

# Inhaltsverzeichnis


<b>Konventionen</b> .....	<b>2</b>
<b>Kommandoüberblick nach Funktionen</b> .....	<b>3</b>
Rechner ein- und ausschalten .....	3
Benutzer- und Systemumgebung .....	3
Dateiverwaltung (kombinierte Anwendung im Konqueror) .....	4
Druckausgabe .....	5
Editoren .....	6
Hilfe .....	6
PC-Datenaustausch .....	6
Prozessverwaltung .....	7
Shell-interne Kommandos .....	7
Übersicht der Kommandos, aus dem Buch ›Linux-Systemadministration‹: .....	8
<b>Kommandos alphabetisch</b> .....	<b>17</b>
<b>Editoren</b> .....	<b>47</b>
Metazeichen – reguläre Ausdrücke (ed, grep, sed, vi) .....	47
Häufig benutzte Kommandos im ed (ex, sed) .....	48
Häufig benutzte Kommandos im vi .....	50
Einige praktische Optionen für den vi .....	55
Befehlszeileneditor im vi, emacs und der bash .....	56
<b>Wichtige Dateien für den Benutzer</b> .....	<b>57</b>
<b>Shell: Bourne (sh), Korn (ksh), Bash (bash)</b> .....	<b>60</b>
Sonderzeichen .....	60
Variable .....	61
Positionsparameter .....	61
Vordefinierte Shell-Variable .....	62
Arbeiten mit Variablen .....	64
Setzen von Optionen .....	65
Konstrukte/Ablaufsteuerung der Shell .....	66
<b>Besonderheiten der Bash</b> .....	<b>68</b>
<b>C-Shell (csh und tcsh)</b> .....	<b>71</b>
Unterschiede zur sh, ksh und bash .....	71

# Konventionen der Kurzreferenz

Darstellung der Kommando-eingabe:

<b>in Fettdruck</b>	Aufruf des Kommandos
[ ]	Mögliche Optionen. Die Klammer selbst wird bei der Kommando-eingabe nicht mitgeschrieben
\	Nur aus Platzgründen wird hier die Kommando-eingabe in der nächsten Zeile fortgesetzt. Das Kommando kann ohne dieses Zeichen in einer Zeile eingegeben werden
in <i>Kursiv</i>	Einsetzen von entsprechenden Namen
<i>Kursiv-Trennung</i>	Das Kommando wird hier nur aus Platzgründen in der nächsten Zeile fortgesetzt
<b>in <i>Kursiv fett</i></b>	Bei Beispielen sind hier Eigennamen gekennzeichnet

Darstellung unter Funktion:

<i>Kursiv</i>	Ableitung vom englischen Ausdruck
<b>in Fettdruck</b>	Hauptmerkmale des Kommandos
→	Steht für: ›siehe auch‹
<b><i>nicht sh</i></b>	Gelten bestimmte Kommandos nicht für die Bourne-Shell, ist dies hiermit gekennzeichnet
 <b>Dateien Suchen</b>	Hinweise auf eine Menüfolge unter der grafischen Oberfläche KD

## Kommandoüberblick nach Funktionen

(mit Hinweisen auf einige grafisch aufbereitete Programme unter SUSE Linux KDE<sup>1</sup>)

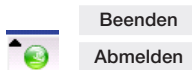
### Rechner ein- und ausschalten

Generell sollten UNIX-Rechner nur vom Systemverwalter ein- oder ausgeschaltet werden, da u.U. wichtige Entscheidungen bei Prüfroutinen getroffen werden müssen. Viele Rechner fahren das System automatisch hoch.

Bei einigen Rechnern geben Sie **boot**  
ein, um UNIX zu starten oder **init 5 [1,2,3]**  
Um einen Rechner neu zu starten **reboot**

Um einen UNIX-Rechner auszuschalten, müssen vorab alle Prozesse abgeschlossen werden. Dies erfolgt mit dem Kommando **shutdown**  
Dieses Kommando darf nur vom Benutzer **root** (Systemverwalter) gestartet werden, der sich hierfür an der Systemkonsole anmeldet.

Als Benutzer z.B. bei SUSE Linux über das Anmeldefenster, bzw.



Bei einigen Rechnern werden, bevor der Rechner ausgeschaltet werden darf, noch zusätzliche Kommandos verlangt:  
z.B. zum Synchronisieren **sync**  
oder **halt**

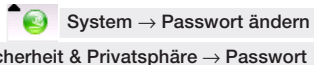
### Benutzer- und Systemumgebung

Informationen über die angemeldeten Benutzer und Ihre eigene Umgebung liefern Ihnen

**finger, tty, who**

Das Passwort ändern Sie mit **passwd**  
oder mit

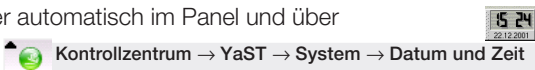
Unter SUSE bis 9.1,  
ab 9.2



Kalender-Informationen, Datum und Uhrzeit zeigen Ihnen

**cal, date**

oder automatisch im Panel und über



1. Die Menüfolge richtet sich nach der jeweiligen Installation. Die Beispiele sind hier auf SUSE Linux 9.2 abgestimmt.

Kommandos zu bestimmten Uhrzeiten  
starten Sie einmalig mit **at**

Den Bildschirm können Sie löschen mit **clear**

Die Versionsnummer des UNIX-Systems  
erhalten Sie mit **uname**

### Dateiverwaltung

#### (kombinierte Anwendung im Konqueror)



Den Namen des aktuellen Directories  
erhalten Sie mit **pwd**

Die Inhaltsliste eines Directories und Attribute von  
Dateien zeigt Ihnen **ls**

Den Inhalt einer Datei können Sie  
ansehen mit **cat, head, pg, more, less, tail**



Dienstprogramme → Editor

Je nach Inhalt der Datei z. B. (pdf-Dateien):



Büroprogramme → Dokumentenansicht

bzw. automatische Zuordnung beim Anklicken  
der Datei oder rechte Maustaste: Öffnen mit



Die Art einer Datei bzw. des Inhalts stellen  
Sie fest mit **file**

oder automatisch in Statuszeile

In ein anderes Directory wechseln Sie mit **cd**

oder durch Anklicken in das Verzeichnis

Directories legen Sie neu an mit **mkdir**

oder über rechte Maustaste: Neu Erstellen

Dateien im Dateibaum suchen Sie mit **find**

oder über Dateien Suchen

Inhalte von Dateien (Wörter, Zeichenketten)  
suchen Sie mit **grep**



oder über Dateien Suchen

Dateien kopieren, sichern können Sie mit **cp, cpio, tar, gzip, zip**

oder durch Drag and Drop


Dateien umbenennen bzw. in ein anderes  
Directory verschieben mit **mv**

oder durch Drag and Drop


Dateien oder Directories löschen mit **rm** oder **rmdir**  
oder durch Drag and Drop  

Den freien oder belegten Plattenplatz  
erfragen Sie mit **df, du**

 **System** → **Dateisystem** → **KDiskFree**

Zugriffsrechte verändern Sie mit **chmod**  
oder Datei auswählen: Eigenschaften 

Dateibesitzer verändern Sie mit **chown**

Gruppe verändern Sie mit **chgrp**  
oder Datei auswählen: Eigenschaften 

Spalten bzw. Zeichen filtern Sie aus  
einer Datei heraus mit **cut**

oder  **Dienstprogramme** → **Editor**

Dateiinhalte oder Zeichenketten sortieren  
Sie mit **sort**

Das Datum einer Datei ändern Sie mit **touch**  
(existiert die Datei noch nicht, wird sie angelegt)

Dateiinhalte zweier Dateien vergleichen Sie mit **cmp**  
oder **diff**

Dateien komprimieren Sie mit **compress,**  
**zip, gzip**  
dekomprimieren mit **decompress, unzip, gunzip**  
bzw. **gzip -d**

Platten/Disketten montieren (einhängen,  
aushängen) Sie mit **mount, umount**

 **System** → **Dateisystem** → **KDiskFree**  
Gerät auswählen, rechte Maustaste


## Druckausgabe

Dateien bereiten Sie für den Druck auf mit **pr**

Dateien können Sie ausdrucken mit **lp, lpr**

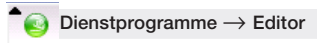
Den Status der einzelnen Druckerqueues/  
Druckaufträge verwalten Sie mit **lpstat, lpq**

und löschen Sie mit **cancel, lprm**  
oder über

 **Dienstprogramme** → **Drucken** → **Druckerverwaltung**

## Editoren

Dateien oder Zeichenketten editieren Sie mit **ed, sed, vi, vim, emacs** oder mit dem grafischen Texteditor (CDE) oder unter Linux KDE:



## Hilfe

Eine Beschreibung der Kommandos und/oder Systemdateien erhalten Sie über **man**

Eine Kurzinformation über das Kommando gibt Ihnen **whatis**

Den absoluten Pfadnamen eines Kommandos erhalten Sie mit **whereis**

Oder Sie benutzen die grafisch aufbereitete Online-Hilfe, z.B. bei IBM **info**

oder unter Linux im Panel **(khelpcenter)**



Welche Kommandos bestimmte Themen behandeln, erhalten Sie mit **man -k Thema**

## PC-Datenaustausch

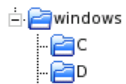
Über die grafische Oberfläche (CDE) bei den meisten UNIX-Anbietern im Dateimanager-Menü integriert

Solaris: Einhängen der Floppy *Open Floppy*

Solaris: Abmelden der Floppy *Eject*

Bei SUSE Linux:

Parallele Anwendung von WindowsXX/Floppy



Symbol auf dem Desktop anklicken (bzw. Desktop → Arbeitsplatz)



Kopieren und Löschen wie bei UNIX-Dateien

Den Inhalt einer DOS-Floppy zeigt Ihnen **dosdir\***  
**mdir\*\***

oder über den Konqueror



Eine Floppy können Sie im DOS-Format formatieren mit

**dosformat\***  
**mformat\*\***

Eine Datei kopieren Sie von einer DOS-Floppy nach UNIX mit

**dosread\***  
**mcopy\*\***

Eine UNIX-Datei kopieren Sie auf eine DOS-Floppy mit

**doswrite\***  
**mcopy\*\***

oder über den Konqueror mit Drag and Drop



## Prozessverwaltung

Welche Prozesse zur Zeit laufen, zeigt Ihnen

**ps**

oder



System → Überwachung → Systemüberwachung

Einen Prozess können Sie abbrechen mit

**kill**

oder



System → Überwachung → Systemüberwachung

über rechte Maustaste (kill)

über Schaltfläche (kill)

## Shell-interne Kommandos

Shell-Variable und Shell-Ablaufsteuerung (ab Seite 60)

Hier nur die Aufrufe zur Shell

Die Shell (Subshell) starten Sie für

- Kornshell mit **ksh**
- C-Shell mit **csh**
- tcsh-Shell mit **tcsh**
- BourneShell **sh**
- Bash **bash**

Eine Shell beenden Sie mit

**exit**

oder

**logout** bzw. **<CTRL+d>**

Wenn Sie bei neuer Mail benachrichtigt werden sollen, starten Sie zu Sitzungsbeginn

**biff**

Eine Funktion bilden Sie mit

**function**

unter bash

**typeset -f, declare -f**

Welche Variablen gesetzt sind, zeigt Ihnen

- für Ihre aktuelle Shell **set**
- auch für evtl. Unterprogramme (exportiert) **env**

Variable exportieren Sie mit

**export**

Integer-Variable bilden und rechnen mit

**typeset -i\***

Rechenoperationen führen Sie durch mit

**expr**

(bzw. über Integer-Variblen) oder mit

**let**

\* lauten nicht auf allen Systemen gleich

\*\* Kommandos unter SUSE Linux

### Übersicht der Kommandos, aus dem Buch

#### ›Linux-Systemadministration‹

Hierbei sind Befehle, die nur vom Systemverwalter/Administrator (root) ausgeführt werden dürfen, mit <sup>s</sup> gekennzeichnet.

#### Systeminformation

<code>fdisk -l</code>	Zeigt die vorhandenen Partitionen auf den Festplatten.
<code>free</code>	Zeigt den freien und belegten Speicher im System an.
<code>hostname</code>	Gibt den Rechnernamen und/oder IP-Adresse aus.
<code>hwinfo</code>	Überprüft die Hardware und gibt eine Aufstellung aus.
<code>kernelversion</code>	Gibt die Version des geladenen Kernels aus.
<code>sitar</code>	Erstellt eine druckbare Version aller wichtigen Informationen des Systems (Hardware und Software, Systemdateien etc.).
<code>uname -a</code>	Gibt Information über das System aus (Kernel-Version, Name des Betriebssystems, Rechnername u. a.).

#### Weitere Hilfe kommandos

<code>info cmd</code>	Gibt, soweit vorhanden, eine Info-Seite aus.
<code>khelppcenter</code>	Startet das grafische Hilfe-Tool unter KDE.
<code>cmd --help</code> <code>cmd [--info]</code>	Soweit vorhanden, werden die Informationen des Kommandos angezeigt (manchmal in deutscher Sprache).

#### Benutzer einrichten – verwalten

<code>chfn</code>	Ändert das Kommentarfeld der <code>/etc/passwd</code> . <b>R</b> <code>chpasswd</code>
<code>chpasswd</code>	Ändert Einstellungen der Passwortdatei.
<code>chsh</code>	Ändert die Login-Shell. <b>R</b> <code>chpasswd</code>
<code>finger</code>	Zeigt die Benutzerinformationen angemeldeter Benutzer innerhalb eines Netzes.
<code>gpasswd</code>	Ändert das Passwort einer Gruppe.
<code>groupadd</code>	Legt eine neue Gruppe an.
<code>groupdel</code>	Löscht eine Gruppe.

## Kommandoüberblick nach Funktionen

groupmod	Ändert eine vorhandene Gruppe.
groups	Zeigt alle Gruppen an, zu denen der Benutzer gehört.
grpck	Prüft die Gruppendateien auf Integrität.
id	Zeigt die Id-Nummer und Gruppenzugehörigkeit an.
last	Zeigt eine Liste der zuletzt angemeldeten Benutzer.
logname	Gibt den Login-Namen des Benutzers aus.
mkpasswd	Gibt das verschlüsselte Passwort aus.
newgrp	Meldet Benutzer unter einer anderen Gruppe an.
passwd -g	Setzt Gruppenpasswörter.
passwd -l	Sperrt den Benutzer.
pwck	Prüft die /etc/passwd auf Konsistenz.
quota	Zeigt die Quota-Werte von Benutzern und Gruppen an.
su	Startet eine neue Shell unter anderer Identität.
useradd	Legt einen neuen Benutzer an.
userdel	Löscht Benutzereinträge.
usermod	Ändert Benutzereinträge.
users	Zeigt die Benutzer aller Shells auf einem Host an.
vipw	Editiert die Passwortdatei mit exklusivem Schreibrecht.

### Dateiverwaltung

bzip2	Komprimiert Dateien (bunzip2 dekomprimiert sie).
bunzip2	Dekomprimiert mit bzip2 komprimierte Dateien.
cat	Gibt den Inhalt von Dateien aus oder führt Dateien zusammen.
chgrp	Ändert die Gruppenzugehörigkeit.
chmod	Ändert die Zugriffsrechte.
chown	Trägt einen neuen Besitzer als Owner ein.

## Kommandoüberblick nach Funktionen

<code>compress</code>	Komprimiert Dateien (aktueller und kompakter sind <code>gzip</code> und <code>bzip2</code> ).
<code>cp</code>	Kopiert eine Datei oder ein ganzes Verzeichnis.
<code>csplit</code>	Zerteilt eine Datei kontextabhängig in mehrere einzelne Dateien.
<code>dd</code>	Kopiert Daten/Dateien im Raw-Format (z. B. blockweise).
<code>df</code>	Zeigt die freie Kapazität der Dateisysteme an.
<code>du</code>	Zeigt die belegten Blöcke der Verzeichnisse an.
<code>expand</code>	Ersetzt Tabulatorzeichen zu Leerzeichen.
<code>file</code>	Versucht, den Inhalt oder die Art einer Datei zu bestimmen.
<code>find</code>	Sucht Dateien in Dateibäumen nach unterschiedlichen Suchkriterien.
<code>fsck</code>	Überprüft die Konsistenz eines Dateisystems je nach Dateisystemtyp – in unterschiedlichen Varianten für die verschiedenen Dateisystemarten wie etwa <code>fsck.ext2</code> , <code>fsck.ext3</code> , <code>reiserfsck</code> oder
<code>getfacl</code>	Zeigt die zusätzlichen Zugriffsrechte über ACL an.
<code>gunzip</code>	Dekomprimiert Dateien im GNU-zip-Format ( <code>gzip</code> komprimiert).
<code>gzip</code>	Komprimiert (und dekomprimiert) Dateien im GNU-zip-Format ( <code>gunzip</code> dekomprimiert).
<code>head</code>	Gibt jeweils die ersten <code>n</code> Zeilen einer Datei aus.
<code>less</code>	Gibt Text seitenweise auf dem Bildschirm aus.
<code>ln</code> <code>ln -s</code>	Vergibt zusätzliche Namen für Dateien (hardlink). Erstellt einen symbolischen Link.
<code>ls</code>	Zeigt den Inhalt von Verzeichnissen
<code>mkdir</code>	Legt ein neues leeres Verzeichnis an.

mkfs <sup>S</sup>	Legt ein neues Dateisystem auf einem Datenträger an. Für die unterschiedlichen Dateisysteme gibt es spezifische Versionen wie etwa <b>mkfs.ext2</b> , <b>mkfs.reiserfs</b> , <b>mkfs.jfs</b> , <b>mkfs.msdos</b> .
mknod <sup>S</sup>	Schafft einen neuen Geräteeintrag oder legt eine FIFO-Datei an.
more	Gibt Text seitenweise auf dem Bildschirm aus.
mount <sup>S</sup>	Hängt Platten/Floppies/Geräte in den Dateibaum ein.
pr	Führt eine einfache Formatierung (z.B. Unterteilung in Druckseiten mit Kopfzeilen) von Dateien für die Druckausgabe durch.
setfacl	Setzt zusätzliche Zugriffsrechte für ACLs.
split	Teilt eine Datei in mehrere kleinere Dateien.
tail	Gibt jeweils die letzten Zeilen einer Datei aus.
umask	Definiert die Maske zur Voreinstellung der Zugriffsrechte bei neu anzulegenden Dateien und Verzeichnissen.
xxd	Hexadezimale Ausgabe eines Dateiinhalts mit zusätzlicher Darstellung in ASCII.
zcat	Gibt eine mit gzip komprimierte Datei aus, ohne dass sie zuvor dekomprimiert werden muss.

