die im selben Verzeichnis liegen müssen.



Willst du eine Kopie der lebenswichtigen Angaben auf eine Diskette setzen, tust du folgendes:

Du setzt eine leere, formatierte Diskette ins A-Laufwerk. Dann findest du das Verzeichnis, das die MSD-Dateien enthält:

```
C:\>DIR /S MSD.*
```

Du wechselst in dieses Verzeichnis über, etwa in *C:\DOS*, und aktivierst *MSD* mit folgendem Befehl:

```
C:\DOS>MSD /P A:VITAL.TXT
```

Der Schalter */P* sorgt dafür, daß zahlreiche wichtige Angaben automatisch in die Datei überführt werden, die du am Ende des Befehls angegeben hast, also in *A:\VITAL.TXT*

Auf dem Bildschirm siehst du die Seitenzahlen deiner Angaben, während sie kopiert werden.

Stehst du wieder im Prompt, nimmst du die Diskette aus dem Laufwerk und benennst sie *VITAL.TXT*

Eine solche Datei ist Gold wert für einen erfahrenen DOS-Benutzer, eventuell einen Händler, der dir helfen soll, wenn deine Batterie eines Tages den Geist aufgibt.

Stichwörter

%1 48
* 20, 27
*. * 20, 27
/M 32
/P 20
/S 21, 30
/W 20
<DIR> 16, 18
^C 29

102 Tasten 6
14-Zoll-Monitor 6
256 Zeichen 44
300 dpi 7

A: 17
A-Attribut 31
Abbrechen des Prozesses 29
Adresse 14
aktuelles Verzeichnis 18
alle möglichen Dateien 20
alle möglichen Zeichen 20
Alt Gr-Taste 6, 18
Alt-Taste 24, 44
amerikanische Tastatur 45
ANSI-Zeichensatz 44
anstecken 57
Antivirusprogramm 57
Anweisungen 25
Archiv 51
Archiv-Attribut 31
ARJ 55
ASCII-Wert 24, 44
ASCII-Zeichensatz 44
ATTRIB 31
Attribut 31
Auflösung 6
augenblicklicher Suchpfad 22
ausdrucken Schirmbild 25
ausgeschaltetes Attribut 31
auslassen den Pathnamen 26
ausschneiden Text 35
AUTOEXEC.BAT 45, 46, 48, 53
automatisches Auspacken 51

B: 17
Backslash 17, 18, 22, 44
BAK 45
BAT 23
BAT-Dateien 23, 47, 51
bearbeiten 35
Befehl wiederholen 10
Befehlszeile 9

Betriebssystem 4
Bezeichnung 40
Bildschirm 6
Bildschirmkarte 5, 6
Bill Gates 4
binär 16
BIOS 8, 9
Boot 8, 24, 48
Boot-Diskette 41
Bytes 16

C:\DOS 30
C: 17
CD 17, 18
CDROM-Laufwerk 5
Change Directory 17
CLS 10, 16, 30
CMOS 9, 58
CODEPAGE 48
COM 23
COMMAND.COM 16, 30, 42, 46
CONFIG.SYS 41, 42, 45, 46, 48
Controllerkarte 5
COPY 26
CPU 5
Ctrl-Alt-Del 42, 46
Cursor 8

DATE 8, 30
Datei 11
Datei nicht gefunden 33
Dateien gleichen Namens 21
Dateien ohne Nachnamen 16
Dateien gleichnamige 28
Dateien Platzierung von 22
Dateierweiterung 23
Dateiname 12
Dateiname nicht gefunden 22
Dateitypen 23
Datum 8
DD 39, 40
Default 45, 46
Dekompressionsprogramm 51
DELETE-TRACKING 37
Desktopmodell 5
deutsche Tastatur 7
deutscher Zeichensatz 48
DIR 16, 17, 30
DIR /P 20
DIR /S 21
DIR /W 20
DIR 21
DIR... 16
Disk Copy Fast 54
DISKCOPY 40

