

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS



Essentielle Kommandos

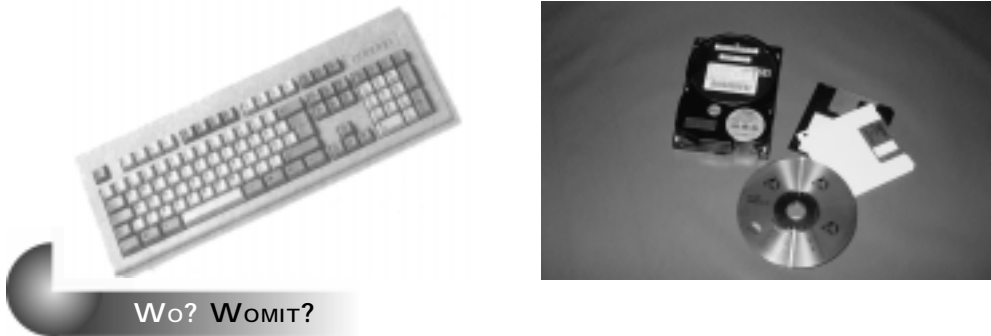
3 KAPITEL

tempo

Allgemeines zum Arbeiten mit Dateien und Verzeichnissen	64
Anzeigen von Verzeichnissen	66
Dateien anzeigen	70
Dateien & Verzeichnisse kopieren	73
Dateien & Verzeichnisse verschieben und umbenennen	76
Dateien & Verzeichnisse löschen	79
Links erstellen	81
Dateien & Verzeichnisse suchen	84
Arbeiten mit Floppys – die »mtools«	88
»MOUNTEN« anderer Dateisysteme	92
Archivieren mit tar, gzip und bzip2	99
Hilfe suchen – man	102
Dateien editieren – der Editor joe	104
Schnell noch mal	110

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Allgemeines zum Arbeiten mit Dateien und Verzeichnissen



Wo? Womit?

Pfade

- ▶ Als Pfadtrenner findet unter Linux der *Slash* »/« Verwendung, anders als unter DOS/Windows, wo der *Backslash* (»\«) benutzt wird.
- ▶ Das oberste Verzeichnis in der Verzeichnishierarchie wird durch einen einzelnen Slash »/« dargestellt.

BEGRIFF

Das oberste Verzeichnis wird als **Wurzel- oder Root-Verzeichnis** bezeichnet.

Eingehängte Datenträger

- ▶ Datenträger wie zum Beispiel Disketten oder CD-ROMs werden nicht wie bei Windows durch einen Laufwerksbuchstaben bezeichnet, sondern in die Verzeichnishierarchie eingehängt.



ACHTUNG

Linux unterscheidet zwischen Groß- und Kleinschreibung. Bei Dateinamen und Pfadnamen muss genau auf die Schreibweise geachtet werden: *datei* und *Datei* sind zwei verschiedene Namen!

Aktuelles Verzeichnis

- ▶ Das aktuelle Verzeichnis, in dem man sich gerade befindet, kann mit dem Befehl `pwd` (engl. *print working directory*) festgestellt werden.

So erfahren Sie, in welchem Verzeichnis Sie sich befinden:

- ▶ Geben Sie `pwd` ein.
- ▶ Sie erhalten als Ausgabe das aktuelle Verzeichnis, z. B. `/home/tb`.

Verzeichnis wechseln

Der Befehl `cd` dient zum Wechseln des aktuellen Verzeichnisses (engl. *change directory*).

- ▶ Um in ein anderes Verzeichnis unterhalb des aktuellen Verzeichnisses zu wechseln, geben Sie `cd Verzeichnispfad` ein.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Um unabhängig vom aktuellen Verzeichnis in ein Verzeichnis zu wechseln, verwenden Sie eine absolute Pfadangabe: `cd /Verzeichnis/pfad`.

Verzeichnis erzeugen

- ▶ Ein neues Verzeichnis wird mit `mkdir Verzeichnisname` erzeugt.
- ▶ Der Verzeichnisname kann auch Pfadangaben enthalten, so dass sich das neue Verzeichnis nicht im aktuellen Verzeichnis befinden muss.

Wechseln ins Benutzerverzeichnis

- ▶ Um ins eigene Benutzerverzeichnis zu wechseln, geben Sie `cd` ohne weitere Argumente ein: `cd`.

Benutzerverzeichnis mit ~

Das Benutzerverzeichnis kann über die Abkürzung `~` erreicht werden.

- ▶ Um von einem beliebigen anderen Verzeichnis in Ihr Benutzerverzeichnis zu wechseln, können Sie auch `cd ~` eingeben.
- ▶ Um in das Verzeichnis *montag* zu wechseln, das ein Unterverzeichnis von *meineDaten* ist, können Sie `cd ~/meineDaten/montag` eingeben, vorausgesetzt, *meineDaten* liegt in Ihrem eigenen Homeverzeichnis.
- ▶ In das Homeverzeichnis eines anderen Benutzers wechseln Sie – so er es Ihnen erlaubt – mit `cd ~benutzername`.

Übergeordnetes und aktuelles Verzeichnis

Wie DOS/Windows kennt auch Linux die Kurzformen `..` für »aktuelles Verzeichnis« und `...` und für »übergeordnetes Verzeichnis«.

- ▶ Um in das Verzeichnis zu gelangen, das eine Ebene höher in der Verzeichnishierarchie liegt, geben Sie ein: `cd ..`.

INFO

Wenn eine Pfadangabe mit einem führenden Slash beginnt, nennt man sie *absolut*, also *unbeeinflusst vom aktuellen Verzeichnis*. Fehlt der führende Slash, so ist die Pfadangabe *relativ zum aktuellen Verzeichnis* zu verstehen.

INFO

Unter Linux hat jeder Benutzer ein eigenes Benutzerverzeichnis, das »Homeverzeichnis«, das für die Ablage seiner persönlichen Daten vorgesehen ist. Normalerweise befindet es sich unter `/home/Benutzername`.

TIP

Unter DOS/Windows kann man auf das Leerzeichen zwischen `cd` und `..` verzichten, unter Linux nicht. Die Shell akzeptiert die Eingabe nur, wenn das Leerzeichen vorhanden ist. Der Grund: Da der Punkt ein grundsätzlich an jeder Stelle im Dateinamen verwendbares Zeichen ist, ruft `cd ..` tatsächlich den Befehl `cd ..` und nicht den Befehl `cd` mit dem Argument auf.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Anzeigen von Verzeichnissen



Wo? WoMIT?

Der Befehl `ls` zeigt den Inhalt von Verzeichnissen, so wie dies der Befehl `dir` bei Verwendung der Windows-Eingabeaufforderung tut.

Anzeigen von Verzeichnissen

- ▶ Um die in einem Verzeichnis enthaltenen Dateien anzuzeigen, geben Sie `ls Verzeichnispfad` ein.
- ▶ In der Ausgabe kann man Verzeichnisse an einem Slash »/« erkennen. Je nachdem, wie das von Ihnen verwendete Terminal konfiguriert ist, werden unterschiedliche Dateitypen und Verzeichnisse zusätzlich noch farblich differenziert.
- ▶ Ausgabe eines Verzeichnisses durch `ls`:

```
Desktop          a                mbox
Mail             autosave        nsmail
ServerLog        axhome          test
SpaceBoom,HighScores b                test2
StartLog         bin             vmware
Trash            loadlin         vmware-distrib
```

Anzeigen von erweiterten Informationen

- ▶ Verwenden Sie die Option `»-l«` hinter dem Befehl `ls`, um die Ausgabe im Langformat zu erzwingen. Dadurch erhalten Sie weitere Informationen zu den Dateien: `ls -l Verzeichnispfad`.

INFO

Linux erkennt ausführbare Dateien am Attribut »eXecute«, nicht an einer bestimmten Endung. Dateierweiterungen aus drei Buchstaben wie Windows kennt Linux nicht. Dateien, deren Namen mit einem Punkt beginnen, sind versteckt.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Die Ausgabe von `ls -l`:

```

Datei (><), Verzeichnis (»«) oder Link (>|«)   Anzahl der harten Links   Besitzer
total 36
drwxr-xr-x 6 root root 1024 Jun 28 12:16 Desktop
drwxr-xr-x 2 root root 1024 Jun 26 13:18 Mail
-rw-r--r-- 1 root root 3804 Jun 29 00:18 ServerLog
-rw-r--r-- 1 root root 190 Jun 27 12:40 Spaceship,Highscores
-rw-r--r-- 1 root root 1064 Jun 29 00:18 StartLog
drwxr-xr-x 2 root root 1024 Jun 29 00:19 Trash
drwxr-xr-x 3 root root 1024 Jul 7 22:50 a
drwxr-xr-x 2 root root 1024 Jun 29 12:07 aufgabe
drwxr-xr-x 2 root root 1024 Jun 28 13:43 schone
drwxr-xr-x 3 root root 1024 Jul 7 22:50 b
drwxr-xr-x 2 root root 1024 Jun 29 02:07 bin
drwxr-xr-x 3 root root 1024 Jun 29 02:00 loadlin
-rw-r--r-- 1 root root 14215 Jul 5 08:24 mbox
drwxr-xr-x 2 root root 1024 Jun 29 00:14 rmail
drwxr-xr-x 3 root root 1024 Jun 30 14:06 test
drwxr-xr-x 3 root root 1024 Jun 29 02:08 test2
drwxr-xr-x 3 root root 1024 Jun 28 12:14 vware
drwxr-xr-x 4 root root 1024 Jun 11 23:52 vware-distrib
    
```

Zugriffsrechte Gruppe Größe in Bytes Datum & Zeit der letzten Veränderung

Anzeigen von versteckten Dateien

- ▶ Manche Dateien sind versteckt und werden nicht ohne weiteres mit dem `ls`-Befehl angezeigt. Dies sind alle Dateien, die mit einem Punkt beginnen.
- ▶ Um versteckte Dateien trotzdem zu sehen, muss die Option `-a` verwendet werden:
- ▶ Der Befehl `ls -l -a ~` zeigt alle Dateien in Ihrem Homeverzeichnis an.
- ▶ Der Befehl `ls -a /` zeigt den Inhalt des Wurzelverzeichnisses an.

Wichtige Optionen des Befehls `ls`

Die folgende Übersicht zeigt eine Auswahl der wichtigsten Optionen für den Befehl `ls`.

Option	Bedeutung
<code>-a</code>	Zeigt auch Dateien an, die mit einem Punkt beginnen.
<code>-d</code>	Zeigt nur den Namen von Verzeichnissen, nicht aber deren Inhalt an.
<code>-l</code>	Zeigt zusätzlich zum Dateinamen weitere Informationen (wie Größe, Zugriffsrechte usw.) über jede Datei an.
<code>-L</code>	Zeigt bei symbolischen Links auf ein Verzeichnis nicht den Pfad des Links, sondern das Verzeichnis an, auf das der Link zeigt.

INFO

Versteckte Dateien sind – je nachdem, wie Sie beim Löschen vorgehen – gegen versehentliches Löschen geschützt und müssen ggf. explizit gelöscht werden.

TIP

Sie können mehrere Optionen bei Befehlen kombinieren, zum Beispiel bei `ls` die Optionen `-a` und `-l`. Dies kann man auch als `ls -la` schreiben.

INFO

Hier sind die Kurzformen der Optionen aufgeführt. Es gibt auch oft eine entsprechende »Langform«, so zum Beispiel `-al` statt `-a`. Entnehmen Sie die entsprechende Langform der man-Page von `ls`.


BEGRIFF

Zu (fast) jedem Linux-Befehl gibt es eine Hilfeseite, auf der der Befehl knapp, aber ausführlich beschrieben wird. Diese Hilfeseiten heißen **man-Pages** (von »Manualpages« – »Handbuchseiten«).

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Option	Bedeutung
-p	Hängt an den Dateinamen ein Sonderzeichen an, so dass der Dateityp unmittelbar ersichtlich ist.
-R	Zeigt auch Dateien in Unterverzeichnissen an.
-t	Die Ausgabe wird nach dem Zeitpunkt der letzten Dateiänderung sortiert.
-u	Die Ausgabe wird nach Datum und Uhrzeit des letzten Lesezugriffs sortiert. Diese Option muss in Verbindung mit -t angewendet werden, da sonst keinerlei Sortierung stattfindet.
-X	Sortiert die Dateien nach der Buchstabenkombination hinter dem letzten Punkt im Dateinamen.