### Kommandos rund ums Drucken

accept <sup>S</sup>	Bewirkt, dass der Spooler Aufträge der angegebenen Drucker oder Druckerklasse akzeptiert.
cancel	Löscht Druckaufträge.
disable <sup>S</sup>	Deaktiviert die angegebenen Drucker.
enable <sup>S</sup>	Aktiviert die angegebenen Drucker (nach einem disable).
lp oder lpr	Schickt Dateien zur Druckausgabe mittels des Print-Spoolers.
lpadmin <sup>S</sup>	Verwaltungskommando für Drucker + Druckaufträge unter CUPS.

## Kommandoüberblick nach Funktionen

<code>lpmove<sup>S</sup></code>	Verschiebt die Aufträge eines Druckers (oder einer Drucker-klasse) in die Warteschlange eines anderen Druckers / einer anderen Druckerklasse.
<code>lppasswd<sup>S</sup></code>	Setzt ein Passwort für die CUPS-Verwaltung.
<code>lpr</code>	Schickt Dateien zur Druckausgabe mittels des Print-Spoolers.
<code>lpstat</code>	Zeigt die Drucker und deren aktuelle Aufträge mit Status an.
<code>reject<sup>S</sup></code>	Bewirkt, dass der Spooler Aufträge der angegebenen Drucker oder Druckerklasse nicht mehr akzeptiert.

### Sicherung und Komprimierung

<code>cp</code>	Kopiert Dateien und Verzeichnisse.
<code>afio</code>	Erlaubt die Übertragung und das Sichern von Dateien und Dateigruppen sowie das Wiedereinlesen (als Alternative zu <code>cpio</code> ).
<code>cpio -i</code>	Liest eine zuvor unter <code>cpio -o</code> erstellte Sicherung wieder ein.
<code>cpio -o</code>	Erstellt eine Sicherung im <code>cpio</code> -Format auf einem Datenträger oder in einer Datei. Die Liste der Dateien kann z.B. über <code>find</code> und Pipe ( <code>find ...   cpio ...</code> ) übergeben werden.
<code>cpio -p</code>	Kopiert die Eingabedateien (z.B. mittels <code>find</code> ) in ein Zielverzeichnis.
<code>dd</code>	Kopiert und/oder konvertiert Dateien und Dateisysteme.
<code>dump</code>	Führt eine dateisystemspezifische Totalsicherung oder inkrementelle Sicherung für Dateisysteme vom Typ <code>ext2</code> oder <code>ext3</code> durch.
<code>restore</code>	Liest eine mit <code>dump</code> erstellte Sicherung eines Dateisystems ( <code>ext2</code> oder <code>ext3</code> ) wieder ein.
<code>file-roller</code>	Einfaches und kleines Archiv-Tool unter GNOME.

find ...   cpio -o	Sucht Dateien in Dateibäumen nach unterschiedlichen Kriterien und erstellt mit den gefundenen Dateien eine Sicherung im cpio-Format. <b>R</b> cpio
gunzip	Dekomprimiert mit gzip komprimierte Dateien.
gzip	Komprimiert (und dekomprimiert) Dateien.
karchiver	Einfaches und kleines Archiv-Tool unter KDE.
rsync	Erlaubt Verzeichnisse zu synchronisieren und über Netz zu kopieren/sichern.
scp	Kopiert Dateien netzwerkweit und verschlüsselt.
taper	Mächtiges Sicherungsprogramm zur Sicherung auf Bandmedien.
tar -c	Erstellt ein Sicherungsarchiv auf Band oder in eine Archivdatei.
tar -t	Erstellt eine Inhaltsliste von einem mit tar erstellten Archiv bzw. sucht im Archiv nach vorgegebenen Dateien.
tar -x	Liest vorgegebene Dateien aus dem Archiv wieder ein. Fehlt die Angabe der Dateien, so werden alle Dateien extrahiert.
unison	Sichert (wie rsync) Verzeichnisbäume und synchronisiert sie (unter Verwendung des rsync-Protokolls).

### Prozessverwaltung

at	Führt Kommandos zu vorgegebenen Zeiten aus.
atq	Zeigt alle anstehenden at-Aufträge an.
atrm	Erlaubt at-Aufträge zu löschen.
atrun	Lässt alle anstehenden at-Kommandos sofort laufen.
batch	Führt Aufträge zu Zeiten niedriger Systembelastung aus.
bg	Lässt Prozess im Hintergrund laufen.
crontab	Setzt zeitgesteuerte Aufträge ab.
fg	Lässt Prozesse im Vordergrund laufen.

## Kommandoüberblick nach Funktionen

<code>fuser</code>	Zeigt die Prozessnummern an, die auf die angegebenen Geräte oder Verzeichnisse zugreifen.
<code>kdesu</code>	Ein grafisches Tool, um Kommandos unter einer anderen Benutzeridentität (z.B. Super-User) auszuführen.
<code>kill Signal PID</code>	Sendet das angegebene Signal an den Prozess mit der <i>PID</i> .
<code>kill -9 PID</code>	Beendet den Prozess mit der angegebenen Prozessnummer.
<code>kill -SIGSTOP PID</code>	Hält einen Prozess an ( <code>kill -SIGCONT PID</code> setzt ihn fort).
<code>killall -c Name</code>	Beendet alle Prozesse eines Prozessnamens.
<code>ksysguard</code>	Grafisches Tool zur Systemüberwachung (ähnlich <code>ps</code> ).
<code>nohup</code>	Lässt Programme nach dem Abmelden weiterlaufen.
<code>ps</code>	Zeigt die aktuell laufenden Prozesse an.
<code>pstree</code>	Zeigt die aktuellen Prozesse in einer Baumstruktur an.
<code>renice</code>	Verändert den nice-Wert eines laufenden Prozesses.
<code>su</code>	Erlaubt temporär unter der Identität eines anderen Benutzers zu arbeiten.
<code>sudo</code>	Erlaubt über die Definitionsdatei <code>/etc/sudoers</code> auch normalen Benutzern, bestimmte Programme mit dem Super-User-Recht auszuführen.
<code>top</code>	Zeigt interaktiv die ›Top‹-Prozesse an.
<code>vcron</code>	Grafisches Tool, um zeitgesteuerte Aufträge abzusetzen.
<code>visudo</code>	Editorbefehl für <code>root</code> , um die Datei <code>sudoers</code> zu bearbeiten.
<code>which</code>	Gibt den absoluten Pfadnamen eines Kommandos aus.
<code>xhost</code>	Setzt die Berechtigung, um grafische Ausgaben von anderen Rechnern/Benutzern zuzulassen.

Netzwerke	
ethereal (tethereal*)	Zeigt sämtlichen Netzwerkverkehr an (eine Art Sniffer).
findsmb	Zeigt alle Rechner an, die auf smb-Anfragen antworten (im Samba-Paket).
finger	Zeigt alle Benutzer an, auch jene, die über Netz angemeldet sind.
ftp	Transferiert Dateien über das Netz.
host	Zeigt die jeweilige Auflösung der Namens in IP-Adresse oder umgekehrt.
hostname	Gibt den in der Datei /etc/host eingetragenen Rechnernamen aus.
ifconfig	Gibt Kontrollwerte zu den installierten Netzwerkkarten aus (analog unter Windows: ipconfig).
ip	Ist mächtiger als ifconfig. Es kann sowohl Netzwerkgeräte, Routen und Tunnels anzeigen als auch diese verändern.
kinternet	Grafisches Tool zur Internetverbindung über Modem (s. a. wvdial)
netstat	Liefert Statusinformationen über das Netzwerk.
nmblookup	Zeigt NetBIOS-Namen an.
nmap	Kommando, um den Netztransfer zu kontrollieren.
ping	Prüft IP-Verbindungen auf unterster Ebene.
rlogin	Remote-Login – ähnlich dem Kommando <b>R</b> telnet.
route	Zeigt die IP-Routen-Tabelle an.
scp	Kopiert (wie cp) Dateien verschlüsselt über ein Netz.
sftp	Transferiert Dateien verschlüsselt über das Netz.
smbclient	Greift auf Windows-Freigaben zu.
smbpasswd	Setzt Passwörter für Samba-Benutzer.
smbstatus	Zeigt die aktuellen auf Samba-basierenden Verbindungen an.

## Kommandoüberblick nach Funktionen

<code>smbtree</code>	Zeigt alle Freigaben über Samba im Netz an.
<code>ssh</code>	Öffnet eine Shell auf einem entfernten Rechner mit verschlüsselter Übertragung.
<code>ssh-keygen</code>	Generiert Schlüsselpaare für eine gesicherte Übertragung.
<code>tcpdump</code>	Zeigt allen Verkehr auf tcp an.
<code>telnet</code>	Anmelden an einem entfernten Rechner.
<code>testparm</code>	Überprüft die Samba-Konfigurationsdatei <code>smb.conf</code> .
<code>traceroute</code>	Kontrolliert die Netzwerkverbindungen über Router.
<code>wvdial</code>	Wählt auf der Basis von PPP über das Modem die eingetragene Telefonnummer des Internetproviders in der Konfigurationsdatei.
<code>xhost</code>	Setzt die Berechtigung, um grafische Ausgaben von anderen Rechnern/Benutzern zuzulassen.
<code>xnmap</code>	Grafisches Tool, um den Netztransfer zu kontrollieren.
<code>yppasswd</code>	Vergibt Passwörter für Benutzer unter NIS.

## Konfigurieren und Software nachinstallieren

<code>insmod</code>	Lädt ein Kernel-Modul.
<code>insserv</code>	Fügt ein installiertes System-Skript in die betreffenden Init-Verzeichnisse der Runlevel.
<code>lsmod</code>	Zeigt den Status der Module im Linux-Kernel.
<code>modprobe</code>	Zeigt die eingebundenen Module des Linux-Kernels an, fügt neue hinzu oder löscht sie.
<code>rpm</code>	RPM installiert oder überprüft Softwarepakete, die als RPM-Pakete aufbereitet sind.
<code>yast (yast2)</code>	Zentrales Administrationswerkzeug und Konfigurationstool unter SUSE Linux

---

## Kommandos alphabetisch

Kommandoeingabe	Funktion
<b>alias [-x]kürzel="Befehl "</b>  <i>Beispiel:</i> <b>alias ll="ls -l"</b>	Zusatzname nicht <i>sh</i> <b>Setzt Kürzel für Befehle</b> → C-Shell, Unterschiede → Wichtige Dateien für den Benutzer, <i>.bashrc</i> , <i>.kshrc</i>
<b>at [ Zeit [Datum]Kommando]</b>  <i>Beispiel:</i> <b>at 18:00 write hans &lt; Ende</b> <i>Weitere Optionen:</i> <b>at [-l][-r Kommando]</b>	<b>at</b> – zu bestimmter Zeit <b>Führt Kommandos zu bestimmten Zeiten aus</b> Um 18:00 wird "hans" die Nachricht, die in der Datei Ende steht, geschickt <b>-l list</b> listet vorhandene at-Jobs <b>-r remove</b> löscht den at-Job für das betreffende Kommando
<b>banner Zeichenkette Zeichenkette</b>	<b>Großdarstellung von Zeichenketten am Bildschirm</b> <i>(nicht unter Linux)</i>
<b>bash</b>	Ruft die Bash auf Die Datei \$HOME/.bashrc wird hierbei gelesen
<b>bg %Jobnummer</b>	<b>background</b> <span style="float:right"><i>nicht sh</i></span> <b>Der Job läuft als Hintergrundprozess weiter</b> → <b>fg</b> → <b>stop</b>
<b>biff [y,n]</b>	<b>Benachrichtigt den Benutzer sobald neue mail eingegangen ist (y)</b>  Dieses Kommando ist oft in die Datei \$HOME/.profile mit aufgenommen (nicht unter bash)
<b>boot</b>	<b>Hochfahren des Rechners</b> Nur vom Systemverwalter einzugeben (u.U. wichtige Entscheidungen bei Prüfrouinen) <i>(nicht unter bash)</i>
<b>break</b>	<i>brechen /abbrechen</i> <b>Beendet vorzeitig eine Schleife</b>
<b>cal [[Monat][Jahr]]</b>  <i>Beispiel:</i> <b>cal 2000   pg</b>	<b>calendar -Kalender-Ausgabe</b> Anzeige des aktuellen Kalenders oder vom angegebene Zeit

Kommandoeingabe	Funktion
<p><b>cancel Druck-Auftragsnr.</b>  <i>Beispiel:</i>  <b>cancel laser-124</b></p> <p>Unter DOS: <b>type /c</b></p>	<p><i>annullieren, abbrechen</i></p> <p><b>Löscht mit lp gestartete Druckaufträge</b></p> <p>Auftragsnr. für Drucker-Queue laser</p> <p>→ <b>lp, lpstat</b></p>
<p><b>case ...</b>  <b>esac</b></p>	<p><i>Case-Verarbeitung (Auswahl)</i></p> <p>→ <b>Ablaufsteuerung der Shell</b></p>
<p><b>cat Dateiname(n)</b></p> <p><i>Beispiele</i>  <b>cat datei1 datei2 &gt; datei-neu</b>  <b>cat &gt; neu</b>  Dies ist eine neue Datei  &lt;Ctrl+d&gt;  Unter DOS: <b>type</b></p>	<p><i>concatenate – zusammenfügen</i></p> <p><b>Zeigt den Inhalt von Dateien</b></p> <p>Mehrere Dateien können in <b>eine</b> Datei umgeleitet werden</p> <p>Mit cat kann so eine neue Datei angelegt werden (Standardeingabe ist das Terminal; Abschluß mit der Tastenkombination Ctrl+d)</p>
<p><b>cd [Directory]</b></p> <p><b>cd</b></p> <p>Beispiele:  <b>cd ..</b></p> <p><b>cd ~/hans</b>  <b>cd /usr/kurs/hans</b></p> <p>Unter DOS: <b>cd</b></p>	<p><i>change directory</i></p> <p><b>Wechsel in das HOME- oder angegebene Directory</b></p> <p>Ohne Angabe kehrt man immer ins Home-Directory zurück</p> <p>Wechselt in ein Directory nach oben</p> <p>Wechselt in das Home-Directory von hans  mit ~&lt; <b>nicht sh</b></p>
<p><b>chgrp [-R]Gruppenname \</b>  <i>Dateinamen/ Directories</i>  <i>Beispiel:</i>  <b>chgrp -R kurs \</b>  <b>/usr/kurs/ben01</b></p>	<p><i>change group</i></p> <p><b>Ändert die Gruppenzugehörigkeit</b></p> <p><b>-R recursive</b>  die Änderung erfolgt für alle Dateien und Unter-Directories des angegebenen Directories</p>
<p><b>chmod [-R] Art Dateina-</b>  <b>men/ Directories</b></p>	<p><i>change modus</i></p> <p><b>Ändert die Zugriffsrechte</b></p> <p>Art: symbolisch oder über Oktalzahl</p> <p><b>-R recursive</b>  Änderung erfolgt für alle Dateien/ Unter-Directories</p> <p><i>Fortsetzung nächste Seite</i></p>

Kommandoingabe	Funktion																					
<p>für wen wie was</p> <p>▼ ▼ ▼</p> <p><b>chmod ugo <math>\pm</math> rwx Dateinamen/ Directories</b></p>	<p><b>Ändert die Zugriffsrechte mit symbolischer Angabe</b></p> <p><b>user</b> der Dateibesitzer  <b>group</b> die gleiche Gruppe  <b>other</b> alle anderen  <b>+</b> hinzufügen  <b>-</b> wegnehmen  <b>=</b> absolut setzen  <b>read</b> Leseerlaubnis  <b>write</b> Schreiberlaubnis  <b>execute</b> ausführbar</p>																					
<p><b>chmod Oktalzahl Dateinamen/Directories</b></p> <p><b>Errechnung der Oktalzahl</b></p> <table> <tr> <td>r</td> <td>read</td> <td>4</td> </tr> <tr> <td>w</td> <td>write</td> <td>2</td> </tr> <tr> <td>x</td> <td>executable</td> <td>1</td> </tr> </table> <p>Beispiel:  <b>chmod 750 ben01</b></p>	r	read	4	w	write	2	x	executable	1	<p><b>Ändert die Zugriffsrechte mit Oktalzahl</b></p> <table> <thead> <tr> <th>Besitzer</th> <th>Gruppe</th> <th>Andere</th> </tr> </thead> <tbody> <tr> <td>r w x</td> <td>r - x</td> <td>- - -</td> </tr> <tr> <td>4+ 2+ 1</td> <td>4+ 0+ 1</td> <td>0+ 0+ 0</td> </tr> <tr> <td>=7</td> <td>=5</td> <td>=0</td> </tr> </tbody> </table> <p>Das Directory erhält die Zugriffsrechte:  rwxr-x---</p>	Besitzer	Gruppe	Andere	r w x	r - x	- - -	4+ 2+ 1	4+ 0+ 1	0+ 0+ 0	=7	=5	=0
r	read	4																				
w	write	2																				
x	executable	1																				
Besitzer	Gruppe	Andere																				
r w x	r - x	- - -																				
4+ 2+ 1	4+ 0+ 1	0+ 0+ 0																				
=7	=5	=0																				
<p><b>chown [-R]Benutzer \ Dateinamen/Directories</b></p>	<p><b>change owner</b>  <b>Ändert den Besitzer</b>  <b>-R recursive</b>  Die Änderung erfolgt für alle Dateien/Unter-Directories</p>																					
<p><b>clear</b></p> <p>Unter DOS: <b>cls</b></p>	<p><i>klären, reinigen</i>  <b>Löscht den aktuellen Bildschirminhalt</b>  Beginnt neu an der obersten Zeile</p>																					
<p><b>cmp</b>  Beispiel:  <b>cmp text1 text2</b></p> <p>Unter DOS: <b>comp, fc</b></p>	<p><i>compare, vergleichen</i>  <b>Vergleicht Dateinhalte</b>  Bei Ungleichheit werden die unterschiedlichen Zeilen angezeigt</p>																					
<p><b>compress Dateiname(n)</b></p> <p>Beispiel:  <b>compress tarsicherung</b></p>	<p><i>verdichten</i>  <b>Verdichten/Komprimieren von Dateien</b>  Oft in Verbindung mit tar/ftp Der Dateiname wird mit <b>.Z</b> erweitert.  Inhalt der Datei anzeigen → <b>zcat</b>; Datei wieder dekomprimieren → <b>uncompress</b></p>																					