Diskette 39
Diskettenlaufwerk 5, 17
DOC 23, 50
Dokument 31
Doppelpunkt 44
DOS-Befehl 9
DOS-Directory 15
DOSKEY 10, 33, 48
DOS-Shell 9
DOS-Verzeichnis 46
Double Density 39
Double Side 39
DR-DOS 10
DropDownFeld 34
Drucker 7, 25
Druckkopf 7
Durchschlag 7
Düsen 7

Editor 34, 45, 48, 53
Eigenschaften von Dateien 23
ein Punkt 18
Einfg-Taste 10
einfügen Text 35
Eingabe 6
eingeschaltetes Attribut 31
Eject-Knopf 39
Empfangsmethode 27
ENTF 36
entfernen ein Verzeichnis 19
ERASE 43
erstellen ein Unterverzeichnis 19
erstellen ein Verzeichnis 19
erweitern Suchpfad 23
erweiterte Tastatur 6
Esc-Taste 29
EXE 23, 51
Expreßknopf 18
Extension 12
externe Befehle 30
Extract 55

F3 10, 33
F5 46
F7 10
Farbmonitor 6
Faxmodem 5
Festplatte 5, 17, 42
FF.BAT 49
FIND 49
finden eine Datei 49
Fingerabdruck 28, 33
Floppydisk 39
FORMAT 40
FORMAT /S 42, 46

formatieren die Festplatte 41, 57
 formatieren eine Diskette 40
 formatierter Text 34
 Fragezeichen 37
 Freeware 49
 Funktionstaste 6

Gebrauchsanweisung 23, 34, 50
 Gehäuse 4
 gespeicherte Befehlszeile 47

Handbuch 49
 Hardcopy 25
 Hardware 4
 HD 39, 40
 HD-Loch 39
 HELP 43
 hidden 31
 High Density 39
 Hilfstext 43
 hinzufügen zum Suchpfad 23

IBM-kompatibel 4
 infiziertes Programm 57
 Inkjet 7
 INSTALL 49, 50, 52
 interne Befehle 30
 IO.SYS 21, 42, 46

Kaltstart 4
 KB 16
 KEYBOARD.SYS 48
 Kilobytes 16
 Klarmeldung 8
 Kodierung 11
 Kodierungen 25, 34
 Kommandoprozessor 30
 komplizierte Befehle 47
 Kompressionsprogramme 53
 komprimiert 51
 Kopie 26
 Kopie der ursprünglichen
 Startdateien 45
 Kopie einer Datei 26
 Kopie eines gesamten Zweiges 30
 Kopie getreue 40
 kopieren eine ganze Diskette 40
 kopieren Text 35
 kurze Einführung 34
 Kurzübersicht 20

Label 16
 Laserdrucker 7

Laufwerk 17, 43
 Laufwerksbezeichnung 17
 Lautsprecher 25
 LED 7
 löschen ein Verzeichnis 19
 löschen eine Datei 36
 Löschung 36

Make Directory 19
 Matrixdrucker 7
 Maus 7
 MB 52
 MD 19
 MegaByte 16
 Menü 8
 Microchip 8
 Microsoft 4
 Monitor 6
 MORE 48
 Motherboard 5
 MOUSE 48
 MOVE 29
 MSAV 57
 MS-DOS 10
 MSDOS.SYS 21, 42, 46
 Mutterplatine 5
 MWAV 57

Nachname 12
 Nadeldrucker 7
 Name 12
 neues Bild auf dem Schirm 10
 neues Verzeichnis 19
 Norton Commander 9
 Notbremse 29
 NOTIZ.TXT 35
 NumLock 6

optional 43
 Originaldatei 26
 Originaldiskette 40
 Ozon 7

PAK Verzeichnis 53
 Panik 46
 parallel 7
 Parameter 43
 parken die Schreibköpfe 42
 Path 22, 53
 PC 4
 PC-DOS 10
 PC-Tools 9
 PCTRACKR.DEL 37
 Permanente Umschalttaste 6
 Personal Computer 4
 Pfad 14, 22, 43