Weitere Informationen über ls

- ▶ Um noch detailliertere Informationen und alle möglichen Optionen kennen zu lernen, rufen Sie die entsprechende *man-Page* ab: `man ls` .

Wildcards (Jokerzeichen)

- ▶ Die Shell bietet gegenüber DOS/Windows eine wesentlich bessere Unterstützung von *Wildcards*.
- ▶ Man benutzt Wildcards, um der Shell mitzuteilen, dass »alle Dateinamen, die diesem Schema entsprechen« gemeint sind.

Dazu bedient man sich der folgenden Zeichen:

- ▶ Der Stern »*« bedeutet »beliebig viele beliebige Zeichen – oder auch gar keines«.
- ▶ Das Fragezeichen »?« bedeutet »irgend ein Zeichen«. An dieser Position muss ein Zeichen vorhanden sein.
- ▶ Durch eckige Klammern kann man Zeichenbereiche zusammenfassen. So könnte »eines der Zeichen a, b oder c« durch `[a, b, c]` oder auch durch `[a-c]` ausgedrückt werden.
- ▶ Ein Ausrufezeichen »!« negiert den Ausdruck: `[!abc]` bedeutet »weder a, noch b, noch c«.

BEGRIFF

Wildcards sind »Jokerzeichen«, die man in Dateinamen angeben kann, um eine Reihe von möglichen Kombinationen darzustellen.

TIP

Bei der Verwendung von *ls* erfasst »*« auch Verzeichnisse, wodurch deren Inhalt angezeigt wird. Will man die tatsächlichen Verzeichnisnamen sehen, muss *ls* mit der Option `-d` aufgerufen werden.

TIP

Der Befehl *ls* bietet die Option `-a`, um versteckte Dateien anzuzeigen. Falls Sie einen Befehl verwenden, der diese Option nicht anbietet, aber trotzdem alle versteckten Dateien erfassen wollen, können Sie folgendes Suchmuster verwenden: `.[!]*`. Dadurch erfassen Sie alle Dateien, deren Name mit einem Punkt beginnt, wobei sich jedoch an der zweiten Position kein Punkt befinden darf, gefolgt von beliebigen anderen Zeichen (oder keinen).

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Beispiele:

- ▶ Is *a? bedeutet »Zeige alle Dateien an, in deren Name der vorletzte Buchstabe ein kleines a ist«.
- ▶ Is *A??? bedeutet »Zeige alle Dateien an, in deren Name der viertletzte Buchstabe ein großes A ist«.
- ▶ Is a* bedeutet »Zeige alle Dateien an, deren Name mit einem kleinen a beginnt«.
- ▶ Is [a,A]* bedeutet »Zeige alle Dateien an, deren Name mit einem kleinen oder großen A beginnt«.

Dateien anzeigen



Wo? Womit?

Natürlich werden Dateien oft mit Spezialprogrammen verarbeitet. Unter Linux kommt man jedoch in sehr viel größerem Maß mit Textdateien in Berührung als unter Windows. Um diese anzuzeigen, finden im Wesentlichen die hier beschriebenen Befehle Verwendung.

Der Befehl `more`

- ▶ Mit dem Befehl `more Datei name` wird der Inhalt einer Datei angezeigt.
- ▶ Durch Drücken der Leertaste wird jeweils die nächste Seite angezeigt.
- ▶ Mit Druck auf `B` (engl. *back*) wird jeweils eine Seite zurückgeblättert.
- ▶ Mit der Taste `h` kann die Hilfe angezeigt werden.
- ▶ `more` beendet sich automatisch, wenn das Dateiende erreicht wurde.
- ▶ Um das Programm bei längeren Dateien vor dem Dateiende zu verlassen, drücken Sie auf `q` (engl. *quit*).

INFO

Programme wie `more` bezeichnet man auch als Pager.

Der Befehl `less`

- ▶ `less` ist – kurz gesagt – ein erweitertes `more`. Es bietet gegenüber `more` eine Reihe von Vorteilen.
- ▶ mit den Tasten `f` `b` kann der Text vertikal (zeilenweise) gerollt werden.
- ▶ In der Datei können Lesezeichen gesetzt werden.
- ▶ Durch Eingabe von `/` Schlüsselwort suchen Sie vorwärts nach dem angegebenen Schlüsselwort, durch `?` Suchbegriff rückwärts.
- ▶ Die Statusmeldung am unteren Bildschirmrand zeigt mehr Informationen und lässt sich über die Option »-P« konfigurieren.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

```

SUMMARY OF LESS COMMANDS

Commands marked with * may be preceded by a number, N.
Notes in parentheses indicate the behavior if N is given.

h H          Display this help.
q ;q Q ;Q ZZ Exit.

e ^E j ^N CR * Forward one line (or N lines).
y ^Y k ^K ^P * Backward one line (or N lines).
f ^F ^V SPACE * Forward one window (or N lines).
b ^B ESC-v    * Backward one window (or N lines).
z            * Forward one window (and set window to N).
w            * Backward one window (and set window to N).
ESC-SPACE   * Forward one window, but don't stop at end-of-file.
d ^D        * Forward one half-window (and set half-window to N).
u ^U        * Backward one half-window (and set half-window to N).
ESC-( RightArrow * Left 8 character positions (or N positions).
ESC-) LeftArrow  * Right 8 character positions (or N positions).
F            * Forward forever; like "tail -f".
r           * Repaint screen.
R           * Repaint screen, discarding buffered input.
HELP -- Press RETURN for more, or q when done
    
```

Der Befehl cat

- ▶ Bei diesem Befehl handelt es sich nicht wie bei *more* und *less* um ein interaktives Anwendungsprogramm.
- ▶ Dieser Befehl erlaubt die Angabe von mehreren Dateinamen und gibt den Inhalt dieser Dateien ohne Unterbrechung aus (daher der Name »cat« von »concatenate« – verketteten).

INFO

cat gibt das Ergebnis auf der Standardausgabe aus. Um die Ausgabe zum Beispiel in eine Datei umzulenken, muss die Ein/Ausgabeumlenkung benutzt werden.

Beispielsweise lassen sich drei *Dateien* *datei_a*, *datei_b* und *datei_c* (die sich im aktuellen Verzeichnis befinden) aneinanderfügen und das Ergebnis in die Datei *ausgabe* schreiben, indem

```
cat datei_a datei_b datei_c > ausgabe
```

eingegeben wird.

Weitere Parameter für *cat*:

Parameter	Wirkung
-b	Nichtleere Ausgabezeilen werden nummeriert.
-n	Alle Ausgabezeilen werden nummeriert.
-v	Kontrollzeichen werden mit einem vorangestellten »^« lesbar dargestellt.

Der Befehl split

- ▶ Dieser Befehl ist das Gegenstück zu *cat* – er zerlegt eine Datei in mehrere Teildateien mit jeweils einer bestimmten Maximalgröße.
- ▶ Die Größe der Teildateien wird in Zeilen oder in Bytes angegeben.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Es muss keine Ausgabedatei angegeben werden. In diesem Fall heißen die einzelnen Ausgabedateien *xaa*, *xab* usw. Falls eine Ausgabedatei angegeben wird, so wird jeweils am Ende *aa*, *ab*, *ac* usw. angehängt.



Anwendung von *split*:

split [optionen] Eingabedatei [Ausgabedatei]

Folgende Optionen sind möglich:


Option	Wirkung
-n	n steht für eine Anzahl Zeilen; jede Ausgabedatei besteht aus maximal dieser Anzahl Zeilen.
-l n	Wie -n.
-b n	Es werden n Bytes als jeweilige Maximalgröße angegeben. Folgt hinter der Zahl n ein »k«, so sind Kilobyte, bei einem »m« Megabyte gemeint.
-C	Arbeitet wie -b, wobei die Dateien jedoch an Zeilengrenzen getrennt werden.

Beispiele:

- ▶ `split -l 10 dateiein dateiaus.`  - erzeugt Dateien mit den Namen *dateiaus.aa*, *dateiaus.ab* usw. mit jeweils 10 Zeilen Inhalt.
- ▶ `split -5k dateiein datei`  - erzeugt Dateien mit den Namen *dateiaa*, *dateiab* usw. mit jeweils 5 Kbyte Größe.

Die Befehle *head* und *tail*

Ähnlich wie *cat* geben die Kommandos *head* und *tail* die Eingabedatei wieder. Sie dienen jedoch nicht zum Verketteten von Dateien, sondern

- ▶ *head* zeigt die ersten Zeilen der Datei,
- ▶ *tail* zeigt die letzten Zeilen der Datei an,
- ▶ standardmäßig werden jeweils zehn Zeilen angezeigt.
- ▶ den Standardwert können Sie ändern, indem Sie eine andere Zeilenzahl als Argument angeben. So zeigt zum Beispiel `head -3 /etc/fstab`  die ersten drei Zeilen der Datei */etc/fstab* an.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Dateien & Verzeichnisse kopieren



Dateien kopieren

- ▶ Dateien werden mit dem Befehl `cp` kopiert.
- ▶ Der Aufruf lautet dabei `cp Quelldatei Ziel`, wobei das Ziel entweder ein anderer Dateiname oder ein Verzeichnis sein kann.

INFO

Das Verhalten von `cp` wird durch zusätzliche Schalter festgelegt.

Kopieren einer Datei:

- ▶ Mit `cp datei1 datei2` wird eine Datei kopiert.
- ▶ Durch diesen Befehl dupliziert man die Datei mit dem Namen `datei1` im aktuellen Verzeichnis in eine Datei mit dem Namen `datei2` im aktuellen Verzeichnis.
- ▶ Falls `datei2` schon existiert, wird die originale `datei2` durch die Kopie von `datei1` ersetzt.
- ▶ Verwenden Sie Pfadangaben, wenn Sie die Datei zwischen verschiedenen Verzeichnissen kopieren wollen.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal kopieren wollen.

→ 64

→ 68

INFO

Natürlich können die *Quelldatei* und *Zieldatei* an völlig verschiedenen Orten liegen: Bei `cp /datei1 ~/datei2` liegt die *Quelldatei* im Wurzelverzeichnis und die *Zieldatei* im Homeverzeichnis.

Einige Beispiele:

- ▶ `cp /datei .` – kopiert die Datei mit Namen `datei` aus dem Wurzelverzeichnis in das aktuelle Verzeichnis.
- ▶ `cp ../a* .` – kopiert alle Dateien aus dem übergeordneten Verzeichnis, deren Name mit einem kleinen »a« beginnen, ins aktuelle Verzeichnis.
- ▶ `cp *.~` – kopiert alle Dateien im aktuellen Verzeichnis, deren Name auf einem Punkt enden, ins Homeverzeichnis.

Kopieren einer Datei mit Nachfrage

- ▶ Mit `cp -i Quelldatei Zieldatei` erzeugen Sie ein Duplikat der *Quelldatei* unter dem Namen *Zieldatei*, wobei

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

gefragt wird, ob die *Zieldatei* überschrieben werden darf, falls sie schon vorhanden ist.

- ▶ Mit `cp -i Quelldatei Zielverzeichnis` erzeugen Sie eine Kopie der *Quelldatei* im Verzeichnis *Zielverzeichnis*, wobei gefragt wird, ob die *Zieldatei* überschrieben werden darf, falls sie schon vorhanden ist.
- ▶ Wenn Sie möchten, dass bereits existierende Dateien auf gar keinen Fall überschrieben werden, verwenden Sie statt der Option »-i« die Option »-u«.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal kopieren möchten.

→ 68

Kopieren einer Datei unter Beibehaltung der Dateiattribute

- ▶ Mit `cp -a Quelldatei Zieldatei` erzeugen Sie ein Duplikat der *Quelldatei* unter dem Namen *Zieldatei*, wobei möglichst alle Dateiattribute beibehalten werden.
- ▶ Mit `cp -a Quelldatei Zielverzeichnis` erzeugen Sie eine Kopie der *Quelldatei* im *Zielverzeichnis*, wobei möglichst alle Dateiattribute übernommen werden.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal kopieren möchten.