Kommandoeingabe	Funktion
<b>continue</b>	<i>fortfahren</i> <b>Überspringt den Rest der Schleife um mit dem nächsten Schleifenwert fortzufahren</b>
<b>cp</b> [-i]Datei1 Dateineu <b>cp</b> [-i]Datei1 Datei2 ... Directory <b>cp</b> [-ir ]Directory Directory	<i>copy</i> <b>Kopiert eine Datei oder kopiert eine oder mehrere Datei(en) in ein anderes Directory</b> <b>-i interactive</b> Falls eine Datei mit gleichem Namen schon existiert, wird nachgefragt, ob sie überschrieben werden darf <b>-r recursive</b> Der gesamte Dateibaum wird kopiert
Unter DOS: <b>copy, xcopy</b>	
<i>Ausgabe</i> Dateiliste   <b>cpio</b> \-p[dmuv]Ziel-Directory	<i>copy input output</i> <b>-p pass – weiterreichen</b> <b>Kopiert Dateien von Platte zu Platte</b> Über eine Dateiliste, z.B. mit <b>find . -print</b> werden die zu kopierenden Dateien übergeben <b>-d directory</b> Unter-Directories werden angelegt, falls sie noch nicht vorhanden sind <b>-m modification date</b> Die kopierte Datei erhält das Datum der Originaldatei <b>-u unconditional</b> Die kopierte Datei überschreibt evtl. schon vorhandene Dateien. Dateien werden sonst nur dann überschrieben, wenn deren Modifikationsdatum älter ist <b>-v verbose – geschwätzig</b> Alle ausgeführten Kopien werden angezeigt Alle Dateien des aktuellen Directories werden mit gleichem Namen in das Directory /tmp/Sicherung/hans kopiert
<i>Beispiel:</i> <b>cd /usr/kurs/hans</b> <b>find . -print   cpio -pvmd \ /tmp/Sicherung/hans</b>	
Unter DOS: <b>xcopy</b>	

Kommandoeingabe	Funktion
<p><i>Beispiel:Ausgabe</i>  Dateiliste   <b>cpio-o[vB]&gt;</b> \  Geräte- oder Archivdatei</p> <p><i>Beispiel:</i>  <b>find . -print   cpio -ovB &gt; /  dev/mt0</b>  Unter DOS: <b>xcopy</b></p>	<p><b>copy input output</b> – Ausgabe  <b>Kopiert Dateien auf einen Datenträger oder in eine Archivdatei</b>  <b>Achtung:</b> Ausgabe-Umleitungszeichen &gt; angegeben!</p> <p><b>-o output</b>  kopiert die als Dateiliste übergebenen Dateien und Unter-Directories auf das angegebene Gerät oder in die Archivdatei</p> <p><b>-v verbose</b> – <i>geschwätzig</i>  Alle ausgeführten Kopien werden angezeigt</p> <p><b>-B block</b> Blockungsfaktor für Magnetband/Streamer  Alle Dateien des aktuellen Directories werden auf Magnetband geschrieben</p>
<p><b>cpio -i[dmuv][Dateien]\</b>  &lt; Gerät- oder Archivdatei</p> <p>Unter DOS: <b>xcopy</b></p>	<p><b>copy input output</b> – Einlesen  <b>Liest Dateien aus einem mit cpio erstellten Datenarchiv (Datei oder Datenträger) zurück</b>  <b>Achtung!</b> Eingabe-Umleitungszeichen &lt; angeben!</p> <p><b>-i input</b>  Einlesen/Zurückschreiben der Sicherung</p> <p><b>-d directory</b>  Unter-Directories werden angelegt, falls sie noch nicht vorhanden sind</p> <p><b>-m modification date</b>  Die kopierte Datei erhält das Datum der Originaldatei</p> <p><b>-u unconditional</b>  Die kopierte Datei überschreibt evtl. schon vorhandene Dateien, sonst werden Dateien nur dann überschrieben, wenn das Modifikationsdatum älter ist</p> <p><b>-v verbose</b> – <i>geschwätzig</i>  Alle ausgeführten Kopien werden angezeigt</p>

Kommandoeingabe	Funktion
<p><b>csh</b></p> <p>Unter DOS: <b>command.com</b></p>	<p>Ruft die <b>C-Shell</b> auf</p> <p>Die Datei <code>.cshrc</code> im Home-Directory des Benutzers wird dabei gelesen</p>
<p><b>cut</b> [-dZeichen -fFeldnr -cZeichenposition]</p> <p>Beispiel: <b>cut -d: -f1,5-6 \ /etc/passwd</b></p>	<p><i>schneiden</i></p> <p><b>Schneidet/separiert Felder aus Dateien oder Zeichenketten</b></p> <p><b>-d delimiter</b> Trennungszeichen (Default ist das Tabulatorzeichen)</p> <p><b>-f field</b> - Feldnummer</p> <p><b>-c character</b> Zeichenposition</p> <p>Gibt über Bildschirm die Felder 1, 5 und 6 aus: Name, Kommentar und Home-Directory</p>
<p><b>date</b> [+“Formatangaben“]</p> <p>Beispiel: <b>date +%d.%m.%y</b></p> <p>Unter DOS: <b>date/time</b></p>	<p><b>Zeigt das Datum an</b> (nur vom Systemverwalter kann es auch geändert werden)</p> <p><b>+</b> Kennzeichen für eigene Formatierung</p> <p>hierbei steht</p> <p>%y für Jahr YY %m für Monat MM %d für Tag TT %A Wochentag</p> <p>weitere Platzhalter:</p> <p>%H – hour 00 – 23 %M – minute 00 – 59 %S – second 00 – 59 %T – time HH:MM:SS %w – day of week (Sunday =0) %h – abgek. Monat (Jan – Dec)</p>
<p><b>declare</b> Name=Wert</p>	<p><b>Zuweisung einer Variablen unter bash</b> → <b>typeset</b> → Besonderheiten der Bash</p>
<p><b>dd</b> if=Dateiname of=Geräte-name \ Option=Wert</p> <p>Beispiel: <b>tar -cvf - .   rsh Rechner \ dd of=/dev/rmt/0 bs=64K</b></p>	<p><i>device to device</i></p> <p><b>Kopiert Dateien, Dateibereiche oder gesamte Platten 1:1</b></p> <p>Beispiel: Ausgabe von <i>tar</i> über Netz auf einen Streamer an einen entfernten Rechner</p> <p><b>bs block size</b> Wert (in diesem Beispiel) 64 kB</p>

Kommandoeingabe	Funktion
<b>df</b> [-k][-h]	<i>disk free</i> <b>Zeigt die verfügbare Plattenkapazität in 512-Byte-Blöcken für alle montierten Plattenbereiche an</b> -k <i>kilo</i> – Zeigt die Kapazität in 1 kB-Blöcken an -h <i>human</i> – Gibt lesbare Werte, wie kB oder MB aus
<b>diff</b> <i>Datei1 Datei2</i>	<b>Vergleicht Dateien</b>
<b>dosdir</b>  Unter DOS: <b>dir</b>	<i>DOS Directory</i> <b>Zeigt den Inhalt eines DOS-Directories auf einer PC-Floppy (Linux: → mdir)</b>
<b>dosformat</b> Unter DOS: <b>format</b>	<i>DOS Floppy formatieren</i> <b>Formatiert eine DOS-Floppy (Linux: → mformat)</b>
<b>dosread</b> <i>DOS-Datei \ UNIX-Datei</i> Unter DOS: <b>copy</b>	<i>DOS Datei lesen</i> <b>Kopiert eine DOS-Datei nach UNIX (Linux: → mcopy)</b>
<b>doswrite</b> <i>UNIX-Datei DOS-Datei</i> Unter DOS: <b>copy</b>	<i>DOS Datei schreiben</i> <b>Kopiert eine UNIX-Datei auf eine DOS-Floppy (Linux: → mcopy)</b>
<b>du</b> [-s][-h][ <i>Directory</i> ]	<i>disk used</i> <b>Zeigt den verbrauchten Plattenplatz</b> -s <i>sum</i> Zeigt nur jeweils die Summe der Directories in 512-Byte-Blöcken an -h <i>human</i> – Gibt lesbare Werte, wie kB oder MB aus
<b>echo</b> [ <i>Text Text</i> ]  <i>Beispiele:</i> <b>echo "Soll die Datei gelöscht werden?"</b> <b>echo \$PATH</b> <b>Steuerzeichen im Text:</b>  Unter DOS: <b>echo</b>	<b>Gibt Zeichenketten auf den Bildschirm aus</b> -n <i>no newline</i> Die Ausgabe erfolgt in derselben Zeile. <i>Gibt in einer Shell-Prozedur die Nachricht über Bildschirm aus</i> Zeigt den Wert einer Variablen <b>\a</b> <i>alert</i> Klingelzeichen <b>\c</b> Cursor in gleicher Zeile <b>\n</b> neue Zeile <b>\t</b> Tabulator

Kommandoeingabe	Funktion
<b>ed</b> <i>Dateiname</i>  Unter DOS: <b>edlin</b>	<b>editor</b> <b>Zeilenorientierter Editor</b> Auch für Shell-Skripts geeignet → <b>Editor</b>
<b>emacs</b> <i>Dateiname</i>	<b>editor</b> <b>Bildschirmorientierter Editor</b> Ähnlich wie der vi, etwas benutzerfreundlicher, doch nicht auf allen Rechnern verfügbar → <b>Editor</b>
<b>env</b>	<i>environment</i> <b>Zeigt die gesetzten Variablen an, die auch für Unterprozesse gelten (exportiert sind)</b> → <b>printenv</b> (bash)
<b>exit</b> [ <i>Status</i> ]  Unter DOS: <b>exit</b>	<i>Ausgang</i> <b>Bricht eine Shell-Prozedur ab bzw. beendet die aktuelle Shell</b> → <b>kill</b> , Signale (Exit-Status)
<b>export</b> <i>Variable(n)</i>	<i>exportieren</i> <b>Die Variablen gelten dann auch für Unterprogramme</b> → <b>Variable</b>
<b>expr</b> <i>Wert1 Symbol Wert2</i>  + - \* / %	<i>expression</i> <b>Rechenoperationen:</b> <b>addieren</b> <b>subtrahieren</b> <b>multiplizieren</b> <b>dividieren</b> <b>modulo (Restwert)</b>
<b>false</b>	<i>falsch, unwahr</i> <b>Der Exit-Status dieses Kommandos ist immer unwahr (ungleich 0)</b>
<b>fc</b> [-l n][-e -]  statt <b>fc -l</b> gibt es ein Alias <b>history</b>	<i>fix command</i> <b>nicht sh</b> <b>Wiederholt bereits eingegebene Kommandos/Befehle oder zeigt sie an (History-Mechanismus)</b> -l <i>list</i> – bzw. <i>history</i> zeigt die letzten 10 bzw. n Kommandos <i>Fortsetzung nächste Seite</i>

Kommandoeingabe	Funktion
<p><b>fc</b> <i>Fortsetzung</i>  <b>statt fc -e -</b>  <b>gibt es ein Alias r</b></p> <p><b>r</b>  <b>r -n</b></p> <p><b>r n</b>  <b>r name</b></p>	<p><b>-e edit</b> – bzw. <b>r repeat</b> (unter <b>csh</b> nur !)</p> <p>wiederholt den letzten Befehl</p> <p>wiederholt den <i>n</i>-tletzten Befehl</p> <p>wiederholt den <i>n</i>-ten Befehl</p> <p>wiederholt den letzten Befehl, in dem "name" enthalten ist</p>
<p><b>fg</b> <i>%Jobnummer</i></p>	<p><i>foreground</i> <b>nicht sh</b></p> <p><b>Der Job läuft als Vordergrundprozess weiter</b></p> <p>Gilt nur für Hintergrund- oder gestoppte Prozesse</p> <p>→ <b>bg</b> → <b>stop</b></p>
<p><b>file</b> <i>Dateiname(n)</i></p>	<p><i>Datei</i></p> <p><b>Versucht den Inhalt oder die Art einer Datei zu bestimmen</b></p> <p>zum Beispiel</p> <p><i>ascii text</i>  <i>c program text</i>  <i>commands text</i>  <i>directory</i></p>
<p><b>find</b> <i>Start Suchkriterien \</i>  <i>[Ausgabeart]</i></p> <p>Beispiel:  <b>find . -print</b></p> <p><b>Suchkriterien:</b>  <b>-name</b> <i>Dateiname</i></p>	<p><i>finden</i></p> <p><b>Sucht (findet) Dateien in Dateibäumen nach unterschiedlichen Suchkriterien</b></p> <p>Es muss ein <b>Start-Directory</b> angegeben werden (aktuelles Directory mit <b>.</b> angeben)</p> <p>Die Ausgabe muss bei den meisten Systemen noch mit <b>-print</b> angegeben werden</p> <p>Es wird eine Liste aller Dateien <b>rekursiv</b> durch alle Unter-Directories mit <b>relativem Pfadnamen</b> ausgegeben</p> <p>Nach bestimmten <b>Namen</b>, Metazeichen ( <i>*?[ ]</i> ) mit Anführungszeichen eingeben</p> <p><i>Fortsetzung nächste Seite</i></p>

Kommandoeingabe	Funktion
<b>find</b> <i>Fortsetzung</i> <b>-mtime</b> <i>n</i> <b>-mtime</b> <i>-n</i> <b>-mtime</b> <i>+n</i> <b>-newer</b> <i>Datei</i> <b>-inum</b> <i>inode-Nr</i> <b>-mount</b> <b>-user</b> <i>Benutzername</i> <b>Ausgabe:</b> <b>-print</b> <b>-exec</b> <i>Kommando</i> {} \; <b>-ls</b> <b>Logische Kombination</b> <b>-a</b> <b>-o</b> <i>Beispiele:</i> <b>find . \(-name "a*" -a -user hans\) -exec head {} \;</b> <b>find /usr \(-name core -o -name "*.tmp"\) -exec rm {} \;</b>	<i>Modifikationsdatum</i> <b>vor genau <i>n</i> Tagen</b> <b>innerhalb von <i>n</i> Tagen</b> <b>vor <i>n</i> Tagen und früher</b> <b>neuer</b> als die angegebene Datei mit der angegebenen <b>inode-Nummer</b> nur auf dem aktuellen ( <b>montierten</b> ) Plattenbereich Dateien des angegebenen <b>Benutzers</b> Dateien werden mit Pfadnamen angezeigt das <b>Kommando</b> wird mit jeder gefundenen Datei <b>durchgeführt</b> die gefundenen Dateien werden <b>mit allen Attributen</b> (wie ls -l) <b>angezeigt</b> Suchkriterien können logisch verbunden werden <b>und-Verknüpfung</b> <b>oder-Verknüpfung</b> Sucht nach Dateien, die mit a beginnen und "hans" gehören. Von den gefundenen Dateien werden die ersten 10 Zeilen angezeigt Sucht ab /usr alle core-Dateien und/oder Dateien, die mit .tmp enden. Die gefundenen Dateien werden gelöscht
<b>finger</b> [ <i>Benutzer</i> ]	<b>Zeigt Benutzerinformation</b> an mit Namen@Rechner, Terminal und Anmeldezeit
<b>for ...</b> <b>do</b> <b>done</b>	<b>Leitet eine Schleife ein</b> → Shell-Ablaufsteuerung