Pfeiltasten 6
 Phil Katz 54
 Piepser 25
 Pixel 6
 PKUNZIP 54
 PKZIP 54
 PKZIPFIX 54
 Plazierung in Verzeichnis 22
 Postscript 7
 Power On Self Test 8
 PRINT 25, 50, 55
 Print Screen 25
 Printer Echo 25
 PRN 25, 55
 Programme 23
 Prompt 8, 17
 Prozessor 5
 Public Domain 49
 Punkt 12
 Punkt fehlt 20
 Punkt ausschließlich 30

RAM 5
 RD 19
 READ.ME 54
 README 50
 read-only 31
 Reihenfolge im Suchpfad 23
 Remove Directory 19
 REN 28, 30
 RENAME 28
 Reset 4, 42
 retten eine gelöschte Datei 36
 Robert Jung 55
 Route 22
 rückgängig machen Befehl 29
 rückgängig machen eine
 Formatierung 41
 rückgängig machen Löschung 36
 rückwärts-kompatibel 4

scannen ein Laufwerk 57
 Schalter 20, 30, 32
 Schalter Plazierung eines 21
 Schirmbild 24
 Schlußzeichen 17
 schreibgeschützte Datei 31
 Schreibschutz 39
 schubweise 47
 Screen Dump 25
 selbstentpackende Archiv 51
 Semikolon im Suchpfad 23
 SENTRY 37
 seriell 7
 SETUP 49, 50
 Shareware 34, 49
 Shell 9

SICHERHEIT 47
 Sicherheitskopie 31, 32, 46
 Sicherheitsregeln für Startdateien 45
 Slot 5
 Soundcard 5
 Source Disk 40
 Speicher 5
 Speicher im Drucker 7
 Speichern 35
 Speichern unter 35
 spezielle Nachnamen 24
 Spiel 50
 Stammverzeichnis 13, 16, 28
 Start eines Programms 51
 Startdateien 45, 46
 statische Elektrizität 7
 Stern 20
 stoppen Befehl 29
 Strg+Alt+Entf 4
 Strg+C 29
 Strg+PrintScreen 25
 Stromversorgung 4
 Stromzufuhr 5
 suchen eine Datei 21
 suchen in allen Verzeichnisse 21
 Suchpfad 22, 47, 48, 53
 SuperVGA 6
 SVGA 6
 Syntax 43
 SYS 42, 46
 System Area 41
 Systemdatei 31, 41
 Systemdateien 46
 Systemdiskette 41, 45, 46
 sämtliche Unterverzeichnisse 30

 Target Disk 40
 Tastatur 6, 45

 Textdatei 11, 23, 25, 34, 45, 47, 49
 Textzeichen 44
 TIME 9, 30
 Tintenstrahldrucker 7
 TREE 15
 TXT 23
 TYPE 24, 48, 50
 TYPE | MORE 25

 über den Stamm 18
 überschreiben 10, 26, 27
 Übersicht breite 20
 überspringen die Startdateien 46
 Übungsdatei 35

 umbenennen eine Datei 28
 umgekehrter Schrägstrich 17
 Umschalttaste 6
 UNDELETE 36
 UNDELETE.INI 38
 undokumentierter Befehl 21
 UNFORMAT 41
 Unterbrechung 29
 Unterverzeichnis 14
 Unterverzeichnisse eingeschlossen 30
 Urheberrecht 49
 Utilities 22

 Warmstart 42
 wechseln Verzeichnis 17
 weglassen einen Path 28
 Ventilator 5
 VER 10, 30

 verschieben eine Datei 29
 Version 10
 versteckte Datei 31
 versteckte Dateien 21
 verstecktes Verzeichnis 38
 Verzeichnis 13
 Verzeichnis mit Dateien löschen 38
 VGA 6
 wieder installieren 31
 Wildcard 20
 Windows 8
 Virus 57
 Virus gefährlicher 57
 Virus harmloser 57
 WKS 24
 Wochentag 8
 Volume 16
 Word 23
 Vorname 12
 Vorteile von BAT-Dateien 47

 XCOPY 30, 32

 Zahlentaste 44
 Zeichen 11, 44
 Zeichensatz 44
 ZEIGE.BAT 48
 ZIP 54
 ZIP2EXE 54
 zuletzt besuchtes Verzeichnis 28
 zwei Punkte 18
 Zwischenablage 35

 ä ö und ü 7