Kopieren einer Datei unter Beibehaltung der Besitzerinformationen

- ▶ Mit `cp -p Quelldatei Zieldatei` erzeugen Sie ein Duplikat der *Quelldatei* unter dem Namen *Zieldatei*. Dabei werden die Informationen über Besitzer, Gruppenzugehörigkeit, Zugriffsrechte sowie der Zeitpunkt der letzten Änderung nicht verändert.
- ▶ Mit `cp -p Quelldatei Zielverzeichnis` erzeugen Sie eine Kopie der *Quelldatei* im *Zielverzeichnis* ohne Veränderung der Besitzerinformationen.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal kopieren möchten.

Verzeichnisse kopieren

Auch zum Kopieren von Verzeichnissen wird der Befehl `cp` verwendet.

- ▶ Mit `cp verzeichnis1/* verzeichnis2` wird der komplette Inhalt von *verzeichnis1* ins *verzeichnis2* kopiert.
- ▶ Das zweite Verzeichnis muss bereits existieren (d.h. es muss mit `mkdir` bereits angelegt worden sein).

→ 89

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Der Inhalt von Unterverzeichnissen wird dadurch nicht mit kopiert.

Verzeichnisse mit Unterverzeichnissen kopieren

- ▶ Mit `cp -R verzeichnis1 verzeichnis2` wird der komplette Inhalt von `verzeichnis1` in `verzeichnis2` kopiert.
- ▶ Die Option »-R« bewirkt, dass auch Unterverzeichnisse und deren Inhalt kopiert werden.

Wichtige Optionen des Befehls cp

Die folgende Übersicht zeigt eine Auswahl der wichtigsten Optionen für den Befehl »cp«.

Option	Bedeutung
-a	Behält, wenn möglich, alle Dateiattribute bei.
-b	Bereits vorhandene gleichnamige Dateien werden nicht überschrieben, sondern in <i>Dateiname</i> plus »~« umbenannt.
-d	Kopiert Links, nicht das Ziel des Links.
-i	Fragt nach, ob Dateien überschrieben werden dürfen.
-l	Erstellt harte Links, statt Dateien zu kopieren.
-p	Lässt die Informationen über Besitzer, Zugriffsrechte usw. unverändert.
-R	Unterverzeichnisse samt Inhalt werden ebenfalls kopiert.
-s	Erstellt symbolische Links, statt die Dateien zu kopieren.
-u	Kopiert Dateien nur dann, wenn nicht bereits eine Zieldatei mit dem entsprechenden Namen existiert.

INFO

Hier sind die Kurzformen der Optionen aufgeführt. Es gibt auch oft eine entsprechende »Langform«, so zum Beispiel `-all` statt `-a`. Entnehmen Sie die entsprechende Langform der *man-Page* von `cp`.

Weitere Informationen über cp




- ▶ Um noch detailliertere Informationen und alle möglichen Optionen kennen zu lernen, rufen Sie die entsprechende *man-Page* ab: `man cp`.

Dateien & Verzeichnisse verschieben und umbenennen



Um Dateien und Verzeichnisse zu verschieben oder ihnen einen anderen Namen zu geben, wird der Befehl *mv* benutzt.

Dateien verschieben

- ▶ Mit *mv Datei Zielverzeichnis*  wird eine Datei verschoben.
- ▶ Die ursprüngliche Datei *Datei* ist anschließend nicht mehr vorhanden, existiert aber unter demselben Namen als Kopie im *Zielverzeichnis*.
- ▶ Falls im *Zielverzeichnis* schon eine Datei mit dem entsprechenden Namen existiert, wird diese überschrieben.
- ▶ Verwenden Sie Pfadangaben im Dateinamen, wenn die Datei nicht im aktuellen Verzeichnis liegt.  64
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal verschieben wollen.  68

INFO

*Ein Verschieben im selben Verzeichnis (d. h. es wird ein Duplikat erstellt und das Original gelöscht) ist lediglich ein Umbenennen. Daher wird *mv* sowohl zum Verschieben, als auch zum Umbenennen benutzt.*

Dateien verschieben mit Nachfrage/ Backup

Verwenden Sie den Befehl *mv* wie oben beschrieben, aber kombinieren Sie ihn zusätzlich

- ▶ mit der Option »-i«, wenn Sie möchten, dass vor dem Überschreiben von evtl. existierenden Dateien nachgefragt wird,
- ▶ mit der Option »-b«, wenn Sie möchten, dass vor dem Überschreiben von evtl. existierenden Dateien eine Sicherheitskopie angelegt wird.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Dateien umbenennen

- ▶ Mit `mv AlterName NeuerName` wird eine Datei umbenannt.
- ▶ Falls schon eine Datei unter dem Namen *NeuerName* existiert, wird diese überschrieben.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal umbenennen wollen.
- ▶ Wenn Sie vor dem Überschreiben von evtl. existierenden Dateien gefragt werden wollen, verwenden Sie die Option »-i«.
- ▶ Wenn Sie möchten, dass vor dem Überschreiben von evtl. existierenden Dateien eine Sicherheitskopie angelegt wird, bietet sich die Option »-b« an.

➔ 68

Verzeichnisse verschieben

- ▶ Mit `mv Verzeichnis NeuesVerzeichnis` wird *Verzeichnis* in *NeuesVerzeichnis* verschoben, sofern *NeuesVerzeichnis* schon existiert.
- ▶ Beispielsweise verschiebt `mv test ~/` das Verzeichnis »test« ins Homeverzeichnis verschoben.

Verzeichnisse umbenennen

- ▶ Mit `mv AlterName NeuerName` wird ein Verzeichnis im selben übergeordneten Verzeichnis »verschoben«, also umbenannt, sofern *NeuerName* noch nicht existiert.
- ▶ Wird beim neuen Namen ein anderes Zielverzeichnis angegeben, so kann das Verzeichnis auch unter einem neuen Namen in ein anderes Verzeichnis verschoben werden (`mv test ~/test2` – verschiebt z.B. das Verzeichnis »test« ins Homeverzeichnis und gibt ihm den neuen Namen »test2«).

Wichtige Optionen des Befehls mv

Die folgende Übersicht zeigt eine Auswahl der wichtigsten Optionen für den Befehl *mv*.

Option	Bedeutung
-b	Falls bereits eine gleichnamige Zieldatei vorhanden ist, wird sie in <i>Dateiname</i> plus »-« umbenannt.
-i	Falls die Zieldatei existiert, wird nachgefragt, bevor sie überschrieben wird.


INFO
 Hier sind die Optionen in Kurzform aufgeführt. Es gibt oft auch eine entsprechende »Langform«. Entnehmen Sie die entsprechende Langform der man-Page für diesen Befehl.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Option Bedeutung

-f	Existierende Dateien werden ohne Rückfrage überschrieben.
-u	Überschreibt bereits existierende Zielf Dateien nur dann, wenn Sie kein neueres Veränderungsdatum aufweisen.
-v	Anzeigen der Datei, die gerade verschoben wird.

Weitere Informationen über *mv*

- Um noch detailliertere Informationen und alle möglichen Optionen kennen zu lernen, rufen Sie die entsprechende *man-Page* ab: `man mv` .

Dateien & Verzeichnisse löschen



Um Dateien zu löschen, wird der Befehl `rm` benutzt. Zum Löschen von Verzeichnissen ist der Befehl `rmdir` vorgesehen.

Dateien löschen

- ▶ Mit `rm Datei` wird die angegebene Datei gelöscht.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Dateien auf einmal löschen wollen.
- ▶ Verwenden Sie Pfadangaben im Dateinamen, wenn die Datei nicht im aktuellen Verzeichnis liegt.

→ 68
→ 64

Verzeichnisse löschen

- ▶ Mit `rmdir Verzeichnis` wird das angegebene Verzeichnis gelöscht.
- ▶ Verwenden Sie *Jokerzeichen*, wenn Sie mehrere Verzeichnisse auf einmal löschen wollen.
- ▶ Verwenden Sie Pfadangaben im Dateinamen, wenn die Datei nicht im aktuellen Verzeichnis liegt.
- ▶ Das Verzeichnis kann nur mit `rmdir` gelöscht werden, wenn es leer ist.

Nicht-leere Verzeichnisse löschen

- ▶ Mit `rm -r Verzeichnis` wird das angegebene Verzeichnis gelöscht, auch wenn es nicht leer ist.
- ▶ Auch alle darin enthaltenen Unterverzeichnisse samt Inhalt werden gelöscht.

INFO

Bei Verwendung der Option »-r« werden auch mit `rm` Verzeichnisse gelöscht.

ACHTUNG

Wenn der Dateiname Sonderzeichen enthält (zum Beispiel »#«, das die Sonderbedeutung eines Kommentarzeichens hat), müssen die Dateinamen in Apostrophe eingeschlossen werden, um eine Fehlinterpretation zu verhindern.


ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Wichtige Optionen des Befehls `rm`

Die folgende Übersicht zeigt eine Auswahl der wichtigsten Optionen für den Befehl `rm`.

Option	Bedeutung
<code>-i</code>	Bevor eine Datei gelöscht wird, muss dies zunächst bestätigt werden.
<code>-f</code>	Löscht ohne jede Rückfrage auch Verzeichnisse.
<code>-r</code>	Löscht auch Dateien in allen Unterverzeichnissen. Wenn das Unterverzeichnis dadurch leer wird, wird auch der Verzeichniseintrag entfernt.
<code>-v</code>	Ausgabe der gerade bearbeiteten Datei.

Weitere Informationen über `rm`

- Um noch detailliertere Informationen und alle möglichen Optionen kennen zu lernen, rufen Sie die entsprechende *man-Page* ab: `man rm` .

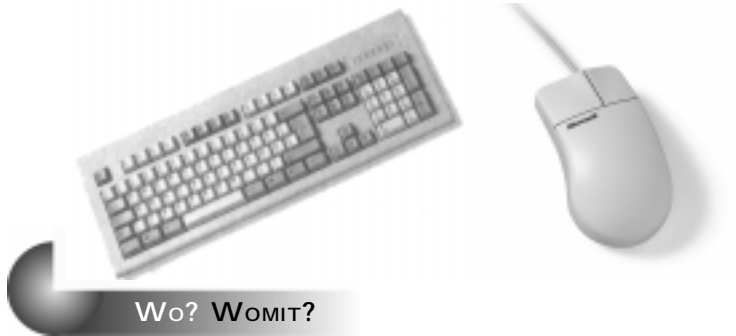
INFO

Hier sind die Kurzformen der Optionen aufgeführt. Es gibt auch oft eine entsprechende »Langform«. Entnehmen Sie die entsprechende Langform der *man-page* für diesen Befehl.

ACHTUNG

Vorsicht: Ein extrem gefährlicher Befehl ist »`rm -rf *`« – denn dadurch werden alle Dateien und Verzeichnisse im aktuellen Verzeichnis gelöscht (mit Ausnahme der hier befindlichen versteckten d.h. mit `.` beginnenden Dateien). Wenn der Benutzer, der diesen Befehl ausführt, als `root` angemeldet ist und sich im Wurzelverzeichnis befindet, ist die Platte anschließend leer.

Links erstellen



- ▶ Links sind Verweise auf Dateien, so dass es möglich wird, von verschiedenen Orten der Verzeichnisstruktur auf ein- und dieselbe Datei zuzugreifen.
- ▶ Um Links einzurichten, wird das Kommando *ln* verwendet.
- ▶ Es gibt zwei verschiedene Arten von Links: die »harten Links« und die »symbolischen Links«.

Harte Links

- ▶ Harte Links werden intern verwaltet, indem die *Inode-Nummer* des Ziels gespeichert wird.
- ▶ Harte Links können nicht zwischen verschiedenen physikalischen Dateisystemen (z.B. zwei Festplatten) verwendet werden.
- ▶ Harte Links können zwar auf Verzeichnisse verweisen, aber das Einrichten solcher harter Links ist nur dem Superuser möglich und verlangt eine spezielle Vorgehensweise.