Kommandoeingabe	Funktion
<b>ftp</b> Rechner connected to ... login: <i>Benutzername</i> Password required for : <i>Paßwort</i> Auswahl der meistbenötigten Kommandos innerhalb von ftp:	<i>file transfer protokol</i> <b>Kopiert Dateien von/auf entfernte Rechner</b> Rechnername des entfernten Rechners sowie ein dort gültiger Benutzername und Paßwort müssen angegeben werden
<b>cd</b>	<i>change directory</i> – wechselt in das Directory auf dem entfernten Rechner
<b>pwd</b>	<i>print working directory</i> – zeigt das aktuelle Directory auf dem entfernten Rechner
<b>ls [dir]</b>	<i>list</i> – zeigt den Inhalt des aktuellen Directories auf dem entfernten Rechner
<b>get</b> <i>Dateiname</i>	<b>get</b> – holen, bekommen kopiert in das aktuelle Directory des lokalen Rechners die Datei des entfernten Rechners
<b>mget</b> <i>Dateiname(n)</i>	<b>multiple get</b> – <i>mehrfach holen, bekommen</i> kopiert alle angegebenen Dateien (z. B. über Dateinamenexpansion) in das aktuelle Directory des lokalen Rechners
<b>put</b> <i>Dateiname</i>	<b>put</b> – <i>abgeben</i> kopiert die angegebene Datei von Ihrem lokalen Rechner in das aktuelle Directory des entfernten Rechners
<b>mput</b> <i>Dateiname(n)</i>	<b>multiple put</b> Kopiert mehrere Dateien (evtl. über Dateinamenexpansion) vom lokalen Rechner in das aktuelle Directory des entfernten Rechners
<b>binary</b>	<i>binary</i> – <i>binär</i> Schaltet in den Binärmodus um. Mit binary werden z. B. Programme, tar- oder cpio-Archiv-Dateien übertragen, die im Binärformat gespeichert sind
<b>ascii</b>	Umstellung wieder auf ASCII-Mode
<i>Fortsetzung nächste Seite</i>	

Kommandoeingabe	Funktion
<p><b>ftp:</b> Fortsetzung</p> <p><b>delete</b> <i>Dateiname(n)</i></p> <p><b>mkdir</b></p> <p><b>lcd</b> <i>Directory</i></p> <p><b>bye</b> oder <b>quit</b></p>	<p><i>löschen</i></p> <p>Soweit die Zugriffsrechte es zulassen, können Dateien auf dem entfernten Rechner gelöscht werden</p> <p><b>make Directory</b></p> <p>legt auf dem entfernten Rechner ein Directory an</p> <p><b>local chance directory</b></p> <p>wechselt auf dem lokalen Rechner das Directory</p> <p>beendet die Verbindung zum anderen Rechner</p>
<p><b>function</b> <i>name</i> {  <i>Kommando</i>  <i>Kommando ...</i>  <i>Kommando</i>  }</p> <p><i>Beispiel:</i></p> <p><b>function</b> <i>wo</i> {  find . -name \$1  }  <i>wo brief1</i></p>	<p><i>Funktion</i></p> <p><b>Bildet eine Funktion, die ähnlich eines Shell-internen Kommandos genutzt werden kann</b></p> <p>Die Funktion kann ein oder mehrere Kommandos enthalten, wobei auch Positionsparameter (\$1, \$2 usw.) verwendet werden können</p> <p>Beim Aufruf von <i>wo</i> wird der Name der Datei mitgegeben, nach der ab aktuellem Directory gesucht wird → Konstrukte</p>
<p><b>functions</b></p> <p>(unter bash: <b>declare -f</b>)</p>	<p><b>Gibt eine Liste der vorhandenen Funktionen aus</b></p> <p>(nicht bash, csh, tcsh)</p>
<p><b>grep</b> [<i>hilnvw</i>]<i>Muster Dateiname(n)</i></p>	<p><b>get regular expression</b></p> <p><b>Durchsucht Dateiinhalte nach bestimmten Zeichenvorgaben/Suchmustern</b></p> <p>Im Muster können <b>Metazeichen</b> (<i>regular expression</i>) wie im <i>ed/vi</i> verwendet werden</p> <p>→ <b>Editor</b></p> <p><b>-h header</b> der Dateiname wird nicht mit ausgegeben</p> <p><b>-i ignore</b> behandelt Groß- und Kleinbuchstaben gleich</p> <p><b>-l line</b> nur die Dateinamen werden angezeigt</p>
<p><i>Fortsetzung nächste Seite</i></p>	

Kommandoingabe	Funktion
<b>grep</b> <i>Fortsetzung</i>  <i>Beispiel:</i> <b>grep -v "^\."</b> <i>trofftext</i> Unter DOS: <b>find</b>  <b>gunzip, gzip</b>	<b>-n number</b> gibt zusätzlich die Zeilennummer mit aus <b>-v invert</b> gibt alle Zeilen aus, die nicht dem Muster entsprechen <b>-w word</b> Suchmuster muss ein einzelnes Wort sein Es werden alle Zeilen aus der Datei trofftext angezeigt, die <b>nicht</b> mit einem "." beginnen <b>Komprimierung von Dateien</b> → <b>compress</b> → <b>zip, unzip</b>
<b>halt</b>	<b>Anhalten – Ausschalten</b> Auf einigen Systemen nötig, um den Rechner auszuschalten Systemverwalter-Kommando
<b>head</b> [-n] <i>Dateiname(n)</i>	<b>Zeigt die ersten 10 Zeilen einer Datei an</b> <b>-n number</b> zeigt <i>n</i> Zeilen der Datei an
<b>history</b> [-n]	<i>Vergangenes</i> <b>nicht sh</b> <b>Zeigt die letzten 10 bzw. n Kommandos an (alias zu → fc -l)</b>
<b>hostname</b>	<i>Wirt – Rechner, von dem das Netz ausgeht</i> <b>Zeigt den Rechnernamen an</b>
<b>if ...</b> <b>then</b> else <b>fi</b>	<b>Leitet eine if-Bedingung ein</b> → <b>Shell Ablaufsteuerung</b>
<b>integer</b> <i>name[=Rechenoperation]</i> (bei bash: <b>typeset -i</b> )  <i>Beispiel:</i> <b>integer zahl=10</b> <b>mögliche Operatoren</b>	<i>ganze Zahl</i> <b>nur ksh</b> <b>Bildet eine Integer-Variable</b> (alias zu typeset -i) → <b>typeset</b> Bei der Zuweisung und innerhalb der Rechenoperation dürfen keine Leerzeichen enthalten sein <b>Auswirkung</b>
+ - * / %	Addition Subtraktion Multiplikation Division Modulo → <b>expr</b>

Kommandoeingabe	Funktion
<b>jobs</b>	<p><i>Jobcontrol</i> <b>nicht sh</b></p> <p><b>Zeigt vorhandene Jobs an</b></p> <p>Jobs sind gestoppte Vordergrundprozesse (<b>Ctrl+z</b>) oder Hintergrundprozesse</p> <p>→ <b>fg</b> → <b>bg</b></p>
<p><b>kill -9 [PID][%Jobnr]</b></p> <p>einige Signale:</p> <p>1 HUP</p> <p>2 INT</p> <p>3 QUIT</p> <p>9 KILL</p> <p>17 STOP</p> <p>19 TSTP</p>	<p><i>kill (töten)</i></p> <p><b>Bricht einen Prozess sicher ab</b></p> <p>Beendet Terminalverbindung</p> <p>Entspricht Ctrl+c Canceln</p> <p>Abbruch mit coredump</p> <p>Absoluter Prozessabbruch</p> <p>Stoppt einen Prozess (stop)</p> <p>Stoppt einen Vordergrundprozess (Ctrl+z)</p>
<p><b>ksh</b></p> <p>Unter DOS: <b>command.com</b></p>	<p><i>Korn-Shell</i></p> <p><b>Startet eine Korn-Shell</b></p> <p>Hierbei wird die Datei <b>\$HOME/.kshrc</b> gelesen, wenn die Variable <b>ENV</b> entsprechend gesetzt ist</p> <p>→ <b>alias</b> → <b>Wichtige Dateien</b></p> <p>→ <b>Variable</b></p>
<b>less</b>	<p><i>weniger (less is more)</i></p> <p><b>Zeigt den Inhalt von Dateien seitenweise an</b></p> <p>Verbesserte Variante von more unter Linux → <b>more</b></p>
<p><b>ln [-s]Originaldatei Linkname</b></p> <p><b>ln -s Original-Directory Link-Directory</b></p>	<p><i>link</i></p> <p><b>Vergibt Dateien zusätzliche Namen bzw. verweist auf eine andere Datei oder ein anderes Directory</b></p> <p>Der lokale Link (ohne -s) kann nur auf Dateien innerhalb der gleichen Plattenpartition erfolgen</p> <p><b>-s symbolic link</b></p> <p>Das Link-Directory soll mit vollem Pfadnamen eingetragen werden.</p> <p>Der symbolische Link wird als eigener Dateityp mit <b>l</b> (klein L – link) gekennzeichnet</p>

Kommandoeingabe	Funktion
<b>logout</b> <i>ebenso</i> <b>exit</b> <i>oder</i> <b>Ctrl+d</b>	<i>login – logout,</i> <i>anmelden – abmelden</i> <b>Beendet eine Shell</b>
<b>lp [-dDruckername]Dateiname</b> <i>me</i>	<i>line printer</i> <b>Erstellt einen Druckauftrag für den lp-Spooler</b> <b>-d destination</b> Mit -d“Druckerqueue“ kann ein Zieldrucker angegeben werden, soweit mehrere Drucker eingerichtet wurden → <b>lpstat</b> → <b>cancel</b>
Unter DOS: <b>print</b>	
<b>lpq</b>	<i>line printer queue</i> <b>Zeigt die Queue der mit lpr gestarteten Druckaufträge an</b> (analog lpstat für lp-Aufträge) → <b>lpr</b>
<b>lpr</b>	<i>line printer</i> <b>Erstellt einen Druckauftrag für den lpq-Spooler</b> (wird manchmal statt lp verwendet) → <b>lpq</b> → <b>lprm</b>
Unter DOS: <b>print</b>	
<b>lprm Auftrag-Nr.</b>	<i>line printer remove</i> <b>Löscht mit lpr gestartete Druckaufträge</b>
<b>lpstat</b>	<i>line printer status</i> <b>Zeigt alle mit lp gestarteten Druckaufträge und deren Status an</b> (analog lpq für lpr-Aufträge) → <b>lp</b> → <b>cancel</b>
<b>ls [-abdFilRst]</b>	<i>list</i> <b>Zeigt den Inhalt von Directories und Attribute von Dateien</b> <b>-a all</b> Auch die mit Punkt beginnenden Dateien (.profile ...) werden angezeigt <b>-b binary</b> Zeigt auch nicht darstellbare Zeichen am Bildschirm an <b>-d directory</b> Zeigt nur das Directory (nicht seinen Inhalt) an <i>Fortsetzung nächste Seite</i>

Kommandoeingabe	Funktion
<p><b>ls [-abdFilRst]</b>  <i>Fortsetzung</i></p> <p>Unter DOS: <b>dir, tree</b></p>	<p><b><i>l-F Format short</i></b>            Directories sind mit "/" gekennzeichnet, ausführbare Kommandos/Programme mit "**", symbolische Links mit "@"</p> <p><b><i>-i inode</i></b> Die Adresse (inode-Nummer) wird angezeigt</p> <p><b><i>-l long format</i></b>            Anzeige mit Attributen</p> <p><b><i>-R recursive</i></b> Der Dateibaum mit sämtlichen Unter-Directories wird angezeigt</p> <p><b><i>-s size</i></b> Es werden zusätzlich die benötigten Blöcke à 512 Bytes angezeigt</p> <p><b><i>-t time</i></b> Die Liste wird chronologisch sortiert ausgegeben</p>
<p><b>mail [benutzer[@rechner] benutzer... ]</b></p> <p><i>Subject:</i>            Eingabe der Nachricht direkt über Tastatur - beenden mit <b>&lt;Ctrl+d&gt;</b></p> <p><b><i>~r dateiname</i></b>  <i>bzw</i></p> <p><b><i>~v</i></b></p> <p><i>Cc:</i></p> <p><b>Befehle innerhalb von mail:</b></p> <p><b>d</b></p> <p><b>h</b></p>	<p><i>Post versenden</i></p> <p><b>Verschickt an einen oder mehrere Benutzer Post (mail)</b>  <i>(elektronische Post – e-mail)</i></p> <p>Betreff eintragen, der in den Headerzeilen erscheint            Statt direkter Texteingabe kann mit</p> <p><b><i>~r read</i></b> – Text aus einer vorhandenen Datei eingelesen</p> <p><b><i>~v vi</i></b> – der vi-Editor aufgerufen werden</p> <p><i>copy</i> – an: z. B. den eigenen Namen als Sicherungskopie            Ohne Angabe von einer Adresse wird die eingegangene Post angezeigt oder "no mail"</p> <p><b>?</b> als Eingabe zeigt die möglichen Befehle innerhalb von mail</p> <p><b>Auswahl:</b></p> <p><b><i>delete</i></b> löscht die aktuelle Nachricht</p> <p><b><i>header</i></b> zeigt eine Liste aller eingegangener Mail mit Subject, Datum und Größe</p> <p><i>Fortsetzung nächste Seite</i></p>

Kommandoeingabe	Funktion
<b>mail</b> <i>Fortsetzung</i>	
<b>r</b>	<b>reply</b> erstellt eine Antwort an den Absender
<b>m</b> <i>benutzer</i>	<b>mail</b> leitet die Nachricht an den angegebenen Benutzer weiter
<b>s</b> <i>dateiname</i>	<b>save</b> sichert die Nachricht mit Kopfzeilen in die angegebene Datei
<b>w</b> <i>dateiname</i>	<b>write</b> schreibt die Nachricht ohne Kopfzeilen in die angegebene Datei Sollen eingegangene Mails sofort gemeldet werden, kann in der .profile-Datei → <b>biff y</b> eingetragen werden → <b>Wichtige Dateien, .mailrc</b>
<b>man</b> [ <i>Kapitel-Nr</i> ] <i>Kommando</i>	<b>manual</b> <b>Gibt eine Beschreibung des angegebenen Kommandos in Verbindung mit → more aus</b>
Unter DOS: <b>befehl /?</b>	
<b>man -k</b> <i>Thema</i>	<b>manual</b> <b>Gibt eine Liste von Kommandos aus, die mit dem Thema zu tun haben</b>
<b>mcopy</b> <b>mdir</b>	<i>DOS-Befehle unter Linux</i> → <b>dosread, doswrite, dosdir</b>
<b>mesg</b> [ <i>y</i> ][ <i>n</i> ]	<b>messages</b> <b>Verhindert oder erlaubt Mitteilungen von anderen Benutzern mit write</b> (Schreibrecht für Terminal) <b>n</b> keine Schreiberlaubnis ( <b>no</b> ) <b>y</b> wieder freigeben ( <b>yes</b> ) Ohne Angabe wird der jeweilige Berechtigungszustand angezeigt ( <i>is yes – is no</i> )
<b>mformat</b>	<b>Formatiert eine DOS-Floppy</b> (Unter Linux) → <b>dosformat</b>

Kommandoeingabe	Funktion
<p><b>mkdir [-p]Directoryname(n)</b></p> <p><i>Beispiel:</i>  <b>cd /usr/kurs</b>  <b>mkdir -p ben01/Uebung</b>            Unter DOS: <b>mkdir, md</b>  <b>more datei(en)</b></p> <p><b>h</b></p> <p><b>/Muster</b>  <b>?Muster</b>  <b>Leertaste</b>  <b>q</b>            Unter DOS: <b>more</b></p>	<p><i>make directory</i></p> <p><b>Legt Directories neu an</b></p> <p><b>-p pass</b> noch nicht vorhandene Unter-Directories werden mit angelegt</p> <p>Soweit noch nicht vorhanden, werden die Directories angelegt:            /usr/kurs/ben01            /usr/kurs/ben01/Uebung</p> <p><i>mehr</i></p> <p><b>Zeigt den Inhalt von Dateien seitenweise an</b></p> <p><b>h help</b> zeigt die möglichen Kommandos, u.a.:</p> <p>sucht vorwärts nach Muster            sucht rückwärts nach Muster            zeigt die nächste Seite an</p> <p><i>quit</i> beendet more            → <b>pg</b> → <b>less</b></p>
<p><b>mount /dev/Gerät \</b>  <i>Directory mit absolutem Pfadnamen</i></p>	<p><i>montieren</i></p> <p><b>Montiert Platten/Floppies</b></p> <p>Formatierte und mit einem UNIX-Dateisystem versehene Platten/Floppies werden in den Gesamtdateibaum unter dem angegebenen Directory eingehängt            → <b>umount</b></p>
<p><b>mt [-rew][-fsf]n</b></p>	<p><i>magnet tape</i>            Kontrollfunktionen</p> <p><b>-rew rewind</b>            Spult um n Datensätze (files) zurück</p> <p><b>-fsf files forward</b>            Spult um n Datensätze vor</p>
<p><b>mv -i datei1 dateineu</b>  <b>mv -i datei1 datei2 ... Directory</b></p> <p>Unter DOS: <b>ren (rename)</b></p>	<p><i>move, bewegen</i></p> <p><b>Ändert einen Dateinamen oder verschiebt Dateien in ein anderes Directory</b></p> <p><b>-i interactive</b></p> <p>Eine bereits bestehende Datei wird nur dann überschrieben, wenn dies mit <b>y</b> bestätigt wird</p>

Kommandoeingabe	Funktion
<b>nice</b> <i>-n</i> <i>Prioritätswert Kommando</i>	<p><b>Startet ein Kommando mit Prioritätsänderung</b></p> <p>Der "normale" Benutzer kann Priorität nur verringern, Wert von 0 – 20 (je höher, desto geringere Priorität)</p> <p>→ <b>renice</b></p>
<b>nohup</b> <i>Kommando</i>	<p><b>no hang up</b></p> <p><b>Der Prozess wird nicht abgebrochen, wenn der Vaterprozess beendet wird</b></p>
<b>passwd</b> [ <i>Benutzername</i> ]	<p><b>Ändert das Paßwort</b></p> <p>Das Paßwort wird nicht angezeigt und sollte mindestens 6 Zeichen enthalten</p>
<b>pg</b> <i>Dateiname</i>	<p><b>page</b></p> <p><b>Zeigt den Inhalt von Dateien oder einer Ausgabe (Pipe) seitenweise an</b></p> <p>Etwas komfortabler als more;</p> <p>→ <b>more</b></p>
Unter DOS: <b>more</b>	
<b>ping</b> <i>Rechnername</i>	<p><i>Ping-Pong</i></p> <p><b>Kontrolliert, ob der angegebene Rechner erreichbar ist</b></p>
<b>pr</b> [ <i>-ln -on -wn -n Spaltenanzahl</i> ] <i>Dateiname(n)</i>	<p><b>print</b></p> <p><b>Bereitet Dateien für den Druck auf</b></p> <p><b>-l length</b> Anzahl der Zeilen (Seitenlänge)</p> <p><b>-o offset</b> Zeicheneinrückung vom linken Rand (eine Einheit entspricht einem m)</p> <p><b>-w width</b> Anzahl der Zeichen pro Zeile (Breite)</p> <p><b>-n numbering</b> Die Zeilennummern werden mit ausgedruckt</p> <p><b>1-9 (Spaltenanzahl)</b> Der Text wird in die angegebene Anzahl Spalten aufgeteilt</p>
<b>printenv</b>	<p><i>print environment</i></p> <p><b>Unter Linux Anzeige der Variablen</b> → <b>env</b></p>