➔ 112

Symbolische Links

- ▶ Symbolische Links speichern intern keine *Inode-Nummer*, sondern den Dateinamen bzw. die Pfadangabe.
- ▶ Dadurch können symbolische Links auch über Dateisystem-Grenzen hinweg verwendet werden.
- ▶ Auch normale Benutzer können mit symbolischen Links Verweise auf Verzeichnisse einrichten.

Link auf eine Datei einrichten

- ▶ Mit `ln Datei1 Linkname` wird ein Verweis *Linkname* erzeugt, so dass nun scheinbar zwei Dateien existieren. Tatsächlich verweist *Linkname* nur auf *Datei1*.

INFO

Mit Links kann auch problemlos eine Datei unter mehreren Namen angesprochen werden.

BEGRIFF

Symbolische Links werden oft auch als **weiche Links** (soft links) bezeichnet.

INFO

Mit dem Befehl `ls -li` werden auch die Inode-Nummern ausgegeben. Bei »hart verlinkten« Dateien ist diese identisch.

INFO

Inodes, »Informationsknoten«, enthalten sämtliche Verwaltungsinformationen zu einer Datei (z.B. wo sich die Datei auf der Festplatte befindet, wie groß sie ist, wem sie gehört etc.).

INFO

Bei `ls -l` wird bei symbolischen Links angezeigt, wohin sie verweisen. Bei harten Links ist dies sehr viel schwieriger festzustellen.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

- ▶ Ein so erzeugter Link ist ein harter Link.
- ▶ Um einen symbolischen Link einzurichten, muss die Option »-s« verwendet werden.

Link auf einem Verzeichnis einrichten

- ▶ Mit `ln -s Verzeichnis Linkname` wird ein Verweis *Linkname* auf das angegebene Verzeichnis erzeugt.
- ▶ Ein so erzeugter Link ist ein symbolischer Link. Als Superuser könnten Sie auch einen harten Link auf ein Verzeichnis einrichten; dazu muss die Option »-d« benutzt werden.

Verändern von »verlinkten« Dateien

- ▶ Wenn eine Datei verändert wird, die in Wirklichkeit ein Link ist, so wird in jedem Fall die Originaldatei geändert. Die Änderung wirkt sich auf alle Links aus, die auf die entsprechende Datei zeigen.

Löschen von harten Links

- ▶ Löschen Sie den betreffenden Link wie eine ganz normale Datei mit *rm*.
- ▶ Der durch den Link dargestellte Dateiname wird dadurch entfernt.
- ▶ Solange noch mindestens ein harter Link auf der Datei existiert, bleibt der Inhalt der Datei weiterhin verfügbar, selbst wenn es sich beim gelöschten Link um den Namen der Originaldatei gehandelt hat.

Löschen von symbolischen Links

- ▶ Löschen Sie den betreffenden Link wie eine ganz normale Datei mit *rm*.
- ▶ Der durch den Link dargestellte Dateiname wird dadurch entfernt.
- ▶ Das Löschen eines symbolischen Links hat keinen Einfluss auf die Originaldatei.
- ▶ Wird die Originaldatei gelöscht, so existieren alle symbolischen Links weiterhin, verweisen nun aber auf eine nicht mehr existierende Datei. Ein Zugriffsversuch über einen solchen Link erzeugt eine Fehlermeldung.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Wichtige Optionen des Befehls `ln`


Die folgende Übersicht zeigt eine Auswahl der wichtigsten Optionen für den Befehl `ln`.

Option	Bedeutung
-b	Erstellt eine Kopie, falls eine Datei unter dem Namen des zu erstellenden Links schon vorhanden ist (durch Anhängen von »-« an den Dateinamen).
-d	Erzeugt einen harten Link auf ein Verzeichnis, wobei diese Option nur mit Superuser- Rechten benutzt werden kann.
-s	Erstellt symbolische Links (standardmäßig werden harte Links erzeugt).
-v	Zeigt die gerade bearbeitete Datei an.

INFO

Hier sind die Kurzformen der Optionen aufgeführt. Es gibt auch oft eine entsprechende »Langform«. Entnehmen Sie diese der *man-Page* für diesen Befehl.

Weitere Informationen über `ln`

- Um noch detailliertere Informationen zu `ln` zu erhalten, rufen Sie die entsprechende *man-Page* ab: `man rm` .

Dateien & Verzeichnisse suchen



Wo? Womit?

Um Dateien zu suchen, werden im Wesentlichen vier Befehle verwendet:

- ▶ *find* – durchsucht (falls nichts anderes angegeben ist: ab dem aktuellen Verzeichnis) die Unterverzeichnisse nach bestimmten Suchkriterien.
- ▶ *locate* – durchsucht nicht das Dateisystem, sondern eine Datenbank mit Dateinamen und ist damit sehr viel schneller als *find*. Jedoch muss die Datenbank mit zusätzlichen Befehlen gepflegt werden.
- ▶ *whereis* – durchsucht bestimmte voreingestellte Pfade für ausführbare Programme, Dokumentationen und Quellcodes.
- ▶ *which* – durchsucht alle im Suchpfad angegebenen Pfade nach einem Befehl und liefert den vollständigen Namen des Befehls.

Dateien & Verzeichnisse mit »find« suchen

Um eine Liste aller Dateien in einem Verzeichnis oder einem darunter liegenden Verzeichniss zu erhalten, haben Sie folgende Möglichkeiten:

- ▶ `find .` listet alle Dateien des aktuellen Verzeichnisses sowie aus allen Unterverzeichnissen des aktuellen Verzeichnisses auf.
- ▶ `find Startverzeichnis` listet alle Dateien im angegebenen Startverzeichnis und seinen Unterverzeichnissen auf.



Eine oder mehrere Dateien, die einem bestimmten *Namensschema* (Dateiname mit Jokerzeichen) entsprechen, finden sie folgendermaßen:

➔ 68


INFO

Der Suchpfad wird wie unter DOS in der Variablen *PATH* abgelegt. Sie können ihn mit dem Befehl `echo $PATH` einsehen. Lediglich Kommandos, die im *PATH* zu finden sind, lassen sich mit ihrem Namen aufrufen, ansonsten ist die komplette Pfadangabe nötig.


• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ `find -name 'Namensschema'`  sucht nach allen entsprechenden Dateien, ab dem aktuellen Verzeichnis und in den darunter liegenden Unterverzeichnissen.
- ▶ `find / -name 'Namensschema'`  sucht wie zuvor, jedoch beginnt die Suche im Wurzelverzeichnis und findet daher jede entsprechende Datei.


Um Jokerzeichen nicht nur im Dateinamen, sondern auch in der Pfadangabe verwenden zu können, muss die Option `»-path«` statt `»-name«` verwendet werden:

- ▶ `find -path 'Namensschema'`  sucht nach allen Verzeichnissen und Dateien, die dem *Namensschema* entsprechen. Dabei werden Jokerzeichen auch in Pfadangaben berücksichtigt.


Dateien finden, die vor n Tagen geändert wurden:

- ▶ `find -ctime n 'Namensschema'`  sucht im aktuellen Verzeichnis nach allen Dateien, die vor genau *n* Tagen geändert wurden.
- ▶ Wenn Sie nach allen Dateien suchen möchten, die älter als die angegebene Anzahl Tage sind, dann stellen Sie der Zahl ein + voran.
- ▶ Um Dateien zu suchen, die jünger als die angegebene Anzahl von Tagen sind, stellen Sie der Zahl ein - voran.

Dateien finden, die größer/kleiner als eine vorgegebene Größe sind:

- ▶ `find -size nc 'Namensschema'`  sucht im aktuellen Verzeichnis nach allen Dateien, die genau die Größe von *n* Bytes haben.
- ▶ Wenn Sie statt Bytes lieber Kbytes angeben möchten, hängen Sie statt des `»c«` ein `»k«` an die Zahl an.
- ▶ Wenn Sie nach allen Dateien suchen möchten, die größer als die angegebene Größe sind, dann stellen Sie der Zahl ein + voran.
- ▶ Um Dateien zu suchen, die kleiner als die angegebene Größe sind, leiten Sie die Zahl mit einem - ein.

Weitere Optionen von »find«

- ▶ *find* bietet noch weitere Suchmöglichkeiten. Zum Beispiel können alle Dateien gesucht werden, die einer bestimmten Gruppe (nicht) gehören, eine bestimmte Größe haben usw.
- ▶ Die folgende Tabelle bietet einen auszugsweisen Überblick über die möglichen Optionen von *find*. Vollständige Informationen dazu erhalten Sie mit `man find` .

ACHTUNG

Vergessen Sie die Hochkommata (um das Namensschema herum) nicht.

TIP

Mit dieser Option können Jokerzeichen benutzt werden, um sowohl Pfade als auch Dateinamen zu filtern: `find / -path '*zzz/a*' findet alle Dateien die mit »a« beginnen, in allen Verzeichnissen, die mit »zzz« enden.`

TIP

Setzen Sie für *n* die Anzahl der gewünschten Tage ein.


ACHTUNG

nc heißt hier »Zahl mit angehängtem Buchstaben c«, also bedeutet beispielsweise `»100c«` »100 Bytes«.

TIP

Die einzelnen Suchkriterien und Pfadangaben lassen sich miteinander kombinieren. So können Sie beispielsweise nach allen Dateien suchen, die einem bestimmten Namensschema entsprechen, älter als *n* Tage und einer bestimmten Gruppe zugehörig sind.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Option	Argument	Bedeutung
-ctime	Anzahl Tage	Zeitpunkt der Dateiveränderung. Kann durch ein vorangestelltes »+« oder »-« zu »älter« bzw. »jünger als« modifiziert werden.
-group	Gruppenname	Zugehörigkeit zur angegebenen Benutzergruppe.
-name	Namensschema	Dateiname muss dem angegebenen Schema (inkl. Jokerzeichen) entsprechen.
-path	Namensschema	Wie -name, jedoch wird auch der Pfadname berücksichtigt.
-perm	Oktalwert	Ein Oktalwert, der die Zugriffsrechte beschreibt, die die Datei aufweisen muss. 
-size	Größenangabe	Größe der Datei im Vielfachen von 512 Bytes. Durch Anhängen von c kann die Größe in Bytes, durch Anhängen von k in Kbytes angegeben werden. Ein vor der Zahl stehendes »+« bzw. »-« bedeutet »größer« bzw. »kleiner«.
-type	Dateityp	Die Dateien müssen dem Dateityp entsprechen. Dabei handelt es sich um ein Zeichen, u.a. »f« für normale Dateien, »d« für Verzeichnisse und »l« für Links.
-user	Benutzername	Die Datei muss dem angegebenen Benutzer gehören.

Dateien & Verzeichnisse mit »locate« suchen

- ▶ Der Befehl sucht nicht im Dateisystem, sondern in einer eigenen Datenbank nach Dateinamen.
- ▶ *locate* hat den Vorteil, dass es wesentlich schneller arbeitet als *find*.
- ▶ Der Nachteil von *locate* ist, dass die Datenbank erst aktualisiert werden muss, bevor eine Datei darin gefunden werden kann.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Um die Datenbank zu aktualisieren, muss root den Befehl `updatedb` ausführen. Dies ist nach jeder Veränderung der Verzeichnisstruktur notwendig.
- ▶ Um eine Datei zu finden, deren Namen die angegebene Suchzeichenfolge enthält, geben Sie `locate` *Suchzeichenfolge* ein.
- ▶ Um Jokerzeichen im Dateinamen zu verwenden, stellen Sie den Suchbegriff in Hochkommata.

TIP
 Meistens wird diese Aktualisierung automatisch von einem »Cron-Job« abgewickelt.

→ 13

→ 68

Dateien mit `whereis` suchen

Um die Pfade zu durchsuchen, in denen normalerweise Programme, Dokumentationen oder Quelltexte gespeichert werden, geben Sie ein

- ▶ `whereis` *Dateiname*
- ▶ Sie erhalten als Ergebnis eine Liste aller Verzeichnisse, die die entsprechende Datei enthalten.

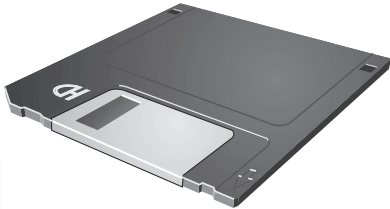
Hinter `whereis` können zusätzliche Parameter angegeben werden:

Parameter	Wirkung
-b	Es wird nur nach Binärdateien gesucht.
-m	Es wird nur nach Dokumentationseinträgen gesucht.
-s	Es wird nur nach Quellcode-Dateien gesucht.
-u	Es wird nur nach Dateien gesucht, auf die die (anderen) übergebenen Parameter nicht zutreffen.

Dateien mit `which` suchen

- ▶ `which` *Dateiname* durchsucht alle in *PATH* angegebenen Verzeichnisse (»Pfade«) nach *Dateiname*. Es liefert dann den vollständigen Namen (inkl. Pfad) des Kommandos.
- ▶ Dies ist hilfreich, wenn von einem Befehl mehrere Versionen existieren, oder wenn festgestellt werden soll, in welchem Verzeichnis das Programm liegt.

Arbeiten mit Floppys – die »mtools«



Wo? WoMIT?

Der Zugriff auf Disketten gestaltet sich unter Linux anders als beispielsweise unter Windows:

- ▶ Eine Diskette muss zuerst mit *mount* in das Dateisystem eingebunden werden. ➔ 92
- ▶ Dann kann auf ihren Inhalt zugegriffen werden.
- ▶ Schliesslich muss sie vor dem Entnehmen wieder »unmountet«, also aus dem Dateisystem ausgehängt werden.
- ▶ Soll eine Diskette beispielsweise formatiert werden, sind noch weitere Schritte erforderlich. ➔ 47

Eine bequemere Alternative als die normalen Linux-Befehle zur Dateiverwaltung bieten die sogenannten *mtools*. Dabei handelt es sich um ein einziges Programm, das aber unter verschiedenen Namen angesprochen wird.

Die *mtools* bieten darüber hinaus den Vorteil, dass man sich nicht mit den Linux-Gerätebezeichnungen für Disketten auseinandersetzen muss, sondern ganz wie gewohnt die Laufwerksbezeichner *A:* und *B:* verwenden kann.

Hinweise zur Verwendung der »mtools«

- ▶ Wenn Sie keinen Laufwerksbezeichner (»A:« oder »B:«) im Dateinamen angeben, beziehen sich alle Angaben auf Laufwerk A:
- ▶ Falls mit *mcd* ein Arbeitsverzeichnis eingestellt wurde, und kein Laufwerk/Verzeichnis im Dateinamen angegeben wurde, werden die Dateien im festgelegten Arbeitsverzeichnis gesucht.
- ▶ Das Jokerzeichen »*« funktioniert wie unter Linux gewohnt, d.h., es muss nur »*« und nicht etwa »*. *« wie unter Windows verwendet werden. ➔ 68

TIP

Auch kann man bei den *mtools* als Trennzeichen für Pfade sowohl den normalen »/« als auch den von Windows gewohnten »\« verwenden. Dann muss der Dateiname aber in Hochkommata eingeschlossen werden.


• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Sie können Disketten problemlos wechseln, ohne sie vorher in das Dateisystem ein- oder aushängen zu müssen.
- ▶ Die Laufwerksbezeichnungen »A:« und »B:« stehen für das erste und zweite Diskettenlaufwerk, »J:« für ein *Jaz-Drive* (falls vorhanden) und »Z:« für ein *Zip-Drive* (falls vorhanden).



TIP

Die Laufwerksbuchstaben können auch in Kleinbuchstaben angegeben werden.


Arbeitsverzeichnis wechseln

- ▶ Das aktuelle Arbeitsverzeichnis wird durch Aufruf des Befehls *mcd* ohne Argumente angezeigt: *mcd* .
- ▶ Wird hinter dem Aufruf von *mcd* ein (Laufwerk und) Verzeichnisname angegeben, so wird in dieses Verzeichnis gewechselt.
- ▶ *mcd* kann benutzt werden, um auf ein anderes (Disketten-)Laufwerk zu wechseln.


Verzeichnisse anzeigen

- ▶ Um den Inhalt eines Verzeichnisses auf der Diskette anzuzeigen, verwenden Sie den Befehl *mdir*: *mdi r Verzeichnisname* .
- ▶ Geben Sie vor dem Verzeichnisnamen den Laufwerksbuchstaben an, wenn es sich nicht um Laufwerk »A:« handelt.
- ▶ Geben Sie nur den Laufwerksbezeichner an, um den Inhalt des Wurzelverzeichnisses der Diskette zu sehen, zum Beispiel *mdi r a:* .
- ▶ Mit zusätzlichen Parametern kann das Verhalten von *mdir* angepasst werden, so werden zum Beispiel mit »*mdi r -a*« auch versteckte Dateien angezeigt.

Verzeichnisse erstellen

- ▶ Um ein neues Verzeichnis auf Diskette zu erstellen, geben Sie ein: *mmd Verzeichnisname* .
- ▶ Geben Sie vor dem Verzeichnisnamen den Laufwerksbuchstaben an, wenn es sich nicht um Laufwerk »A:« handelt.


Dateien oder Verzeichnisse verschieben

- ▶ Mit *mmove* *Quelldatei Zieldatei*  wird eine Datei auf der Diskette verschoben.
- ▶ Statt einer Quelldatei/ Zieldatei kann auch ein Verzeichnis angegeben werden.


ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

- ▶ *mmove* kann, im Gegensatz zu Windows, Unterverzeichnisse verschieben.
- ▶ Wenn kein Laufwerksbuchstabe angegeben wird, wird angenommen, dass es sich um Laufwerk »A:« handelt.
- ▶ Wird nur ein Laufwerksbuchstabe angegeben, so wird angenommen, dass sowohl Quelle als auch Ziel auf demselben Datenträger sind.
- ▶ Es ist nicht möglich, Dateien zwischen der Diskette und dem Unix-Dateisystem zu bewegen.

Verzeichnisse löschen

- ▶ Der Befehl *mrd Verzeichnis*  löscht das angegebene Verzeichnis auf der Diskette, falls dieses leer ist.
- ▶ Nichtleere Verzeichnisse können mit allen Unterverzeichnissen mit dem Befehl *mdeltree* entfernt werden.

Kopieren von Dateien auf und von Diskette

- ▶ Mit dem *mcopy*-Befehl können Dateien zwischen der Diskette und dem Unix-Dateisystem kopiert werden.
- ▶ Mit *mcopy Quelldatei Zieldatei*  wird eine Datei kopiert, wobei sowohl Quelle als auch Ziel jeweils auf der Diskette oder auf einem anderen eingehängten Dateisystem liegen können.
- ▶ Bei der Datei, die keinen Laufwerksbezeichner beinhaltet, wird angenommen, dass es sich um eine Datei aus dem Unix-Dateibaum handelt.
- ▶ Wenn ein Laufwerksbezeichner ohne weitere Dateiangaben als Quelle angegeben wird (z.B. *mcopy a: .*) werden alle Dateien dieses Datenträgers kopiert.
- ▶ Wenn kein Ziel angegeben wird, wird das aktuelle Verzeichnis im Unix-Dateibaum als Ziel angenommen (z. B. *mcopy a: test.txt*)

Es gibt zwei vereinfachte Varianten des *mcopy*-Befehls, die nur das Lesen/Schreiben in eine Richtung erlauben:


- ▶ *mread DosDatei LinuxDatei* kopiert eine Datei von der Diskette.
- ▶ *mwrite LinuxDatei DosDatei* kopiert eine Datei vom Linux-Dateisystem auf die Diskette.

Löschen von Dateien auf Diskette


- ▶ Mit *mdel Dateiname Return* wird eine Datei auf der Diskette gelöscht.
- ▶ Wie üblich wird Laufwerk »A:« angenommen, wenn kein Laufwerk angegeben ist.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Umbenennen von Dateien und Verzeichnissen

- ▶ Der Befehl `mren AlterName NeuerName`  benennt die Datei `AlterName` um.
- ▶ Im Gegensatz zu Windows kann `mren` verwendet werden, um Verzeichnisse umzubenennen.

Diskette formatieren

- ▶ Um eine Diskette zu formatieren, verwenden Sie den Befehl `mformat`.
- ▶ `mformat a:`  formatiert die im Laufwerk liegende Diskette.
- ▶ Um der Diskette einen Namen zu geben, geben Sie hinter `mformat` die Option `»-l«` und den gewünschten Namen an.
- ▶ `mformat` kennt eine Vielzahl weiterer Optionen. Sie erhalten eine Übersicht, wenn Sie `mformat` ohne weitere Angaben aufrufen.

ACHTUNG

Die Diskette muss bereits Low-level-formatiert sein. Ist das noch nicht der Fall, kann dies mit `fdformat /dev/fd0H1440` durchgeführt werden.

Weitere Befehle der mtools


- ▶ Um Dateien anzuzeigen, wird `mtype` verwendet.
- ▶ Um die Dateiattribute zu verändern, wird `mattrib` verwendet.
- ▶ Um die Datenträgerbezeichnung einer Diskette zu ändern, wird `mlabel` verwendet.
- ▶ Mit `mdu` wird der von einem Verzeichnis belegte Platz angezeigt.

Die Konfigurationsdateien

Die `mtools` kennen zwei Konfigurationsdateien:

- ▶ Die allgemeine Konfigurationsdatei ist `/usr/local/etc/mtools.conf`.
- ▶ Für jeden Benutzer kann eine individuelle Konfigurationsdatei namens `.mtoolsrc` in seinem Homeverzeichnis angelegt werden.

Weitere Informationen zu den mtools

- ▶ Die `mtools` beinhalten noch einige zusätzliche seltener benötigte Befehle, die hier nicht angesprochen wurden.
- ▶ Zusätzlich kennen viele der Kommandos noch weitere Optionen.
- ▶ Detaillierte Informationen liefert der Befehl `man mtool s` .

»Mounten« anderer Dateisysteme



[m] [o] [u] [n] [t]

Wo? Womit?

- ▶ Alle unter Linux zugänglichen Dateien werden in einer Verzeichnisstruktur organisiert.
- ▶ Den Ausgangspunkt bildet das *Wurzelverzeichnis* »/«. Alle weiteren Verzeichnisse befinden sich darunter.
- ▶ Die Inhalte dieser Verzeichnisse müssen sich nicht auf denselben Datenträgern befinden. Deshalb können eine zweite Festplatte, ein CD-ROM und beliebige weitere Datenträger in diese Struktur aufgenommen werden.
- ▶ Diesen Vorgang nennt man »Einhängen ins Dateisystem« (engl. *mounten*).
- ▶ Um Datenträger in das Dateisystem einzuhängen, wird der Befehl *mount* verwendet.

Anzeigen der eingehängten Dateisysteme

Um alle momentan verwendeten Dateisysteme zu sehen, rufen Sie den Befehl *mount* ohne weitere Parameter auf:

- ▶ *mount* .
- ▶ Als Ausgabe erhalten Sie eine Übersicht über alle Verzeichnisse, die in das Dateisystem eingehängt wurden.

Einhängen von Datenträgern

Dieser Befehl wird normalerweise in der Form

mount -t Typ Gerät Verzeichnis

verwendet. Es sind also drei Angaben notwendig, um Datenträger ins Dateisystem einzuhängen:

- ▶ Mit *Typ* ist der Typ des Dateisystems gemeint, das auf dem Datenträger verwendet wird. So verwenden Disketten meistens ein *MS-DOS*-, CD-ROMs ein mit *iso9660* bezeichnetes Dateisystem.

INFO

Um Gegensatz zu Windows müssen alle Datenträger zuerst in das Dateisystem eingehängt werden, bevor man auf ihren Inhalt zugreifen kann. Dies gilt auch für Disketten und CD-ROMs.

INFO

Statt »Wurzelverzeichnis« wird oft vom »Root-Directory« bzw. »Root-Verzeichnis« gesprochen.

ACHTUNG

Veränderungen am Dateisystem können Sie nur als Superuser durchführen!

TIP

Mit dem Befehl *df* erhalten Sie eine Anzeige aller eingehängten Festplatten, zusätzlich mit einer Anzeige des freien Speicherplatzes.