Kommandoeingabe	Funktion
<p><b>ps</b> [-efl][-u <i>Benutzer</i>]  <b>ps</b> -[axl]</p>	<p><i>process status</i>  <b>Anzeige der aktuellen Prozesse</b>  <b>-e every</b> Anzeige aller Prozesse  <b>-f full</b> volles Format  <b>-l long</b> mit allen Attributen  <b>-u user</b> Prozesse des angegebenen Benutzers  Bei einigen Systemen werden unterschiedliche Optionen verlangt so z. B.  <b>-a all</b> alle Prozesse eines Terminals  <b>-x</b> alle Systemprozesse</p>
<p><b>pwd</b></p>	<p><i>print working directory</i>  <b>Zeigt das aktuelle Directory mit absolutem Pfadnamen an</b>  → <b>Variable \$PWD (nicht sh)</b></p>
<p><b>rcp</b> <i>Datei(en)[@Rechner]\</i>  <i>Datei(/Directory)[@Rechner]</i></p>	<p><i>remote copy</i>  <b>Kopiert Dateien von/auf entfernte Rechner, die mit TCP/IP verbunden sind</b>  Etwa gleiche Syntax wie → <b>cp</b></p>
<p><b>read</b> <i>var1 [var2 ... varn]</i></p>	<p><i>lesen</i>  <b>Liest von der Standardeingabe und weist die gelesenen Zeichen der/den Variable(n) als Wert zu</b>  Werden mehrere Variable angegeben, gilt das Leerzeichen als Trennzeichen, sonst werden alle Zeichen bis Zeilenende als Wert zugewiesen</p>
<p><b>renice -n</b> <i>Prioritätswert \PID</i></p>	<p>Verändert den nice-Wert eines laufenden Prozesses  <b>Prozess-ID-Nummer</b>  → <b>nice</b></p>
<p><b>rlogin</b> [-l <i>Login-Name</i>]<i>Rechner</i></p>	<p><i>remote login</i>  <b>Anmelden an einem entfernten Rechner, der mit TCP/IP verbunden ist</b>  Wird die Verbindung hergestellt, muss ein gültiger Benutzername und ein Paßwort eingegeben werden  → <b>telnet</b></p>

Kommandoeingabe	Funktion
<p><b>rm</b> [-fir][-]Dateiname(n)</p> <p>Unter DOS: <b>del</b></p>	<p><i>remove</i> – löschen</p> <p><b>Löscht Dateien (unwiederbringlich!)</b></p> <p><b>-f forced</b> – verstärkt Auch bei Dateien, die schreibgeschützt sind, wird ohne Nachfrage gelöscht</p> <p><b>-i interactive</b> die Löschung erst mit "y" bestätigt werden</p> <p><b>-r recursive</b> Vorsicht! Löscht radikal alle Dateien und evtl. Unter-Directories!</p> <p>- ohne weitere Angabe Die nachfolgenden evtl. mit "-" beginnenden Namen sind keine Optionen (sondern z. B. versehentlich mit "-" beginnend angelegte Dateien)</p>
<p><b>rmdir</b> Directory</p> <p>Unter DOS: <b>rmdir (rd)</b></p>	<p><i>remove directory</i></p> <p><b>Löscht Directories, die keine Dateien mehr enthalten</b></p>
<p><b>rsh</b> Rechner</p>	<p><i>remote shell</i></p> <p><b>Startet eine Shell auf einem entfernten Rechner</b></p> <p>Auf dem entfernten Rechner muss der gleiche Benutzername eingetragen sein. Das Paßwort wird abgefragt</p>
<p><b>Rsh</b></p>	<p><i>Restricted Shell</i></p> <p><b>Eine Shell, die dem Benutzer nur eingeschränkte Rechte einräumt</b></p>
<p><b>scp</b> [[Benutzer1@]Host1:]\ Datei [[Benutzer2@]Host2:]\ [Datei2]</p> <p>Beispiel: <b>scp -r carsten@toshili:Texte</b> <b>Textekopie</b></p>	<p>(secure copy) Kopiert netzwerkweit Dateien verschlüsselt. Bei Kopien übers Netz kann sowohl die Quelle als auch das Ziel am entfernten Rechner sein.</p>

Kommandoeingabe	Funktion
<p><b>sed 'script' datei &gt; dateineu</b>  <b>sed -f script-datei datei &gt; dateineu</b></p> <p><i>Beispiel:</i>  <b>sed 's/neu/alt/g' text1 \&gt;text2</b></p> <p><b>sed 's/Herr/&amp;n/ ' text1 \&gt;text2</b></p>	<p><b>stream editor</b></p> <p><b>Ist ein batch-orientierter Editor.</b></p> <p>Wird keine Umleitung in eine Datei angegeben, erfolgt die Ausgabe über Bildschirm</p> <p>Unter script werden Kommandozeilen (etwa gleiche Syntax wie ed und ex) in Hochkomma eingegeben</p> <p>In Such- und Ersetzungsfunktionen können reguläre Ausdrücke verwendet werden (Metazeichen)</p> <p>Das Script kann auch in einer eigenen Datei abgelegt sein, die dann mit <b>-f file</b> gelesen wird</p> <p>Von der Datei text1 werden alle Zeichenketten ›alt‹ in ›neu‹ ersetzt und in text2 gespeichert. text1 bleibt unverändert.</p> <p>Alle Zeichenfolgen aus der Datei text1 mit ›Herr‹ werden in ›Herr‹ abgeändert und in text2 gespeichert</p> <p>Diese Substitutionen können auch unter dem vi/vim über ›;‹ genutzt werden</p> <p>→ <b>Editor, Metazeichen</b></p>

Kommandoingabe	Funktion
set	<i>setzen</i> <b>Zeigt die gesetzten Variablen der aktuellen Shell</b>
set [-vxn]	<b>Setzt Shell-Optionen</b> <b>-n no execution</b> die Kommandos werden nur gelesen und nicht ausgeführt <b>-x execute</b> zeigt alle ausgeführten Befehle an <b>-v verbose</b> zeigt alle Schritte einer Prozedur an
set -	Hebt gesetzte Optionen (-vxn) wieder auf
set -o	<b>Setzt zusätzliche Optionen (ksh)</b> hier nur einige: der angegebene Editor wird zum Editieren der Befehlszeile verwendet bestehende Dateien werden bei einer Ausgabeumleitung nicht überschrieben Hebt die zusätzliche Option wieder auf
<b>emacs oder vi</b>	
<b>noclobber</b>	
set +o [emacs, ... ]	→ <b>Bourne- und Korn-Shell, Optionen</b>
Unter DOS: set	
sftp	<i>secure file transfer protocol</i> → <b>ftp</b>
sh	<b>Startet eine Bourne-Shell</b>
shutdown	<b>Herrunterfahren des Rechners synchronisieren</b>
sync	<b>anhalten – ausschalten</b>
halt	Bei einigen Systemen werden <i>sync</i> und <i>halt</i> noch zusätzlich eingegeben, bevor der Rechner ausgeschaltet wird. Aufruf nur durch den Systemverwalter möglich
sleep <i>Sekunden</i>	<i>schlafen</i>
	<b>Leitet einen Wartezustand ein (Anzahl Sekunden)</b>
<i>Beispiel:</i>	Wird meist in Shell-Prozeduren verwendet.
sleep 180	Der Prozess wartet 3 Minuten

Kommandoeingabe	Funktion
<p><b>sort [-bfnr]-tZeichen \ [-kPosition, Position[nr]]Datei</b></p> <p><i>Beispiel:</i> Ausschnitt /etc/passwd: Spaltenzuordnung getrennt durch »:«</p> <pre>0      1      2 3 4      ...* ben01:xcziui:101:20:Meier:... ben02:xcber:102:20:Huber:... ben03:nmx fz:103:20:Beck:..</pre> <p><b>sort -t: +4 -4 /etc/passwd</b></p> <p><b>sort -t: -k4.1,4.4 \ /etc/passwd</b></p> <pre>ben03:nmx fz:103:20:Beck:.. ben02:xcber:102:20:Huber:... ben01:xcziui:101:20:Meier:...</pre> <p>* Vorsicht: Bei einigen Rechnern mit 1 beginnend</p>	<p><i>sortieren</i></p> <p>Sortiert Dateiinhalte oder Zeichenketten nach verschiedenen Kriterien</p> <ul style="list-style-type: none"> <li>-b <i>blank</i> führende Leerzeichen ignorieren</li> <li>-f <i>fold</i> Groß- und Kleinbuchstaben werden gleich behandelt</li> <li>-n <i>number</i> Numerische Werte am Anfang werden numerisch sortiert</li> <li>-r <i>reverse</i> es wird in umgekehrter Richtung sortiert</li> <li>-t <i>Trennzeichen</i> Wenn nicht das Leerzeichen als Trennung zwischen Spalten dient, wird hier das Trennungszeichen vorgegeben</li> </ul> <p><i>Kriterien für den Sort</i></p> <p>Von Position Zahl1.Zahl2 , bis Position Zahl1.Zahl2 Zahl1=SpaltenNr Zahl2=ZeichenNr innerhalb der Spalte</p> <p>Die Ausgabe erfolgt standardmäßig auf den Bildschirm. Mit der Zusatzoption <b>-o Datei</b> kann die Ausgabe umgeleitet werden (auch in die Eingabedatei)</p> <p>Neuere Eingabe: über + kann die Position über - die Zeichenanzahl angegeben werden</p> <p>Die Datei /etc/passwd wird nach den ersten 4 Buchstaben im Kommentar (Name) in aufsteigender Reihenfolge sortiert am Bildschirm angezeigt</p>
<p><b>ssh</b></p>	<p><i>secure shell</i> → <b>rlogin, telnet</b></p>
<p><b>stop %Jobnummer</b> unter Bash:</p> <p><b>kill -SIGSTOP %Jobnummer</b> oder <i>PID</i></p>	<p><i>anhalten</i> <span style="float: right;"><b>nicht sh</b></span></p> <p><b>Stoppen eines Jobs/Prozesses</b> (bei Vordergrundprozessen: <b>Ctrl+z</b>)</p>

Kommandoeingabe	Funktion
<b>tail [-fn,+n]Datei</b>	<p><b>Zeigt die letzten Zeilen einer Datei an</b></p> <p><b>-n number</b> Anzahl Zeilen (default 10)</p> <p><b>+n</b> Gesamte Datei nach <i>n</i> Zeilen vom Beginn anzeigen</p> <p><b>-f following :</b> Bei laufender Verarbeitung werden kontinuierlich hinzukommende Zeilen angezeigt</p>
<b>talk Benutzername</b>	<p><i>sprechen</i></p> <p><b>Mit einem anderen Benutzer Mitteilungen über Bildschirm austauschen, Verbindung eröffnen bzw. sich dazuschalten</b></p> <p>Komfortabler als → <b>write</b></p>
<p><b>tar -c[bhvzf Gerät-/Dateinamen]Start-Directory</b></p> <p>Bei einigen Systemen (z. B. Linux) gibt es weitere Optionen (<b>-z, -Z</b>), um die Archivdatei gleich mit <i>gzip</i> (<b>z</b>) oder <i>compress</i> (<b>Z</b>) zu komprimieren bzw. dekomprimieren (<b>tar -x</b>)</p> <p><i>Beispiel:</i></p> <p><b>tar -cvf /tmp/Sich.tar .</b></p>	<p><i>tape archive – create</i></p> <p><b>Sichert Dateien auf Magnetband/Streamer oder in eine Archivdatei</b></p> <p><b>-B Block</b> Blockungsfaktor für Magnetband</p> <p><b>-h hardlink</b> verfolgt alle Links und kopiert die dort aufgefundenen Dateien</p> <p><b>-v verbose</b> zeigt alle kopierten Dateien an</p> <p><b>-z gzip</b> das Archiv wird mit <i>gzip</i> komprimiert</p> <p><b>-f file</b> Das direkt nachfolgende Wort ist die Bezeichnung für das Gerät oder der Name für die Archivdatei</p> <p>Kopiert alle Dateien des aktuellen Directories ( <b>.</b> ) in die Archivdatei <i>/tmp/Sich.tar</i></p> <p>Die Datei <i>/tmp/Sich.tar</i> wird automatisch neu angelegt bzw. eine bereits vorhandene überschrieben</p>
<b>tar -t[vf Gerät-/Dateinamen]</b>	<p><b>-t table</b> Es wird nur ein Inhaltsverzeichnis des Archivs ausgegeben</p> <p><b>-v verbose</b> zeigt zusätzlich alle Attribute der Dateien an</p>

Kommandoeingabe	Funktion
<p><b>tar -x[mvzf</b> <i>Gerät-/Dateinamen</i> <b>][Datei(en)]</b></p> <p>weitere Optionen (<b>-z, -Z</b>) siehe <i>tar -c</i></p> <p><i>Beispiel:</i> <code>tar -xvmf /tmp/Sich.tar \</code> <code>./text1</code></p>	<p><i>tape archiver</i></p> <p><b>-x extract</b> extrahieren</p> <p><b>Liest Dateien von einem mit tar erstellten Datenträger oder einer Archivdatei zurück</b></p> <p><b>-m modify</b> die zurückgelesenen Dateien erhalten das Originaldatum (nicht das aktuelle Datum)</p> <p><b>-v verbose</b> zeigt alle kopierten Dateien an</p> <p><b>-z gzip</b> das zuvor mit zip komprimierte Archiv wird mit gunzip dekomprimiert</p> <p><b>-f file</b> Das nachfolgende Wort ist die Bezeichnung für das Gerät oder der Name für die Archivdatei</p> <p>Kopiert aus der Archivdatei die Datei <code>./text1</code> zurück ins aktuelle Directory</p>
<p><b>tee</b></p> <p><i>Beispiel:</i> <code>ls -l   tee inhalt   wc -l</code></p>	<p><i>T-Stück einer Pipeline</i></p> <p><b>Leitet bei Pipe-Mechanismus die Ausgabe zusätzlich in eine Datei um</b></p> <p>Die Liste aller Dateien wird in die Datei <code>inhalt</code> geschrieben. Am Bildschirm wird nur die Anzahl der Dateien angezeigt</p>
<p><b>telnet</b> <i>Rechner</i></p>	<p><b>telecommunication netware</b></p> <p><b>Anmelden an einem entfernten Rechner</b></p> <p>Wird die Verbindung hergestellt, muss ein gültiger Benutzername und ein Paßwort eingegeben werden</p> <p>Etwa gleich wie → <b>rlogin</b></p>
<p><b>test -fdrwxs</b> <i>datei</i></p>	<p><i>testen, prüfen</i></p> <p><b>Prüft Dateien auf Typ, Inhalt oder Zugriffsrechte</b></p> <p><i>Fortsetzung nächste Seite</i></p>

Kommandoingabe	Funktion
<b>test -fdrxs</b> <i>datei</i> <i>Fortsetzung</i> [ <b>-f</b> <i>datei</i> ]	test wird meist in Verbindung mit if .. verwendet Modernere Schreibweise statt test nur die eckige Klammer [ ], hierbei muss auf Leerzeichen vor und nach den Klammern geachtet werden! Bei den Optionen wird geprüft, ob die angegebene Datei: <b>-f file</b> eine normale Datei ist <b>-d directory</b> ein Directory ist <b>-r read</b> Leseerlaubnis hat <b>-w write</b> Schreiberlaubnis hat <b>-x execute</b> ausführbar ist <b>-s size</b> nicht leer ist <b>Negation</b> wird mit einem ! gefolgt von einem <b>Leerzeichen</b> abgefragt
<b>Negation:</b> <b>test ! -f</b> <i>datei</i>	
<b>test -zn</b> <i>Zeichenkette</i>	<b>Prüft Zeichenketten auf leer/nicht leer</b> Abfrage auf : <b>-z zero</b> ›ist leer?‹ <b>-n not zero</b> ›ist nicht leer?‹
<b>test String-a = String-b</b>  <b>Negation:</b> <b>test String-a != String-b</b>	<b>Prüft Zeichenketten auf Gleichheit</b> Zwischen den Argumenten und = müssen Leerzeichen stehen Negation, d.h. test ist erfolgreich, wenn die Zeichenketten nicht gleich sind. <b>!=</b> ohne Leerzeichen!
<b>test n1 -.. n2</b> <i>moderne Schreibweise:</i> [ <b>n1 -.. n2</b> ]  <b>test n1 -eq n2 [ n1 -eq n2 ]</b> <b>test n1 -ne n2 [ n1 -ne n2 ]</b> <b>test n1 -lt n2 [ n1 -lt n2 ]</b> <b>test n1 -le n2 [ n1 -le n2 ]</b> <b>test n1 -gt n2 [ n1 -gt n2 ]</b> <b>test n1 -ge n2 [ n1 -ge n2 ]</b>	<b>algebraischer Vergleich:</b>  <b>equal</b> gleich <b>not equal</b> nicht gleich <b>less than</b> kleiner als <b>less equal</b> kleiner gleich <b>greater than</b> größer als <b>greater equal</b> größer gleich