TIP

Oft muss man den notwendigen Dateisystem-Typ nicht kennen: Durch Verwendung von »-t autofs« kann Linux den Typ oft selbst feststellen.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Das *Gerät*, das den Datenträger bezeichnet, also Angaben der Art »Floppy-Laufwerk A« oder »IDE-Festplatte 2«. Für jedes mögliche Gerät sind Bezeichnungen vorgesehen.
- ▶ Das *Verzeichnis*, unter dem der Datenträger ansprechbar sein soll. Dieses Verzeichnis muss existieren. Enthält es vorher bereits Daten, so sind diese nach dem Einhängen nicht mehr sichtbar. Enthält der Datenträger Unterverzeichnisse, so erscheinen diese nun als Unterverzeichnisse des angegebenen Verzeichnisses.

→ 97

Verwendung von »umount«

- ▶ Ein eingehängtes Dateisystem kann auch wieder entfernt werden.
- ▶ Dies ist nur möglich, wenn auf dem Datenträger keine offenen Dateien mehr vorhanden sind, d.h. keine Dateien mehr benutzt werden.
- ▶ Auch sollte ein Datenträger nicht ohne Aushängen (engl. *unmounten*) entfernt werden, da dabei Daten verloren gehen können.

INFO

Solange eine CD-ROM nicht ausgehängt wurde, kann sie nicht aus dem Laufwerk entfernt werden. Linux sperrt den Auswurfknopf.

Aushängen eines Datenträgers aus dem Dateisystem:

- ▶ Mit `umount Verzeichnisname` wird das eingehängte Verzeichnis wieder ausgehängt. Das Verzeichnis wird durch das Aushängen nicht entfernt, es besteht nur keine Verbindung zum eingehängten Gerät mehr.
- ▶ Alternativ kann auch der Gerätenamen (statt des Verzeichnisses) angegeben werden.

Die Datei /etc/fstab

Die Datei `/etc/fstab` erfüllt zwei Zwecke:

1. Sie nimmt Ihnen Tipparbeit beim Eingeben des `mount`-Kommandos ab.
2. Sie enthält die Einträge für `mount`, die beim Systemstart automatisch ausgeführt werden sollen.

Diese Datei enthält eine Zeile für jeden `mount`-Eintrag (und ggf. Kommentarzeilen, die mit einem »#« beginnen).

Jede Zeile der Datei besteht aus mehreren Spalten, die durch Leerzeichen oder Tabulatoren getrennt werden. Die einzelnen Spalten haben folgende Bedeutung:

- ▶ *Gerätenamen*
- ▶ *Verzeichnis*
- ▶ *Dateisystem* (oder »auto«, um automatisch ein passendes Dateisystem zu wählen.)

TIP

Ist ein Eintrag für das Verzeichnis, das Sie einhängen möchten, schon in der Datei `/etc/fstab` vorhanden, brauchen Sie bei `mount` nur noch den Verzeichnisnamen eingeben. Alle anderen Informationen werden dann automatisch aus `/etc/fstab` entnommen.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

- ▶ *Zugriffsoptionen* für den Zugriff auf den Datenträger – mehrere Optionen können durch Komma getrennt angegeben werden.
- ▶ Die fünfte Spalte wird zur Zeit von einigen Distributionen ignoriert – hier steht dann immer 0. Ansonsten wird der Wert vom *dump*-Kommando (siehe man-Page) verwendet, um festzustellen, welche Dateisysteme mit *dump* bearbeitet werden müssen.
- ▶ In der letzten Spalte steht einer der Werte 0, 1 oder 2, der angibt, wie das Dateisystem geprüft werden soll. Nur das Wurzelverzeichnis sollte den Wert 1 haben, alle anderen veränderlichen Dateisystem den Wert 2. Der Wert 0 gibt an, dass gar nicht geprüft wird, was zum Beispiel für CD-ROMs sinnvoll ist.

In der dritten Spalte jeder Zeile wird das Dateisystem angegeben. Folgende Dateisysteme sind möglich:

Wert	Bedeutung
<i>ext2</i>	Das Linux-Standarddateisystem
<i>msdos</i>	Das alte DOS-(FAT-)Dateisystem
<i>vfat</i>	Das Dateisystem von Windows9x (VFAT)
<i>iso9660</i>	Dateisystem für CD-ROMs
<i>HPFS</i>	Dateisystem des Betriebssystems OS/2
<i>minix</i>	Dateisystem einer anderen Unix-Variante. Es wird oft für Unix-Disketten verwendet
<i>nfs</i>	Dient dem Zugriff über das Netzwerk
<i>ntfs</i>	Zugriff (nur lesend) auf das Dateisystem von Windows NT 4.0
<i>proc</i>	Virtuelles Dateisystem zur Prozessverwaltung
<i>swap</i>	Dateisystem für Auslagerungsdateien

Die vierte Spalte beinhaltet die Zugriffsoptionen. Hier sind folgende Werte möglich:


• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Option	Bedeutung
block=n	Festlegen der Blockgröße
conv=auto	Konvertieren von DOS-Textdateien automatisch durchführen (DOS verwendet am Zeilenende CR/LF (»Carriage Return«/»Line Feed«), Linux lediglich LF. Dies kann automatisch konvertiert werden.
exec	Programme können von diesem Datenträger gestartet werden.
default	Voreinstellung
gid=n	Falls das Dateisystem keine Besitzerinformationen speichert (Windows): Gruppenzugehörigkeit der Dateien. Diese Option kann nicht für das <i>ext2fs</i> -Dateisystem genutzt werden.
noauto	Datenträger wird beim Starten nicht automatisch eingebunden.
noexec	Es können keine Programme von diesem Datenträger gestartet werden.
ro	Nur lesender Zugriff möglich (read only).
sw	Swap-Datei oder -Partition
sync	Schreibzugriffe werden nicht gepuffert, wodurch das Risiko eines Datenverlustes beim unkontrollierten Ausschalten des Rechners sinkt.
uid=n	Hier wird der Benutzer festgelegt, dem die Dateien »gehören«. Diese Option findet nur Anwendung, wenn das Dateisystem keine Besitzerinformationen speichern kann, und kann z.B. nicht für das <i>ext2fs</i> -Dateisystem genutzt werden.
umask=m	Zugriffs-Bits (invers) der Dateien, falls das Dateisystem diese Information nicht speichert.
user	Jeder Benutzer darf <i>mount</i> und <i>umount</i> (für diesen Eintrag) durchführen.

→ 121

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS


Notwendige Rechte zum Einhängen von Dateisystemen


Grundsätzlich können Sie Veränderungen am Dateisystem nur dann durchführen, wenn Sie als Superuser angemeldet sind.  112

Diese Einschränkung der Rechte normaler Benutzer betrifft auch das Einhängen von Disketten und CD-ROMs, und führt dazu, dass normale Benutzer dies nicht von vornherein dürfen.

Einhängen einer Windows-Platte

Damit Sie unter Linux Ihre Windows-Platte ansprechen können, müssen Sie zuerst zwei Dinge wissen:

1. Wie ist die Platte angeschlossen (d.h. IDE oder SCSI)?  27
2. Welche Partition wird verwendet?

Aufgrund dieser zwei Angaben wählen Sie den Gerätenamen.  97


Gehen Sie nun wie folgt vor:

- ▶ Wählen Sie einen Verzeichnisnamen, unter dem Sie die Windows-Platte ansprechen wollen (zum Beispiel */win*).
- ▶ Wenn das Verzeichnis noch nicht vorhanden ist, erzeugen Sie es mit `mkdir /win`.
- ▶ Hängen Sie die Windows-Partition mit `mount -t vfat Geräteiname /win` in das Dateisystem ein.
- ▶ Überprüfen Sie mit `ls /win`, ob Sie den Inhalt erkennen können.

Einhängen von Disketten

Diskettenlaufwerke werden unter den Gerätenamen */dev/fd0* und */dev/fd1* angesprochen.

Um auf eine Diskette zuzugreifen, gehen Sie folgendermaßen vor:

- ▶ Legen Sie die Diskette ein.
- ▶ Wählen Sie einen Verzeichnisnamen, unter dem Sie auf die Diskette zugreifen wollen – oft ist das Verzeichnis */mnt/floppy* dafür vorgesehen.
- ▶ Falls das Verzeichnis nicht existiert, legen Sie es mit dem Befehl `mkdir` an.
- ▶ Binden Sie die Diskette mit `mount -t auto /dev/fd0 Verzeichnisname` in das Dateisystem ein (Vergessen Sie nicht, den von Ihnen gewählten Verzeichnisnamen einzusetzen).  89

TIP

Tragen Sie das CD-ROM-Laufwerk und das Diskettenlaufwerk in der Datei */etc/fstab* mit der Option »user« ein. Dadurch können auch normale Benutzer CD-ROM- und Diskettenlaufwerk verwenden. Zudem müssen diese Benutzer Mitglied einer Benutzergruppe (oft: *disk*) sein, der das Lesen (und ggf. Schreiben) des entsprechenden Geräts gestattet ist. Für Diskettenlaufwerke stehen alternativ die **mttools** zur Verfügung.

TIP

Wenn in Ihrem Rechner IDE-Platten verwendet werden, bezeichnet man die erste Platte normalerweise mit »hda«. Im Falle von SCSI ist dies »sda«. Daher lautet der Name für die jeweils erste primäre Partition »hda1« bzw. »sda1«.

TIP

Sie können Disketten wie bereits beschrieben »mounten«, es gibt aber einen einfacheren Weg: Durch Verwendung der **mttools** wird die Arbeit mit Diskettenlaufwerken erleichtert.

INFO

Bei einer SuSE-Distribution ist das Verzeichnis »floppy« bereits vorhanden, bei Red-Hat ist dies »mnt/floppy«.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Wenn Sie die Diskette wieder entnehmen wollen, führen Sie vorher den Befehl `umount Verzeichnisname` aus.

Einhängen von CD-ROMs

Um auf das CD-ROM zuzugreifen, gehen Sie wie folgt vor:

- ▶ Entscheiden Sie, an welchem Anschluss die CD-ROM angeschlossen ist, also welcher Geräte-Name zu wählen ist (siehe dazu die nachfolgende Tabelle).
- ▶ Entscheiden Sie, mit welchem Verzeichnis Sie die CD-ROM ansprechen möchten. Oft ist das Verzeichnis `»/mnt/cdrom«` bzw. `»/cdrom«` schon vorhanden.
- ▶ Falls das von Ihnen gewählte Verzeichnis (zum Beispiel `/cdrom`) noch nicht vorhanden ist, legen Sie es mit `mkdir Verzeichnisname` (`Verzeichnisname` ist also zum Beispiel `»/cdrom«`) an.
- ▶ Binden Sie die CD mit `mount -t iso9660 -o ro Geräte-Name Verzeichnisname` in das Dateisystem ein.
- ▶ Wenn Sie die CD wieder entnehmen wollen, geben Sie `umount Verzeichnisname` ein.

TIP

Statt den richtigen Geräte-Name zu wissen, kann meist auch der Name `/dev/cdrom` verwendet werden. Dabei handelt es sich um einen Link auf das richtige Gerät, der bei der Installation eingerichtet wurde.

Gerätebezeichnungen

Disketten

Gerät	Gerätebezeichnung
Diskette A:	<code>/dev/fd0</code>
Diskette B:	<code>/dev/fd1</code>

Festplatten

Gerät	Gerätebezeichnung
Festplatte am IDE-Adapter	<code>/dev/hdgp</code>
Festplatte am SCSI-Adapter	<code>/dev/sdgp</code>

- ▶ Die Buchstaben `gp` sind Platzhalter für die Plattennummer und die Partitionsnummer. Wenn beide entfallen, ist die gesamte Platte gemeint.
- ▶ Die Plattennummer wird durch einen Buchstaben (`»a«`, `»b«` usw.) und die Partitionsnummer durch eine Ziffer beginnend ab 1 bezeichnet.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

- ▶ Bei der Partitionsnummer sind 1 bis 4 für die primären oder eine erweiterte Partition reserviert, Nummern ab 5 für logische Partitionen.

Beispiele:

- ▶ */dev/hda* – gesamte erste IDE-Platte.
- ▶ */dev/sda1* – die erste SCSI-Platte und darauf die erste primäre Partition.
- ▶ */dev/sdb5* – die erste logische Partition auf der zweiten SCSI-Platte.

CD-ROM- Laufwerke

Gerät	Gerätebezeichnung
Voreingestelltes CD-ROM-Gerät	<i>/dev/cdrom</i>
IDE/ATAPI-Adapter 1, Laufwerk 1	<i>/dev/hdc</i>
IDE/ATAPI-Adapter 1, Laufwerk 2	<i>/dev/hdb</i>
IDE/ATAPI-Adapter 2, Laufwerk 1	<i>/dev/hdc</i>
IDE/ATAPI-Adapter 2, Laufwerk 2	<i>/dev/hdd</i>
SCSI-CD-ROM 1	<i>/dev/scd0</i>
SCSI-CD-ROM 1	<i>/dev/scd1</i>

Archivieren mit tar, gzip und bzip2



Wo? Womit?

[t] [a] [r]

[g] [z] [i] [p]

- ▶ Dateien, die auf die Zeichenkombination ».tar« enden, sind sogenannte Archive. Archive bestehen aus einzelnen Dateien, die zusammen mit ihren Pfadinformatio- nen in einer großen Datei zusammengefasst wurden. Solche Archive werden mit dem Programm *tar* bear- beitet.
- ▶ Dateien die auf ».tgz« oder ».tar.gz« enden, sind Archive, die mit dem Kompressionsprogramm *gzip* gepackt (also komprimiert) wurden.
- ▶ Das Programm *gzip* wird von *tar* automatisch aufgeru- fen, wenn entsprechende Optionen verwendet werden.
- ▶ Archive, die mit dem – effizienteren, aber dafür lang- sameren – Kompressionsprogramm *bzip2* gepackt wur- den, haben meistens die Endung ».tar.bz«.

INFO

Es ist nicht erforderlich, Archive mit »...tar« oder bestimmten Endungen zu benennen. Trotzdem wird es meistens so gemacht, da dadurch sofort ersichtlich ist, dass es sich um Archive handelt.

Ein Archiv anlegen

- ▶ Ein Archiv wird mit dem Befehl `tar -cf Archivname Verzeichnisname` angelegt. Dadurch werden alle Dateien und Unterverzeichnisse im angegebenen Ver- zeichnis im Archiv gespeichert.
- ▶ Um das Archiv zu komprimieren, verwenden Sie statt »-cf« die Optionen »-czf«.

TIP

Es ist sinnvoll, den Archiv- namen mit der Endung ».tar« zu versehen, also zum Bei- spiel »test.tar«. Kompri- mierte Archive sollten mit ».tgz« oder ».tar.gz« enden.

Den Inhalt eines Archivs anzeigen

- ▶ Der Inhalt einer Archivdatei wird mit `tar -tf Archiv- name` angezeigt.
- ▶ Handelt es sich um ein komprimiertes Archiv, dann muss statt »-tf« die Option »-tzf« verwendet werden.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Ein Archiv extrahieren

- ▶ Der Befehl `tar -xf Archivname -C Verzeichnis` extrahiert die Dateien aus dem Archiv in das angegebene Verzeichnis.
- ▶ Wenn es sich um ein komprimiertes Archiv handelt, muss statt »-xf« die Option »-xzf« verwendet werden.
- ▶ Um in das aktuelle Verzeichnis zu entpacken, müssen die Verzeichnisangabe und die Option »-C« weggelassen werden.
- ▶ Um nur bestimmte Dateien zu extrahieren, geben Sie das entsprechende Namensschema in Hochkommata hinter dem Archivnamen an (zum Beispiel '*.txt').

ACHTUNG

Achten Sie auch hier auf Groß- und Kleinschreibung: »-C« ist »Groß« geschrieben!

Einzelne Dateien komprimieren

Einzelne Dateien werden mit dem Programm *gzip* komprimiert und mit dem Programm *gunzip* entpackt.

- ▶ Um eine einzelne Datei zu komprimieren, lautet der Befehl: `gzip Dateiname`.
- ▶ Das Ergebnis ist die Datei »Dateiname.gz«, d.h., ».gz« wird automatisch angehängt.
- ▶ Um mehrere Dateien zu komprimieren, müssen Jokerzeichen verwendet werden (zum Beispiel `gzip *.bin` – dies erzeugt für alle Dateien im aktuellen Verzeichnis, die auf .bin enden, entsprechende ».bin.gz« Dateien).

Nach der Ausführung von *gzip* ist die unkomprimierte Originaldatei nicht mehr vorhanden.

- ▶ Damit die Originaldatei nicht entfernt wird, muss die Option »-c« verwendet werden. Dadurch wird das Ergebnis jedoch auf der Standardausgabe (dem Bildschirm) ausgegeben.
- ▶ Diese Ausgabe leiten Sie mit »> Dateiname« vom Bildschirm in die angegebene Datei um.
- ▶ So sieht der gesamte Befehl in diesem Fall aus: `gzip -c Eingabedatei > Ausgabedatei`.

Einzelne Dateien dekomprimieren

- ▶ Zum Dekomprimieren einer einzelnen Datei wird der Befehl `gunzip Dateiname.gz` verwendet.
- ▶ Das Ergebnis ist die neue Datei »Dateiname«, d.h., die Endung ».gz« wird automatisch entfernt.
- ▶ Um mehrere Dateien zu dekomprimieren müssen Jokerzeichen verwendet werden.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO**Verwendung von bzip2**

Das Programm *bzip2* erfüllt den gleichen Zweck wie *gzip*, erreicht jedoch eine bessere Kompressionsrate.

- ▶ Die Kompressionsrate wird durch Angabe von -1 bis -9 hinter dem Befehl *bzip2* festgelegt.
- ▶ -9 bietet die stärkste Komprimierung (d.h. die Dateien werden am kleinsten), benötigt jedoch auch temporären Platz auf der Platte, um die Komprimierung durchzuführen.
- ▶ Komprimierte Dateien dieses Programms sind an der Endung »bz2« erkennbar.

tar in Verbindung mit bzip2

- ▶ Um ein Archiv zu dekomprimieren, das mit *bzip2* komprimiert wurde (erkennbar an der Endung ».bz2«) verwenden Sie den Befehl `bunzip2 Archivname`.

ACHTUNG

Auch gzip kennt diese Optionen zur Festlegung der Kompressionsrate.

ACHTUNG

Neuere tar-Versionen können schon von Haus aus mit bzip2 umgehen. Schauen Sie mit `tar --help | less` nach, ob Ihr tar eine entsprechende Option bereithält. Beenden können Sie die Anzeige mit `q`.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Hilfe suchen – man

[m] [a] [n]

Wo? WoMIT?

Der Befehl *man* ist das zentrale Informationselement unter Linux. Er liefert die »Manual Pages«, die Hilfeseiten der meisten Befehle.

Informationen über ein bestimmtes Thema abrufen

Um Informationen über einen bestimmten Befehl zu erhalten, muss der gewünschte Befehl als Argument hinter *man* angegeben werden.

- ▶ `man Befehl` zeigt die zum entsprechenden Befehl vorhandenen Erklärungen an.
- ▶ Durch Drücken von `↑` und `↓` kann der Text auf- und abgerollt werden.
- ▶ Durch Drücken von `q` wird die Anzeige verlassen.
- ▶ Es stehen Ihnen weiterhin alle Möglichkeiten offen, die Sie auch beim Anzeigen eines Textes durch *less* haben.
- ▶ Wenn Sie den Text in einer Datei speichern möchten, können Sie dies durch `man Thema > Dateiname` erreichen.

INFO

Einige Programme bieten Ihre Beschreibung nicht mit `man`, sondern mit dem Befehl `info an`, der im Gegensatz zu `man` auch Querverweise innerhalb der Texte erlaubt.

Verschiedene Bereiche anzeigen

- ▶ Wenn Sie vor dem Namen des Themas, zu dem Sie mehr wissen, eine Bereichsnummer angeben, so wird die Suche auf diesen Bereich eingeschränkt.

Folgende Bereiche gibt es:

Bereich	Bedeutung
1	Benutzerkommandos
2	Systemaufrufe
3	Funktionen der Programmiersprache »C«
4	Dateiformate, Geräte-Dateien
5	Konfigurationsdateien

TIP

In der Unix-Literatur wird oft eine entsprechende man-Nummer in Klammern hinter dem Kommando angegeben. Geben Sie diese Nummer beim Abfragen von Informationen mit `man` vor dem Befehlsnamen ein.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Bereich	Bedeutung
6	Spiele
7	Diverses
8	Systemadministrationskommandos
9	Kernelfunktionen
n	Neue Kommandos

Weitere Möglichkeiten mit man

Durch die Angabe von zusätzlichen Optionen hinter dem Befehl `man` kann das Verhalten von `man` zusätzlich beeinflussen:

- ▶ `man -a [Bereich] Thema` zeigt alle gleichnamigen Seiten an, sonst wird für gewöhnlich nur die erste gefundene Seite angezeigt.
- ▶ `man -f Thema` zeigt nur eine einzeilige Erklärung zum angegebenen Thema an.
- ▶ `man -k Thema` zeigt alle vorhandenen `man`-Texte an, in denen das Thema vorkommt.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Dateien editieren – der Editor joe

```

Help Screen      turn off with ^KH      more help with ESC . (^[,.)
CURSOR           GO TO           BLOCK           DELETE           MISC           EXIT
^B left ^F right ^U prev. screen ^KB begin ^D char. ^KJ reformat ^KX save
^P up ^N down ^V next screen ^KK end ^Y line ^T options ^C abort
^Z previous word ^A beg. of line ^KM move ^W >word ^R refresh ^KZ shell
^X next word ^E end of line ^KC copy ^O word< ^@ insert FILE
SEARCH           ^KU top of file ^KW file ^J >line SPELL           ^KE edit
^KF find text ^KV end of file ^KY delete ^_ undo ^[N word ^KR insert
^L find next ^KL to line No. ^K/ filter ^^ redo ^[L file ^KD save
IW Unnamed Row 1 Col 1 5:56 Ctrl-K H for help
    
```


Wo? WoMIT?

- ▶ Unter Linux gibt es eine Vielzahl von Editoren zum Bearbeiten von Textdateien.
- ▶ Der Editor *joe* zeichnet sich vor allem durch seine einfache Bedienung aus, da seine Kommandos an das Textverarbeitungsprogramm *Wordstar* angelehnt sind.

INFO

Weitere Editoren sind zum Beispiel *vi*, *vim*, *elvis*, *pico* und der sehr leistungsstarke, aber komplizierte *emacs*. Mithilfe von man erhalten Sie Informationen über diese Editoren.

Dateien bearbeiten

- ▶ Um mit *joe* eine Datei zu bearbeiten, geben Sie einfach den Dateinamen hinter dem Aufruf von *joe* an: *joe Dateiname*  zeigt die entsprechende Datei am Bildschirm an, so dass sie bearbeitet werden kann.
- ▶ Sie können auch mehrere Dateinamen angeben. Dadurch wird der Bildschirm aufgeteilt und jede Datei in einem eigenen Bereich angezeigt.
- ▶ Wenn Sie Hilfe zu den Tastaturkürzel benötigen, drücken Sie **Strg+K**, **H**.
- ▶ Drücken Sie **Strg+K**, **X**, um Ihre Änderungen in der Datei zu speichern.

TIP

Wenn Sie *joe* mit der Option *-asis* (also *joe -asis Dateiname*) – aufrufen, werden auch alle Sonderzeichen und Umlaute korrekt dargestellt.