Kommandoeingabe	Funktion
<b>test - Fortsetzung Beispiele:</b> Auszug aus einem Skript: <b>read datei</b> <b>if test -f \$datei -a -w \$datei</b> <b>then ...</b> <i>read antw</i> <b>if test \$antw = ja -o \</b> <b>\$antw = Ja</b> <b>then ...</b>	<b>Kombination von test-Kommandos:</b>  <b>-a and</b> logische <b>Und-Verknüpfung</b>  <b>-o or</b> logische <b>Oder-Verknüpfung</b>
<b>touch Datei(en)</b>  <i>Beispiel:</i> <b>touch neu1 neu2</b>	<i>berühren</i> <b>Aktualisiert das Datum einer Datei bzw. legt sie neu an</b> Legt die Dateien neu1 und neu2 an
<b>trap "Kommandos" Signale</b> <i>Beispiel:</i> <b>trap "rm hilfs.dat; exit " 2</b>  <b>trap "" 1 2 3</b>	<b>Behandelt Signale</b> → <b>kill</b> (Signale) Wird z. B. das Signal 2 (Ctrl+c) geschickt, wird, bevor der Prozess abgebrochen wird, die Datei hilfs.dat gelöscht. Ohne Angabe, d. h. nur mit Anführungszeichen " ", werden die Signale ignoriert. In diesem Fall kann z. B. nicht mit Ctrl+c (Signal 2) abgebrochen werden → <b>kill</b>
<b>true</b> <i>Beispiel:</i> <b>while true</b> <b>do ...</b> <b>done</b>	<i>wahr</i> <b>Gibt immer den Exit-Status 0 aus, ist also immer wahr</b> Hiermit können z. B. Endlosschleifen gestartet werden
<b>tty</b>	<i>terminal type</i> <b>Zeigt den aktuellen Terminalnamen</b>
<b>typeset [-iulx]</b>	<b>Setzt Variable nurksh/bash</b> <b>-i integer</b> alias: <b>integer</b> für ganze Zahlen <b>-u upper case</b> ( <i>nicht bash</i> ) nur für Großbuchstaben <b>-l lower case</b> ( <i>nicht bash</i> ) nur für Kleinbuchstaben <b>-x export</b> Die Variable wird gleich exportiert → <b>Variable</b>

Kommandoeingabe	Funktion
<b>umount /dev/Geräte-namen</b>	<i>wieder abmontieren</i> <b>Hängt montierte Dateisysteme wieder ab</b>
<b>unalias</b> <i>Kürzel</i> bzw. nur nachfolgender Alias wird aufgehoben: <i>✓Kürzel</i>	<b>Löst die Bedeutung eines Alias wieder auf</b> <i>nur ksh/bash/csh</i> Alias wird nur temporär aufgelöst
<b>uname</b>  Unter DOS: <b>ver</b>	<i>unix name</i> des Rechners <b>Gibt rechner spezifische Informationen aus</b>
<b>uncompress</b> <i>Datei[.Z]</i>	<i>dekomprimieren</i> <b>Eine zuvor mit compress verdichtete Datei wird wieder in den Normalzustand gebracht</b>
<b>unset</b> <i>Variable</i>	<b>Hebt den zugewiesenen Wert einer Variable wieder auf</b>
<b>until ...</b> <b>do</b> <b>done</b>	<i>solange nicht ...</i> <b>Leitet eine Schleife ein</b> → <b>Shell-Ablaufsteuerung</b>
<b>unzip</b> <i>Datei.zip</i>	<b>Extrahiert und dekomprimiert von mit zip-erstellten Dateien</b> → <b>zip</b>
<b>vi [-rR]Datei(en)</b>	<i>visual editor</i> <b>Bildschirmorientierter Editor</b> <b>-r recovery</b> Es wird ein Protokoll mitgeschrieben, das bei einem Absturz alle Eingaben nachvollziehen kann <b>-R Read only</b> die angegebene Datei darf mit dem vi nur gelesen nicht verändert werden → <b>sed</b> (Umschaltung über : in den ex-Modus) → <b>Editoren Metazeichen</b> → <b>Häufig benutzte Kommandos im vi</b>
<b>wc [-wcl]</b>	<i>word count, Wörter zählen</i> <b>Zählt Zeilen, Wörter und Buchstaben</b> <b>-w word</b> Anzahl der Wörter <b>-l line</b> Anzahl der Zeilen <b>-c character</b> Anzahl der Zeichen

Kommandoeingabe	Funktion
<b>whatis</b> <i>Kommando</i>	<i>was ist das Kommando</i> <b>Gibt eine Kurzinformation des angegebenen Kommandos aus</b>
<b>whereis</b> <i>Kommando</i>	<i>wo ist das Kommando</i> <b>Zeigt den absoluten Pfadnamen eines Kommandos an</b>
<b>while</b> .. <b>do</b> <b>done</b>	<i>solange ...</i> <b>Leitet eine Schleife ein</b> → <b>Shell-Ablaufsteuerung</b>
<b>who</b> <b>who am i</b>	<i>Wer arbeitet am System?</i> <b>Zeigt die angemeldeten Benutzer und Terminals</b> Es wird der Benutzernamen, die Terminalbezeichnung und die Anmeldezeit angezeigt → <b>finger</b>
<b>write</b> <i>Benutzername@Rechner</i>	<i>schreiben</i> <b>Einem anderen Benutzer Mitteilungen auf den Bildschirm schicken</b>
<b>zcat</b> <i>Dateiname.Z</i>	<i>cat von .Z-Dateien</i> <b>Zeigt den Dateiinhalt von mit compress komprimierten Dateien an</b>
<b>zip</b> <i>Datei.zip Datei1 [Dateien]</i>	<b>Erstellt eine komprimierte Archivdatei (<i>Datei.zip</i>) mit den nachfolgend angegebenen Dateien</b> → <b>unzip</b>

Kommandos nur für den **Systemverwalter**: An der Systemkonsole und als Benutzer **root** einzugeben (ausführliche Kurzreferenz im Buch Linux-Systemadministration → Seite 7)

## Editoren

## Metazeichen – reguläre Ausdrücke (ed, grep, sed, vi)

Meta-zeichen	Such-beispiel	Bedeutung
.	/d/	<b>Beliebiges einzelnes Zeichen</b> sucht nach allen Wörtern, die ein beliebiges Zeichen gefolgt von einem d, enthalten
*	/m*/	<b>Mögliche Wiederholung des vorangestellten Zeichens</b> sucht nach einem "m" oder hintereinander mehrfach auftretenden ›m‹
.*	/*.*/	steht für mehrere beliebige Zeichen sucht nach Zeichenfolgen, in denen ein ›m‹ vorkommt
^	/^ */	<b>Nachfolgender Suchbegriff steht am Anfang einer Zeile</b> sucht ein oder mehrere Leerzeichen am Anfang einer Zeile
\$	/EUR\$/	<b>Nachfolgender Suchbegriff steht am Ende einer Zeile</b> sucht nach einer Zeile, die mit EUR endet
[ ]	/[a-z]/	<b>Suchen eines in der Klammer angegebenen Zeichens</b> sucht nach einem beliebigen Kleinbuchstaben
	/[137]/	sucht nach Ziffern 1, 3 oder 7
\	/\*/	<b>Fluchtsymbol – Aufheben einer evtl. Sonderfunktion des nachfolgenden Zeichens</b> sucht nach einem ›*‹
&	s/Muster/&../	<b>Einsetzen des gefundenen Musters als neuen Begriff</b>
	s/Herr/&n/	ersetzt ›Herr‹ durch ›Herrn‹





## Häufig benutzte Kommandos im ed (ex, sed)

Verwendung	Befehl/ Beispiel	Bedeutung
Aufruf	<b>ed</b> <i>Datei</i> <b>ed sprueche</b>	Von der angegebenen Datei wird eine Kopie in den Arbeitsspeicher geladen Es wird nur die Anzahl der Bytes angezeigt
Positionieren bzw. Angabe des Bereichs	<i>n</i> <b>3</b>	Der Arbeitszeiger wird auf die Zeile <i>n</i> , im Beispiel Zeile 3, positioniert
	<i>n1,n2</i> <i>Kommando</i> <b>2,5p</b>	Das Kommando betrifft den Bereich <i>n1</i> bis <i>n2</i> Die Zeilen 2 bis 5 sollen angezeigt ( <i>print</i> ) werden
	<i>n,\$d</i>	Von Zeile <i>n</i> bis Ende der Datei löschen ( <i>delete</i> )
	<i>4,\$n</i>	Zeile 4 bis Ende der Datei wird mit der lfd. Nummer angezeigt
	<b>.n</b>	Die aktuelle Zeile wird mit der Zeilennummer angezeigt
Umschalten in den Eingabemodus	<b>a</b>	Text wird nach der aktuellen Zeile als neue Zeilen angehängt ( <i>append</i> )
	<i>n a</i> <b>3a</b>	Nach Zeile <i>n</i> , im Beispiel Zeile 3, Text als neue Zeile anhängen
	<b>i</b> <i>ni</i> <b>1i</b>	( <i>insert</i> ) Text als neue Zeilen vor der aktuellen Zeile einfügen, bzw. vor Zeile <i>n</i> Beispiel vor Zeile 1
	<b>c</b> <b>5c</b>	( <i>change</i> ) Die aktuelle Zeile wird ersetzt bzw. als Beispiel wird die Zeile 5 ersetzt. Wird mehr als eine Zeile Text eingegeben, werden die weiteren Zeilen eingefügt.
Beenden des Eingabemodus	<b>.</b>	Der Punkt muss am Anfang der Zeile stehen. Er beendet den Eingabemodus.




## Häufig benutzte Kommandos im ed – Fortsetzung

Verwendung	Befehl/ Beispiel	Bedeutung
Suchen	<i>/Muster/</i>	Nach dem Begriff/Muster wird in der Datei <b>vorwärts</b> gesucht
	?Muster?	<b>rückwärts</b> gesucht
	//	Zuletzt gesuchtes Muster vorwärts suchen bzw.
	??	rückwärts suchen
Ersetzen	<i>s/alt/neu/</i>	<i>substitute</i> – ersetzen In der aktuellen Zeile wird das erste Auftreten von <i>alt</i> durch <i>neu</i> ersetzt
	<b>1,\$s/alt/neu/g</b>	In der gesamten Datei (1,\$) wird jedes Vorkommen von <i>alt</i> durch <i>neu</i> ersetzt ( <i>global</i> – d.h., auch mehrmals in einer Zeile)
Löschen von Zeilen	<b>d</b>	( <i>delete</i> ) löscht die aktuelle Zeile
	<b>3d</b>	Zeile 3 wird gelöscht
	<b>4,6d</b>	Zeilen 4 bis 6 werden gelöscht
Rückgängig machen	<b>u</b>	( <i>undo</i> ) Der zuletzt eingegebene Befehl wird <i>ungeschehen</i> gemacht
Benden von ed	<b>w [Datei]</b>	( <i>write</i> ) Zurückschreiben des Arbeitspuffers in die beim Aufruf bzw. in mit <b>w</b> angegebene Datei
	<b>q</b>	( <i>quit</i> ) Beenden Falls die Datei noch nicht mit <b>w</b> gesichert wurde, wird eine Warnung ausgegeben
	<b>q!</b>	Beenden von ed ohne Sicherung

## Häufig benutzte Kommandos im vi

Verwendung	Befehl/ Beispiel	Bedeutung
Aufruf	vi <i>Datei</i> vi sprueche	Von der angegebenen Datei wird eine Kopie in den Arbeitsspeicher geladen
Beenden	:w [ <i>Datei</i> ]  :q  :q!  ZZ	Schreibt in die beim Aufruf oder bei :w angegebene Datei zurück  Beendet den vi (falls noch nicht gesichert wurde (:w), wird eine Warnung ausgegeben)  Beendet den vi <b>ohne Warnung</b> , falls vorher noch nicht gesichert (:w) wurde  Sichert (schreibt den Arbeitspuffer zurück) und beendet den vi
Cursor positionieren	 oder l  oder h  oder j  oder k	Bewegung des Cursors durch die Cursortasten oder den angegebenen Buchstaben l wie <i>Ludwig</i> nach <b>rechts</b> h nach <b>links</b> j nach <b>unten</b> k nach <b>oben</b>
	W oder w  B oder b	Vorwärts springen um ein <b>W</b> ort (Wortanfang, -ende)  ( <i>backwards</i> ) rückwärts springen um ein Wort
	\$ ^ ( ) nG 1G  4G G	Zum Zeilenende springen Zum Zeilenanfang gehen Zum Satzanfang Zum Satzende gehe zur n-ten Zeile Go Gehe zur <b>Zeile 1 – Dateianfang</b> gehe zur 4. Zeile gehe zum <b>Ende</b> der Datei

## Häufig benutzte Kommandos im vi – Fortsetzung

Verwendung	Befehl/ Beispiel	Bedeutung
Blättern	 + f	Blättert eine Bildschirmseite vor
	 + b	Blättert eine Bildschirmseite zurück
Wechsel in den Eingabemodus	A	<i>append</i> Text anhängen am <b>Zeilenende</b>
	a	<b>nach</b> dem Cursor
	I	<i>insert</i> Text einfügen am <b>Zeilenanfang</b>
	i	<b>vor</b> dem Cursor
	o	<i>open</i> Fügt eine Zeile <b>unterhalb des</b> Cursor ein
	O	<b>oberhalb</b> des Cursor ein
	R	<i>replace</i> bestehende Zeichen ab Cursorposition ersetzen
	c Objekt	<i>change</i> ersetzt das nachfolgende Objekt:
	cw	<i>change word</i> nachfolgendes Wort
	cG	Ersetzt den nachfolgenden Text bis zum Ende der Datei
	c^	Ersetzt vom Anfang der Zeile bis zur Cursorposition
	c\$ oder C	Ersetzt von der Cursorposition bis zum Ende der Zeile
	c(	Ersetzt vom Anfang des Satzes bis Cursorposition
c)	Ersetzt von der Cursorposition bis zum Satzende	
Abschluß des Eingabemodus		Schließt die Eingabe ab und wechselt in den Kommandomodus

## Häufig benutzte Kommandos im vi - Fortsetzung

Verwendung	Befehl/ Beispiel	Bedeutung
Löschen	<b>x</b>	<i>durch-x-en</i> Löscht das aktuelle Zeichen, auf dem der Cursor steht
	<b>d</b> <i>Objekt</i>	<i>delete</i> Löscht das nachfolgende Objekt:
	<b>dw</b>	das nachfolgende Wort
	<b>dG</b>	nachfolgenden Text bis zum Ende der Datei
	<b>d^</b>	Anfang der Zeile bis zur Cursorposition
	<b>d\$</b> oder <b>D</b>	ab Cursorposition bis zum Zeilenende
	<b>d(</b>	vom Anfang des Satzes bis zur Cursorposition
	<b>d)</b>	von der Cursorposition bis zum Ende des Satzes
	<b>dd</b>	die aktuelle Zeile
	<b>ndd</b> <b>3dd</b>	<i>n</i> Anzahl Zeilen 3 Zeilen
Übernahme aus dem Puffer	<b>p</b>	<i>paste (einsetzen)</i> Fügt das zuletzt Gelöschte (oder Gespeicherte) nach der Cursorposition ein
	<b>"np</b> <b>"nP</b>	Fügt die <i>n</i> -te Speicherung/ Löschung nach der Cursorposition ein - mit <b>P</b> oberhalb der aktuellen Zeile
	<b>"4p</b>	z.B. 4-letzte Löschung
	<b>xp</b>	Vertauscht 2 Buchstaben an der Cursorposition (z.B. hc in ch)
Rückgängig machen	<b>u</b>	<i>undo</i> Das zuletzt durchgeführte Kommando wird ›ungeschehen‹ gemacht
	<b>U</b>	Die aktuelle Zeile wird aus der Originaldatei wiederhergestellt

## Häufig benutzte Kommandos im vi - Fortsetzung

Verwendung	Befehl/ Beispiel	Bedeutung
Speichern und Einfügen	yy oder Y	<b>yank</b> Setzt die aktuelle Zeile in den Speicherpuffer
	p	Fügt die letzte Speicherung/ Löschung nach der aktuellen Zeile ein
	"[a-z]yObjekt	Speichert das angegebene Objekt in den Puffer (a-z)
	" ayw	Speichert das Wort, auf dem der Cursor steht, in den Pufferspeicher ›a‹
	"[a-z]p	<b>paste</b> Fügt den Inhalt des Pufferspeichers (a-z) nach der Cursorposition ein
"ap ↑ <b>Vorsicht!</b> <b>Anführungszeichen nicht vergessen!</b>	Fügt den Inhalt des Pufferspeichers a nach der Cursorposition ein  Ohne Anführungszeichen würde <b>a</b> für <i>append</i> ausgeführt	
Suchen in der gesamten Datei	/Suchbegriff	Sucht nach dem angegebenen Muster vorwärts
	/	Wiederholt den letzten Suchvorgang (vorwärts)
	?Suchbegriff	Sucht nach dem angegebenen Muster rückwärts
	?	Wiederholt den letzten Suchvorgang (rückwärts)
	%	Wenn der Cursor auf einer Klammer steht, sucht dieses Kommando die dazugehörige schließende oder öffnende Klammer
Suchen in der aktuellen Zeile	fx Fx	<b>find</b> Sucht in der aktuellen Zeile nach dem Zeichen x vorwärts (f) rückwärts (F)
	;	Wiederholt den letzten Suchvorgang nach rechts
	,	nach links