Hilfe zur Tastenbelegung

Das wichtigste Tastaturkürzel für Anwender, die mit *joe* noch unerfahren sind, ist **Strg+K**, **H**, denn diese Tastenkombination zeigt den Hilfebildschirm an:

```

Help Screen      turn off with ^KH      more help with ESC . (^[,.)
CURSOR           GO TO           BLOCK           DELETE           MISC           EXIT
^B left ^F right ^U prev. screen ^KB begin ^D char. ^KJ reformat ^KX save
^P up ^N down ^V next screen ^KK end ^Y line ^T options ^C abort
^Z previous word ^A beg. of line ^KM move ^W >word ^R refresh ^KZ shell
^X next word ^E end of line ^KC copy ^O word< ^@ insert FILE
SEARCH           ^KU top of file ^KW file ^J >line SPELL           ^KE edit
^KF find text ^KV end of file ^KY delete ^_ undo ^[N word ^KR insert
^L find next ^KL to line No. ^K/ filter ^^ redo ^[L file ^KD save
IW Unnamed Row 1 Col 1 5:56 Ctrl-K H for help
    
```

TIP

joe emuliert mehrere populäre Editoren. Die *joe*-Programmdatei kann deshalb auch unter den Namen *jstar*, *jmacs*, *rojoe*, *jpico* aufgerufen werden.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Bewegen und Ändern im Text

Die wichtigsten Tastenkürzel sind:

Tastenkürzel	Bedeutung
und	Cursor eine Zeile nach oben bzw. nach unten.
und	Cursor nach links bzw. rechts bewegen.
	Zeichen am Cursor löschen.
	Umschalten zwischen Einfügen und Überschreiben.
	Zeichen links vom Cursor löschen.
	Neue Zeile einfügen.
	Bewegt den Cursor an den Zeilenanfang.
	Bewegt den Cursor an das Zeilenende.
	Cursor an den Textanfang bringen.
	Cursor an das Textende bringen.
	Die aktuelle Zeile löschen.
	Vom Cursor bis zum Zeilenende löschen.

Laden und Speichern von Dateien

- ▶ Um eine neue Datei zu laden, drücken Sie und geben Sie den gewünschten Dateinamen ein.
- ▶ Speichern Sie die Datei mit , wodurch *joe* auch gleichzeitig beendet wird.
- ▶ Wenn Sie Ihre Änderungen verwerfen wollen, drücken Sie , um den Editor zu beenden.

	Speichert die Datei und beendet das Programm.
	Bricht das Editieren ab – nach einer Nachfrage wird der Editor verlassen, ohne die Änderungen zu speichern.
	Lädt eine neue Datei.
	Speichert die aktuelle Datei ab, ohne den Editor zu verlassen.

TIP

Wann immer *joe* einen Dateinamen erwartet, können Sie drücken, um einen Dateinamen automatisch zu vervollständigen. Ist dies nicht möglich, hören Sie einen Signalton. Ein erneutes Drücken von zeigt Ihnen dann das aktuelle Verzeichnis.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Strg + **K**, **R** Fügt eine andere Datei an der aktuellen Cursorposition ein.

Bearbeiten von Blöcken

- ▶ Bewegen Sie den Cursor an den Anfang der gewünschten Textstelle.
- ▶ Drücken Sie **Strg** + **K**, **B**.
- ▶ Bewegen Sie den Cursor ans Ende der gewünschten Textstelle.
- ▶ Drücken Sie **Strg** + **K**, **K**.
- ▶ Der gewählte Bereich wird nun invers dargestellt.

BEGRIFF

*Sie können einen Teil des Textes markieren, um diesen Teil zu bearbeiten. Dies ist ein **Block**.*

Der Block ist damit markiert. Sie haben nun verschiedene Möglichkeiten:

- ▶ Drücken Sie **Strg** + **K**, **Y**, um den gesamten Bereich zu löschen.
- ▶ Bewegen Sie den Cursor an eine andere Textstelle und drücken Sie **Strg** + **K**, **M**, um den gesamten markierten Text an diese Stelle zu bewegen, oder **Strg** + **K**, **C** um an dieser Stelle eine Kopie des markierten Bereiches anzufertigen.

Strg + **K**, **B** Markiert den Anfang eines Blocks.

Strg + **K**, **K** Markiert das Ende des Blocks.

Strg + **K**, **B** Hebt die Markierung eines Blocks wieder auf, wenn die Tastenkombination am Blockende gedrückt wird.

Strg + **K**, **M** Markierten Block an die aktuelle Cursorposition verschieben.

Strg + **K**, **C** Markierten Block an die aktuelle Cursorposition kopieren.

Strg + **K**, **Y** Markierten Block löschen.

Suchen und Ersetzen von Text

- ▶ Um einen Text in der aktuellen Datei zu suchen, drücken Sie **Strg** + **K**, **F**.
- ▶ Geben Sie nun den zu suchenden Text ein, und drücken Sie **↵**.
- ▶ Sie werden nach Suchoptionen gefragt. Drücken Sie hier einfach **↵**.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

- ▶ Der Cursor springt nun zum ersten Fundort des gesuchten Textes.
- ▶ Um weitere Vorkommen des Textes zu suchen, drücken Sie **Strg**+**L**.

Strg + K , F	Text suchen.
Strg + L	Nächstes Vorkommen eines gefundenen Textes suchen.

Beim Suchen von Text werden Sie nach Suchoptionen gefragt. Geben Sie hier einen oder mehrere Buchstaben ein:

- ▶ »b« – Sucht ab dem Cursor rückwärts.
- ▶ »i« – Keine Unterscheidung zwischen Groß- und Kleinschreibung. Normalerweise wird zwischen Groß- und Kleinschreibung unterschieden.
- ▶ »nnn« – Sie können hier auch eine Zahl eingeben. Dann wird das nnn-te Vorkommen des Textes gesucht.
- ▶ »r« – Ersetzen von Text. Wenn Sie diesen Buchstaben eingeben, werden Sie anschließend nach dem Text gefragt, durch den der gefundene Text ersetzt werden soll.

TIP
 Sie können beim Suchen auch reguläre Ausdrücke verwenden. Die man-Page von joe gibt hierzu detaillierte Auskünfte.

Arbeiten mit mehreren Fenstern

- ▶ Wenn beim Aufruf des Editors mehrere Dateinamen angegeben werden, so wird der Bildschirm in mehrere Teilbereiche (Fenster) unterteilt, von denen jeder eine Datei anzeigt.
- ▶ Ein neues Fenster kann jederzeit mit **Strg**+**K**, **E** geöffnet werden.
- ▶ Fenster lassen sich in der Größe anpassen und individuell bearbeiten.

Tastencodes für das Arbeiten mit mehreren Fenstern:

Tastenkürzel	Bedeutung
Strg + K , O	Fenster splitten.
Strg + K , G	Aktuelles Fenster um eine Zeile vergrößern.
Strg + K , T	Aktuelles Fenster um eine Zeile verkleinern.
Strg + K , N	Zum Fenster weiter unten.
Strg + K , P	Zum Fenster weiter oben.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

Tastenkürzel	Bedeutung
Strg + K , E	Datei in das aktuelle Fenster laden.
Strg + K , I	Umschalten zwischen der Anzeige aller Fenster und des aktuellen Fensters.
Strg + C	Fenster schließen.

Weitere praktische Tastencodes:

Tastenkürzel	Bedeutung
Strg + Z	Cursor zum vorigen Wort.
Strg + X	Cursor zum nächsten Wort.
Strg + K , L	Cursor zu einer bestimmten Zeilennummer.
Strg + T	Einstellungs-Bildschirm aufrufen.

INFO

Weitere, nicht so oft benötigte Tastenkürzel werden in der man-Page zu joe aufgeführt.

Kommandozeilenoptionen

- ▶ Sie können zusätzlich zu den Dateinamen Optionen beim Aufruf von *joe* festlegen, die das Verhalten des Editors ändern.
- ▶ Diese Optionen müssen vor den Dateinamen angegeben werden.

Einige wichtige Optionen:

Option	Bedeutung
-asis	Sonderzeichen und Umlaute werden auch angezeigt.
-backpath Verzeichnis	Sicherungsdateien werden im angegebenen Verzeichnis gespeichert.
-lightoff	Die Block-Markierung wird wieder aufgehoben, sobald ein Block-Kommando durchgeführt wurde. Normalerweise muss Strg + K , B am Ende des Blocks gedrückt werden, um die Markierung wieder aufzuheben.

• ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDO

Option	Bedeutung
-crlf	Diese Option kann unmittelbar vor dem Dateinamen eingegeben werden – bei mehreren Dateien für jeden individuell. Diese Option sollte benutzt werden, wenn Windows-Textdateien bearbeitet werden.
-wordwrap	Auch diese Option kann individuell für jede Datei angegeben werden. Sie bewirkt, dass Wörter am Zeilenende wie bei einem Textverarbeitungsprogramm umbrochen werden. Normalerweise wird das Eingabefenster nach rechts gerollt, und Zeilen können beliebig lang werden.

Die Konfigurationsdateien

- ▶ Die Einstellungen, nach denen *joe* sich richtet, werden in der Datei »/usr/local/lib/joerc« abgelegt.
- ▶ Für jeden Benutzer kann eine individuelle Konfigurationsdatei erstellt werden.

Zum Erstellen einer Konfiguration für einen einzelnen Benutzer gehen Sie wie folgt vor:

- ▶ Kopieren Sie die Datei »/usr/local/lib/joerc« unter dem Namen ».joerc« in das Homeverzeichnis des Benutzers.
- ▶ Verändern Sie in der Datei ».joerc« die gewünschten Einstellungen.
- ▶ Diese Einstellungen gelten nun, wenn der jeweilige Benutzer *joe* verwendet.

ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS • ESSENTIELLE KOMMANDOS

BEGRIFF

Das oberste Verzeichnis wird als **Wurzel-(Root-)verzeichnis** bezeichnet.

INFO

Unter Linux hat jeder Benutzer ein eigenes Benutzerverzeichnis, das für die Ablage seiner persönlichen Daten vorgesehen ist. Normalerweise findet sich dieses Homeverzeichnis unter `/home/Benutzername`.

BEGRIFF

Wildcards sind »Jokerzeichen«, die man in Dateinamen angeben kann, um eine Reihe von möglichen Kombinationen darzustellen.

INFO

Um Gegensatz zu Windows müssen alle Datenträger zuerst in das Dateisystem eingehängt werden, bevor man auf ihren Inhalt zugreifen kann – dies gilt auch für Disketten und CD-ROMs.

Datenträgerverwaltung

- ▶ Um Datenträger in das Dateisystem einzuhängen, wird der Befehl *mount* verwendet.
- ▶ Eine bequemere Alternative, mit Disketten umzugehen, bieten die sogenannten *mtools*. Dabei handelt es sich um ein einziges Programm, das aber unter verschiedenen Namen angesprochen wird.
- ▶ Dateien, die auf die Zeichenkombination »tar« enden, sind sogenannte Archive. Archive bestehen aus einzelnen Dateien, die zusammen mit ihren Pfadinformationen in einer großen Datei zusammengefasst wurden. Solche Archive werden mit dem Programm *tar* bearbeitet.

ACHTUNG

Linux unterscheidet zwischen Groß- und Kleinschreibung. Bei Dateinamen und Pfadnamen muss genau auf die Schreibweise geachtet werden: *datei* und *Datei* sind zwei verschiedene Namen!

Dateiverwaltung

- ▶ Um die in einem Verzeichnis enthaltenen Dateien anzuzeigen, geben Sie ein: `ls Verzeichnispfad` [1].

Programme wie *more* und *less* bezeichnet man als Pager – sie dienen zum Anzeigen von Dateien.

- ▶ Dateien werden mit dem Befehl *cp* kopiert.
- ▶ Mit `mv Datei Zielverzeichnis` [1] wird eine Datei verschoben.
- ▶ Mit `rm Datei` [1] löscht man die angegebene Datei.
- ▶ `rmdir Verzeichnis` [1] löscht das angegebene Verzeichnis.
- ▶ Um Links einzurichten, wird das Kommando *ln* verwendet.
- ▶ *find* durchsucht (falls nichts anderes angegeben ist: ab dem aktuellen Verzeichnis) die Unterverzeichnisse nach bestimmten Suchkriterien.

Hilfe unter Linux

- ▶ Der Befehl *man* ist das zentrale Informationselement unter Linux. Er zeigt die »Manual Pages«, die Beschreibungsseiten der meisten Befehle, an.

Editoren

- ▶ Unter Linux gibt es eine Vielzahl von Editoren.
- ▶ Der Editor *joe* zeichnet sich vor allem durch seine einfache Bedienung aus, da seine Kommandos an das Textverarbeitungsprogramm *Wordstar* angelehnt sind.