## Häufig benutzte Kommandos im vi - Fortsetzung

Verwendung	Befehl/ Beispiel	Bedeutung
Korrekturmöglichkeitenim Eingabemodus	<b>BS</b> <i>oder</i>	Backspace oder Delete löscht das zuletzt eingegebene Zeichen
	<b>Ctrl</b> + <b>h</b>	löscht das zuletzt eingegebene Wort
	<b>Ctrl</b> + <b>w</b>	löscht die zuletzt eingegebene Zeile
	<b>Ctrl</b> + <b>x</b>	
UNIX-Kommando	<b>!</b> <i>Kommando</i>	Führt das angegebene Kommando aus.
	<b>!</b> <i>date</i>	Gibt das Datum und Uhrzeit auf der untersten Bildschirmzeile aus. Die editierte Datei wird dadurch nicht verändert Mit der Returnntaste kehren Sie zum vi zurück
Sonstiges	<b>Ctrl</b> + <b>l</b>	<i>Buchstabe klein L</i> Bereitet den Bildschirm neu auf
	<b>r</b> <i>Datei</i>	<i>read</i> Liest die angegebene Datei in den Arbeitspuffer und fügt sie nach der Cursorzeile ein
	<b>r!</b> <i>Kommando</i>	Führt das angegebene Kommando aus und fügt das Ergebnis hinter der aktuellen Zeile ein
	<b>J</b>	<i>join</i> Fügt die aktuelle Zeile mit der nachfolgenden Zeile zusammen
	<b>~</b>	Vertauscht Groß-/Kleinbuchstaben
	<b>:</b>	Umschalten in Ex-Modus Hiermit können <i>ed/sed</i> -ähnliche Befehle eingegeben werden.
	<b>:1,\$s/alt/neu/g</b>	Suchen und Ersetzen in der gesamten Datei
	<b>:1,\$s/Herr/&amp;n/g</b>	Ersetzt Herr in Herrn in gesamter Datei

## Einige praktische Optionen für den vi












Kommando	Bedeutung
<b>:set redraw</b>	Änderungen werden am Bildschirm sofort nachvollzogen. Ausnahmen sind Korrekturen im Eingabemodus - meist als Default eingestellt
<b>:set nore</b>	Setzt die obige Funktion zurück
<b>:set wm=<i>n</i></b>	<i>wrap margin - Zeilenumbruch</i> Mit Angabe von <i>n</i> Anzahl Zeichen wird automatisch ein Zeilenumbruch vorgenommen, sobald die maximale Zeilenlänge der Anzahl Zeichen erreicht wurde
<b>:set wm=0</b>	Hiermit erfolgt kein automatischer Zeilenumbruch
<b>:set nu</b>	<i>number</i> Die Datei wird mit laufender Zeilennummer angezeigt
<b>:set nonu</b>	Setzt die obige Funktion zurück
<b>:set showmode</b>	Hiermit wird in der letzten Bildschirmzeile beim Eingabemodus der Hinweis ›Input Mode‹ angezeigt
<b>:set nomagic</b>	Die Sonderzeichen <i>.</i> , <i>[]</i> und <i>*</i> haben dann keine Sonderbedeutung mehr, d.h., bei dem Such- und Ersetzungsmechanismus des ex-Modus werden sie nicht als Metazeichen behandelt. Z.B. gilt der Stern nicht mehr als Wiederholungsfaktor, sondern wird als Stern erkannt.
<b>:set magic</b>	Setzt die obige Funktion zurück
<b>:set all</b>	Zeigt alle eingestellten Parameter an

In der Datei **\$HOME/.exrc** können diese Voreinstellungen generell gesetzt werden. Die Kommandos werden dann **ohne :** eingegeben. Mehrere Angaben können in einer Zeile stehen. z. B. **set number showmode**

Eine weitere Möglichkeit ist, die Optionen der Variablen **EXINIT** mitzugeben:

```
EXINIT="set nu showmode"
export EXINIT
```

## Befehlszeileneditor im vi, emacs und der bash

Funktion	Taste im vi	Taste im emacs	Taste in bash
<b>Positionieren:</b>			
Vorherige Befehlszeile (nach oben)	<b>k</b>	<b>&lt;Ctrl&gt; p</b> <i>previous</i>	
Nächste Befehlszeile (nach unten)	<b>j</b>	<b>&lt;Ctrl&gt; n</b> <i>next</i>	
Zeichen nach links wandern (zurück)	<b>h</b>	<b>&lt;Ctrl&gt; b</b> <i>back</i>	
Zeichen nach rechts wandern (vorwärts)	<b>l</b> (kleines L)	<b>&lt;Ctrl&gt; f</b> <i>forward</i>	
Anfang der Zeile	<b>^</b>	<b>&lt;Ctrl&gt; a</b> <i>anfang</i>	 <b>a</b>
Ende der Zeile	<b>\$</b>	<b>&lt;Ctrl&gt; e</b> <i>ende</i>	 <b>e</b>
<b>Korrigieren:</b>			
Zeichen löschen	<b>x</b>	<b>&lt;Ctrl&gt; d</b> <i>delete</i>	 
Zeichen einfügen	<b>i</b>	automatisch voreingestellt	autom. voreingestellt
Wort löschen	<b>dw</b>	<b>&lt;ESC&gt; d</b> <i>(bis Wortende)</i>	 <b>d</b>
Bis zum Ende der Zeile löschen	<b>d\$</b>	<b>&lt;Ctrl&gt; u</b> <i>ende</i>	 <b>k</b>
Zuletzt Gelöschtes einfügen	<b>p</b>	<b>p</b> <i>paste</i>	

## Erläuterung der Symbole für Tasten



Insert /Einfügen



Alternate



Backspace/Löschen

**<Ctrl>** Control/Steuerung

Delete/Entfernen

**<ESC>** Escape

## Wichtige Dateien für den Benutzer

Datei	Bedeutung
<b>/etc/passwd</b>	Datei, die maßgebend dafür ist, ob Sie sich an einem Rechner anmelden dürfen. Nur der Systemverwalter kann hier Einträge über entsprechende Systemkommandos vornehmen. Der Benutzer kann nur über das Kommando <code>passwd</code> in dieser Datei ein verschlüsseltes Passwort verändern
<b>/etc/group</b>	Hier werden Benutzer bestimmten Gruppen zugeordnet (zusätzlich zur Gruppennr., die in <code>/etc/passwd</code> dem Benutzer zugewiesen wird (4. Spalte))
<b>/etc/hosts</b>	In dieser Datei sind alle Rechner eingetragen, die im Netz angesprochen werden können
<b>/etc/hosts.equiv</b>  Beispiel eines Eintrages <i>hostname [username]</i> <i>-hostname [username]</i> <i>+ username</i>	Datei, um den Zugriffsschutz im Netz zu steuern für die Netzkommandos wie <i>rcp</i> , <i>rsh</i> , <i>rlogin</i> oder <i>telnet</i> Für alle oder nur den angegebenen Benutzer des Rechners erlaubt nicht erlaubt Für den angegebenen Benutzer von allen Rechnern aus erlaubt
<b>/dev/lp</b>	Oft dem Drucker als Gerätebezeichnung zugeordnet
<b>/dev/mt0</b> <b>/dev/rmt/0</b>	Meist dem Bandlaufwerk als Gerätebezeichnung zugeordnet
<b>/dev/null</b>	Der umweltfreundliche Papierkorb unter UNIX. Alle Ausgaben, die in diese Datei umgeleitet werden, sind null und nichtig. Eingabe-Umleitungen bewirken ein EOF (End of File)
<b>/dev/fd0</b>	Meist dem Disketten-Laufwerk als Gerätebezeichnung zugeordnet

## Wichtige Dateien für den Benutzer

Datei	Bedeutung
<code>/dev/tty</code> oder <code>/dev/pts</code>	Gerätebezeichnung für Terminals (bzw. Windows - Pseudo-Terminals pts)
<code>/tmp</code>	Ein temporäres Directory, das allen Benutzern Schreib-, Lese- und Ausführrechte erlaubt, wobei Dateien nur vom Besitzer oder root gelöscht werden dürfen
<code>/root</code>	Anmeldeverzeichnis der root unter SUSE linux
<b>Dateien in Ihrem Home-Directory:</b>	Das Home-/Login-Directory wird beim Anmelden zugewiesen aus der Datei <code>/etc/passwd</code> (6. Spalte)
<code>.bashrc</code>	Wird von der Bash bei jedem Aufruf einer neuen Bash (Subshell) gelesen, Inhalt der Dateien sind z.B. Alias-Zuweisungen, Shell-spezifische Variablen, Funktionen und Optionen → <b>.kshrc</b>
<code>.cshrc</code>	Wird von der C-Shell jedesmal gelesen, wenn eine C-Shell gestartet wird → <b>.login</b>
<code>.dtprofile*</code>	Wird als Login-Profil-Datei von CDE gelesen. Sollen zusätzlich auch andere Login-Dateien wie <b>.profile</b> , <b>.kshrc</b> etc. gelesen werden, dann muss die <b>Variable DTSOURCEPROFILE=true</b> hier aktiviert werden
<code>.kshrc*</code>	Wird von der Korn-Shell bei jedem Aufruf einer neuen Korn-Shell (Subshell) gelesen, wenn im <b>.profile</b> die <b>Variable ENV=\$HOME/.kshrc</b> gesetzt wurde.  In <b>.kshrc</b> werden u.a. Alias-Zuweisungen und Korn-Shell-spezifische Variablen, Funktionen und Optionen gesetzt. → <b>.profile</b>



## Shell: Bourne (sh), Korn (ksh), Bash (bash) Sonderzeichen

Gelten Sonderzeichen nur für eine bestimmte Shell, wird unter Bedeutung daraufhingewiesen, z.B. *nicht sh*

Zeichen	Bedeutung
<b>Anzeige am Bildschirm:</b>	<b>Bereitzeichen (Variable PS1/PS2)</b> → Seite 70
\$	für den ›normalen‹ Benutzer ( <i>csh</i> : %)
#	für den Systemverwalter (root)
>	Folgezeichen in der nächsten Zeile, wenn das Kommando noch nicht abgeschlossen war
>	<b>Umleitung der Ausgabe</b> Neuerstellung (bzw. Überschreiben)
>>	<b>Umleitung der Ausgabe</b> Anhängen an eine bestehende Datei
2>	<b>Umleitung der Fehlerausgabe</b>
<	<b>Umleitung der Eingabe</b>
;	<b>Verkettung von Kommandos</b> Mehrere Kommandos in einer Zeile
	<b>Pipe-Zeichen</b>
&& <i>kom1 &amp;&amp; kom2</i>	<b>Und-Verknüpfung von Kommandos</b> kom2 wird nur dann ausgeführt, wenn kom1 erfolgreich war
 <i>kom1    kom2</i>	<b>Oder-Verknüpfung von Kommandos</b> kom2 wird nur dann ausgeführt, wenn kom1 nicht erfolgreich war
*	<b>Metazeichen - Dateinamenexpansion</b> beliebige Zeichenfolge
?	<b>ein</b> beliebiges Zeichen
[ a b c ]	<b>eines</b> der in Klammer angegebenen Zeichen
[ a-z ]	<b>eines</b> der in Klammer angegebenen Zeichen ›von - bis‹
[ !a-z ]	<b>ein</b> beliebiges Zeichen außer den in Klammer angegebenen Zeichen
!	<b>History-Mechanismus (statt r) nur csh</b>
\$Name	<b>Ersetzung durch den der Variablen Name zugewiesenen Wert</b>
\Befehl\ \$( Befehl )	<b>Ersetzung durch das Ergebnis des Befehls</b> <b>neuere Schreibweise</b>

Fortsetzung

Zeichen	Bedeutung
.	aktuelles Directory
..	darüberliegendes Directory
~ ~Benutzer	Home-/Login-Directory <i>nicht sh</i> Home-/Login-Directory des betreffenden Benutzers
%jobnummer	Kennzeichen für Jobcontrol-Nummer <i>nicht sh</i>
\	Aufhebung der Bedeutung eines nachfolgenden Sonderzeichens bzw. bei <i>ksh bash csh</i> zusätzlich: Aufhebung eines nachfolgenden Alias
"text ... text"	Keine Ersetzung der Metazeichen (Dateinamenexpansion): * ? [ ] <b>Aber</b> Ersetzung von Shell-Variablen (\$) Ersetzung durch Ergebnisse von Kommandos ( ` ` , bzw. \$( )) Ersetzung von Alias
'text ... text'	Keinerlei Substitution

## Variable

### Positionsparameter

Positionsparameter	Bedeutung
\$0	Name der Shell-Prozedur
\$1	Wert des 1. Parameters
\$2 ...	Wert des 2. Parameters ...
\$9	Wert des 9. Parameters
\$*	Werte aller angegebenen Parameter
\$#	Anzahl der Parameter
\$?	Exit-Status des letzten Kommandos
\$\$	Prozeßnummer der Shell-Prozedur

## Vordefinierte Shell-Variable

Shell-Variable	Bedeutung
<b>\$DISPLAY</b>	Diese Variable muß gesetzt sein, wenn eine grafische Oberfläche benutzt wird. Als Wert wird der Rechnername (bzw. Name des X-Terminals) eingetragen, gefolgt von :0.0 z.B. <b>DISPLAY=amadeus:0.0</b> <b>export DISPLAY</b>
<b>\$ENV</b>	Hier wird als Wert der Dateiname angegeben, der Voreinstellungen für die Korn-Shell enthält. In der Regel <b>\$HOME/.kshrc</b> → <b>Wichtige Dateien .profile, .kshrc</b>
<b>\$EXINIT</b>	Als Wert werden Voreinstellungen für den ex- bzw. vi-Editor eingetragen → <b>vi</b>
<b>\$HISTSIZE</b>	Als Wert wird die Anzahl der Kommandos eingetragen, auf die über den History-Mechanismus und den Befehlszeileneditor zurückgegriffen werden darf <b>nicht sh</b>
<b>\$HOME</b>	Beim Login wird der absolute Pfadname des Login-Directories aus der /etc/passwd der Variablen zugeordnet. Diese Variable wird u.a. von cd (ohne weitere Angaben) als default verwendet
<b>\$IFS</b>	Hier sind die Separatorzeichen zugewiesen. Standard: Leerzeichen, Tabulator und Neue Zeile
<b>\$LANG</b>	Als Wert wird die jeweilige Sprache zugewiesen, in der die Systemmeldungen erfolgen sollen - soweit im System enthalten
<b>\$LOGNAME</b>	Hier wird als Wert der Login-Name des Benutzers zugewiesen
<b>\$OLDPWD</b>	Enthält als Wert das vor einem cd benutzte Directory

## Vordefinierte Shell-Variable - Fortsetzung

Shell-Variable	Bedeutung
<b>\$PATH</b>	<p>Als Wert sind all jene Directories mit absolutem Pfadnamen aufgeführt, unter denen die Shell nach Kommandos sucht. Getrennt werden die einzelnen Directories durch den Doppelpunkt (:)</p> <p><i>Beispiel einer Neubesetzung:</i>  <b>PATH=/usr/bin:/bin:/usr/ucb/bin</b></p> <p><i>Beispiel einer Ergänzung:</i>  <b>PATH=\$PATH:\$HOME/Eigenbefehle</b></p>
<b>\$PS1</b>	<p>Als Wert ist das Bereitzeichen der Shell zugewiesen</p> <p><i>Beispiel einer Neubesetzung:</i>  <b>PS1="` \$PWD ` &gt;"</b> <i>nicht sh</i></p> <p>Hiermit wird jeweils das aktuelle Directory als Bereitzeichen angezeigt  z.B. /usr/kurs/hans/Texte &gt;</p>
<b>\$PS2</b>	<p>Als Wert ist das Zeichen &gt; zugewiesen, (Fortsetzungszeile eines Kommandos)</p>
<b>\$PWD</b>	<p>Als Wert wird jeweils das aktuelle Directory zugeordnet <i>nicht sh</i></p>
<b>\$SHELL</b>	<p>Manche Programme fragen den Wert dieser Variable ab, um die entsprechende Shell zu starten (z.B. csh, ksh)</p>
<b>\$TERM</b>	<p>Als Wert ist hier der Terminaltyp der Dialogstation zugeordnet. Die richtige Zuordnung ist wichtig bei vielen Bildschirm-orientierten Programmen (z.B. grafische Oberfläche, vi, more etc.)</p>
<b>\$TZ</b>	<p>Dieser Variable können Angaben zur Zeitzone zugewiesen werden (z.B. für automatische Berechnung der Ortszeiten bei mail)</p>
<b>\$VISUAL</b>	<p>Ist diese Variable belegt, kann die Befehlszeile editiert werden <i>nicht sh</i>  → <b>Setzen von Optionen</b></p>

## Arbeiten mit Variablen

Befehlseingabe	Bedeutung/Hinweise
<p><b>Name=Wert</b></p> <p><i>Beispiel:</i>  <b>Name="Hans Meier"</b></p> <p><b>Datum="\`date`"</b>  <b>Datum=\$( date )</b></p>	<p>Der Variablen <i>Name</i> wird ein Wert zugewiesen.</p> <p>Enthält der Wert Leer- oder Sonderzeichen, muß der Wert in Anführungszeichen gesetzt werden</p> <p>Bei der Zuweisung eines Wertes kann auch das Ergebnis eines Kommandos eingesetzt werden: <code>`cmd`</code></p>
<p><b>export Name</b></p> <p><i>Alternativen in der ksh:</i>  <b>export Name=Wert</b>  <b>typeset -x</b>  <i>Name=Wert</i></p>	<p>Um Variable auch für Unterprogramme zur Verfügung zu stellen, werden sie exportiert</p> <p>In der Korn-Shell kann dies gleich bei der Zuweisung erfolgen</p> <p style="text-align: right;"><b>nicht sh</b></p> <p>oder über eine generelle Voreinstellung mit</p> <p><b>set -o allexport</b>      <b>nicht sh</b>  → <b>Setzen von Optionen</b></p>
<p><b>typeset -i</b>  <i>Name[=Wert]</i>  oder Eingabe über <i>alias</i>  <b>integer Name[=Wert]</b>  <i>Beispiele</i>  <b>integer erg=5</b></p> <p><b>erg=\$erg+3</b>  <b>erg=\$erg-3</b>  <b>erg=\$erg*3</b>  <b>erg=\$erg/3</b>  <b>erg=\$erg%3</b></p>	<p><b>-i integer</b></p> <p>Der Variablen können nur ganze Zahlen zugewiesen werden. Mit dieser Variablen kann direkt gerechnet werden:</p> <p><b>+ Addition</b> (<math>\\$erg=8</math>)  <b>- Subtraktion</b> (<math>\\$erg=2</math>)  <b>* Multiplikation</b> (<math>\\$erg=15</math>)  <b>/ Division</b> (<math>\\$erg=1</math>)  <b>% Modulo</b> (<math>\\$erg=2</math>)</p>
<p><b>typeset -l</b>  <i>Name[=Wert]</i></p>	<p><b>-l lower case</b></p> <p>Alle zugewiesenen Werte der Variable werden in Kleinbuchstaben umgesetzt      <b>(nicht bash)</b></p>
<p><b>typeset -u</b>  <i>Name[=Wert]</i></p>	<p><b>-u upper case</b></p> <p>Alle zugewiesenen Werte der Variable werden in Großbuchstaben umgesetzt      <b>(nicht bash)</b></p>

Fortsetzung

Befehlseingabe	Bedeutung/Hinweise
<i>Beispiel:</i> <b>echo \$Name</b>	Der Wert einer Variablen kann mit <b>\$Name</b> abgefragt werden
<i>Beispiel:</i> <b>touch \${Name}datei</b>	Mit { } kann der Name der Variablen abgegrenzt werden, um ihn z.B. bei einem neuen Dateinamen zu integrieren
<b>unset Variable</b>	<i>Aufheben</i> <b>nicht sh/csh</b> Hiermit wird die Variable wieder aufgehoben

## Setzen von Optionen

Mit dem Kommando **set -o**

*nicht sh*

können u.a. folgende Optionen gesetzt werden:

(mit **set + o** werden gesetzte Optionen wieder ausgeschaltet)

Option	Bedeutung
<b>allexport</b>	Alle gebildeten Variablen werden grundsätzlich exportiert
<b>bgnice</b>	Hintergrundprozesse laufen mit einer niedrigeren Priorität
<b>emacs</b> oder <b>vi</b>	Wird diese Option gesetzt, kann die Befehlszeile mit dem angegebenen Programm editiert werden. Hierfür muß, um zu korrigieren, die ESC-Taste gedrückt werden. Der Befehlszeileneditor kann auch durch Setzen der Variable <b>VISUAL=vi</b> eingeschaltet werden
<b>ignoreeof</b>	Mit der Tastenkombination Ctrl + d kann die Shell nicht mehr beendet werden (wird ignoriert)
<b>noclobber</b>	Bereits bestehende Dateien können über Umleitungszeichen > nicht mehr überschrieben werden

Als Voreinstellung können Variable, set-Kommandos und alias-Funktionen in die Datei **\$HOME/.kshrc** eingetragen werden, bzw. in der Bash in **\$HOME/.bashrc**

→ **Wichtige Dateien**

## Konstrukte/Ablaufsteuerung der Shell (für sh, ksh und bash)

Eingabe	Funktion
<pre>if <i>Befehl1</i>   then <i>Befehlsfolge2</i>   [ else <i>Befehlsfolge3</i> ]   oder   [ elif <i>Befehl4</i>     then <i>Befehlsfolge5</i> ] fi</pre> <p><i>Beispiel:</i></p> <pre>if test -f <i>\$Antwort</i>    then pr -n <i>\$Antwort</i>   pg   else echo "<i>\$Antwort ist keine Datei</i>" fi</pre>	<p><b>If-Verzweigung</b> wenn das und das zutrifft dann tue ... [ sonst tue ... ] [ sonst wenn dann tue ... ] fertig (Ende der if-Verzweigung) Wenn die Datei (<i>\$Antwort</i>) existiert und es eine normale Datei ist, dann soll sie angezeigt werden, sonst soll eine entsprechende Nachricht ausgegeben werden</p>
<pre>for <i>Name</i>  do   <i>Befehlsfolge</i> done</pre>	<p><b>Schleifenverarbeitung</b> <b>Solange</b> die Variable <i>Name</i> einen Wert enthält (sie erhält nacheinander den Wert von \$1-\$n, also die angegebenen Parameter beim Aufruf des Kommandos) tue ... fertig - gehe zu Beginn der Schleife</p>
<pre>for <i>Name</i> in <i>Wert1...Wertn</i>  do   <i>Befehlsfolge</i> done</pre>	<p><b>Solange</b> die Variable <i>Name</i> einen Wert enthält (hier wird nacheinander der <i>Wert1 .. Wertn</i> zugewiesen) tue ... fertig - gehe zu Beginn der Schleife</p>
<pre>while <i>Kommando</i>  do   <i>Befehlsfolge</i> done</pre>	<p><b>Solange</b> das <i>Kommando</i> erfolgreich ist (Exit-Status 0) tue ... fertig - gehe zu Beginn der Schleife</p>

Fortsetzung

Eingabe	Funktion
<pre>until Kommando do   Befehlsfolge done</pre>	<p><b>Solange</b> das <i>Kommando</i> nicht erfolgreich ist (Exit-Status nicht 0)</p> <p>tue ...</p> <p>fertig - gehe zu Beginn der Schleife</p>
<pre>case Muster in   M1) Befehlsfolge1;;   M2 M3) Befehlsfolge2;;   ... esac</pre> <p style="text-align: right;">↑</p>	<p><b>Case-Verarbeitung</b> (Auswahl)</p> <p><b>Falls</b> die Zeichenkette <i>Muster</i> übereinstimmt mit <i>M1</i>, dann führe die Befehlsfolge1 aus, stimmt das <i>Muster</i> mit <i>M2</i> <b>oder</b> <i>M3</i> überein, führe Befehlsfolge2 aus</p> <p>Ende der case-Bedingung</p>
<p>2 Semikolons sind als Abschluß der Befehlsfolge nötig!</p>	

### Sonstige Steuerungen in Prozeduren

Eingabe	Funktion
<pre>function name () {   Kommandofolge }</pre>	<p>Funktionen</p> <p><b>Bildet eine Funktion, die ähnlich eines Shell-internen Kommandos genutzt werden kann</b></p> <p>s. a. Alphabetische Kommandos</p>

## Besonderheiten der Bash

Kommando	Auswirkung
<b>basename</b>	Gibt den Namen einer Datei ohne Pfadangabe zurück
<b>basename /bin/date</b>	Ergebnis ist: <i>date</i>
<b>declare</b>	Zuweisung von Variablen <i>declare</i> entspricht <i>typeset</i> (gleiche Optionen) - beide Kommandos werden akzeptiert
<b>dialog [--clear] \</b> <b>[setterm --clear]\</b> <b>[--title text] \ [Optionen]</b>	Ist ein eigenes Kommando (nicht Bestandteil der <i>bash</i> ). Es gibt ein Dialogfenster aus, um Nachrichten, Textdateien oder Abfragen für den Benutzer auszugeben und evtl. Texteingaben vom Benutzer aufzunehmen.
<b>Optionen:</b>	
<b>--msgbox</b> <i>Text Höhe Breite</i>	Meldetext mit Return bestätigen
<b>--infobox</b> <i>Text Höhe Breite</i>	Meldetext ohne Bestätigung
<b>--yesno</b> <i>Text Höhe Breite</i>	Dialogbox für ja/nein Entscheidung
<b>--textbox</b> <i>Textdatei Höhe \</i> <i>Breite</i>	Zeigt die Textdatei an (ohne Editiermöglichkeit - mit Scrolling über Cursortasten)
<b>--inputbox</b> <i>Text Höhe</i> <i>Breite</i>	Ermöglicht Eingabe einer Textzeile, die als Fehlerausgabe weitergeleitet wird
Höhe=Zeilenanzahl Breite=Zeichen pro Zeile Für fortgeschrittene Programmierung gibt es auch noch Menü- und Checklisten.	Über 2> Datei kann die Texteingabe bearbeitet werden. Rückgabewert bei OK oder Yes ist 0. Rückgabewert bei Cancel, No oder Esc ist 255
Zu beachten: Es dürfen keine deutschen Umlaute und Tabs enthalten sein. Die Höhe und Breite des Dialogfeldes darf nicht größer sein, als die Maße des Terminals, in dem dialog aufgerufen wurde.	
<b>dirname</b> <b>dirname /usr/bin/zip</b>	Liefert nur den Pfad einer Datei Ergebnis: <i>/usr/bin</i>

Kommando	Auswirkung
<b>dirs</b>	Listet mit <i>pushd</i> gespeicherte Verzeichnisse/Directories (s. <i>popd</i> und <i>pushd</i> )
<b>local var[=wert]</b>	Innerhalb von function Zuweisung von Variablen
<b>popd</b>	Wechselt in das letzte mit <i>pushd</i> gespeicherte Verzeichnis und löscht es aus dem Stack wieder heraus. Mit <i>popd</i> können somit mehrere Verzeichnisse nacheinander abgearbeitet werden, die zuvor gespeichert wurden.
<b>pushd Directory</b>	Speichert das angegebene Verzeichnis/Directory in einer eigenen Liste (Stack), die mit <i>popd</i> zurückverfolgt werden kann
<b>pushd .</b>	Speichert das aktuelle Verzeichnis/Directory
<b>printf format [Argumente]</b> z. B. <b>printf \</b> <b>"%6,2f Euro\n" 30 150</b>	Aufbereitung der Druckausgabe Druckformatangaben wie in der Programmiersprache C Druckausgabe: <i>300,00 Euro</i> <i>150,00 Euro</i>
<b>printenv</b>	Anzeige der globalen Variablen (entspricht <i>env</i> in <i>ksh</i> )
<b>setterm [option]</b> <i>Optionen:</i> <b>-bold off   on</b> <b>-clear</b> <b>-default -</b>  <b>-half-bright on   off</b> <b>-underline on   off</b> <b>-reverse on   off</b>	Terminal-Voreinstellung (s.a. man <i>setterm</i> ) Fettschrift an/aus Löscht Bildschirm Setzt auf Defaulteinstellung zurück Text hervorgehoben Text unterstrichen Inverse Textdarstellung
<b>source datei</b>	Entspricht <i>. datei</i>
<b>trap -l</b>	Zeigt alle Signale an

## Besonderheiten der Bash

Kommando	Auswirkung
<b>ulimit</b> <i>option grenzwert</i> Optionen: <b>-f</b> <i>dateigröße</i>	Grenzwerte setzen (Angabe in kByte) - - verhindert Erzeugung von Dateien, die größer sind als der angegebene Grenzwert

### Weitere Besonderheiten der Bash

Begriff	Auswirkung
<b>~/.bashrc</b>	Entspricht in etwa ( <i>.kshrc</i> ) - Wird von der interaktiven Shell gelesen
<b>&gt;&amp;</b>	Umleitung von Standardausgabe und Standardfehler
<b>{}</b> <b>ls {a,b}{1,2,3,4}</b>	Erweiterungsmechanismus ( <i>brace expansion</i> ) ergibt: <i>a1 a2 a3 a4 b1 b2 b3 b4</i>
<b>\$( )</b> <b>echo \$(4+7*2)</b>	Berechnung arithmetischer Ausdrücke Ergebnis: 18
<b>PS1='\u@\h:\W\$'</b>	Zuweisen des Prompts mit Prompting-Kürzel (Auswahl der oft genutzten Kürzel): <b>\h</b> Hostname vor dem Punkt <b>\H</b> Vollständiger Hostname <b>\A</b> die aktuelle Zeit im Format: HH:MM (24 Std.) <b>\u</b> der Benutzername <b>\w</b> das aktuelle Verzeichnis <b>\W</b> der Basisname des aktuellen Verzeichnisses <b>\\$</b> ist die Benutzerkennung 0 (root), wird # ausgegeben, andernfalls das \$-Zeichen

## C-Shell (csh und tcsh)

Während bash oder ksh ohne weiteres Shell-Skripte richtig interpretiert und ausführt, führen sh-Skripte, die mit csh gestartet werden, oft zu fehlerhaftem Abbruch. Werden verschiedene Shells genutzt, sollte in der ersten Zeile einer Shell-Prozedur der run-Befehl für die entsprechende Shell eingegeben werden, zum Beispiel:

```
#!/bin/sh oder #!/usr/bin/bash oder #!/usr/bin/ksh
           oder #!/usr/bin/csh  und  #!/usr/bin/tcsh
```

Der Prompt der C-Shell ist in der Regel das ›%‹, falls der Variablen PS1 kein anderer Wert zugewiesen wurde. Dateinamenexpansion, Ein-/Ausgabeumleitung außer Fehlerausgabeumleitung, Aufruf eines Hintergrundprozesses und die Positionsparameter werden genauso wie in der Bourne-, Korn-Shell oder Bash behandelt.

## Unterschiede zur sh, ksh und bash

Eingabe unter ksh/bash	Eingabe unter csh/tcsh
<b>Fehlerumleitung:</b> 2>	>&
<b>Ablaufsteuerung:</b>	
<b>If-Bedingung</b>	<i>if ( Ausdruck ) Befehl</i> oder <b>if ( Ausdruck ) then</b> <i>Befehlsfolge</i> <b>[else</b> <i>Befehlsfolge</i> <b>endif</b>
<i>if Befehl</i> <b>then</b> <i>Befehlsfolge</i> <i>[else Befehlsfolge]</i>	<b>[else</b> <i>Befehlsfolge</i> <b>endif</b>
<b>fi</b> oder <b>if</b> <i>Befehl</i> <b>then</b> <i>Befehlsfolge</i> <i>[elif Befehl</i> <b>then</b> <i>Befehlsfolge</i> <i>[else Befehlsfolge]</i> <b>fi</b>	<b>if ( Ausdruck ) then</b> <i>Befehlsfolge</i> <b>else if ( Ausdruck ) then</b> <i>Befehlsfolge</i> <b>[else</b> <i>Befehlsfolge</i> <b>endif</b>
<b>Schleifenverarbeitung</b>	
<b>for</b> <i>Name in Argumente</i> <i>do Befehlsfolge</i> <b>done</b>	<b>foreach</b> <i>Name ( Argumente )</i> <i>Befehlsfolge</i> <b>end</b>
<b>while</b> <i>Befehl</i> <i>do Befehlsfolge</i> <b>done</b>	<b>while ( Ausdruck )</b> <i>Befehlsfolge</i> <b>end</b>

Eingabe unter ksh/bash	Eingabe unter csh/tcsh
<b>Case-Verarbeitung</b> <b>case</b> <i>Textmuster</i> <b>in</b> <i>Muster_1</i> ) Befehlsfolge ;; <i>Muster_2</i> ) Befehlsfolge ;; ... *) Befehlsfolge ;; <b>esac</b>	<b>switch</b> ( <i>Textmuster</i> ) <b>case</b> <i>Muster1</i> ; Befehlsfolge; <b>breaksw</b> ... <b>case</b> <i>Mustern</i> ; Befehlsfolge; <b>breaksw</b> <b>default:</b> Befehlsfolge (optional) <b>endsw</b>
<b>Goto-Anweisung</b> nicht möglich	<b>goto</b> <i>Marke</i> ... <b>Marke:</b> Befehlsfolge
Repeat-Anweisung nicht vorgesehen	<b>repeat</b> <i>n</i>
<b>Variablen:</b>	
<b>Positionsparameter</b> <b>\$0, \$1, ...</b>	<b>\$argv[0], \$argv[1], ...</b> und <b>\$0, \$1, ...</b>
<b>Vordefinierte Variable</b> <b>\$HOME</b> oder (nicht sh): <b>~[benutzer]</b>	Viele auch in Kleinbuchstaben <b>\$home</b> oder <b>~[benutzer]</b>
<b>Setzen von Variablen</b> <b>export</b> <i>PATH</i> = <i>\$PATH</i> : <i>/usr</i> \ <i>/kurs/bin</i>	<b>setenv</b> <i>path</i> \ ( <i>\$path</i> <i>/usr/kurs/bin</i> )
<b>Bereitzeichen</b> mit aktuellen Pfadnamen zuweisen unter ksh: <b>export</b> <i>PS1</i> = <i>' \$PWD &gt; '</i> unter bash: <b>export</b> <i>PS1</i> = <i>'\u@\h:W'</i>	<b>setenv</b> <i>prompt</i> \ " <b>%B</b> <b>%m</b> <b>%b</b> <b>%C2</b> <b>%#</b> " (nicht Original-BSD-C-Shell) hierbei steht: <b>%B</b> ( <b>%b</b> ) für Fettschrift einstellen (aufheben) <b>%m</b> für den Hostnamen bis zum ersten >.< <b>%C2</b> für die letzten beiden Teile des Pfadnamens
<b>Rechnen mit Variablen</b> <b>zahl=10</b> <b>zahl=</b> `expr \$zahl + 1` <b>typeset -i zahl=10</b> <b>zahl=\$zahl+1</b>	<b>set</b> <i>Name</i> = <i>Wert</i> <b>set</b> <i>Name</i> = <i>~Expression`</i> <b>set</b> <i>zahl</i> = 10 <b>set</b> <i>zahl</i> = <i>~expr \$zahl + 1`</i>
<b>Aliasbildung/-aufhebung:</b>	
<b>alias</b> <i>kürzel</i> = " <i>Befehl</i> " <b>alias</b> <i>ll</i> = " <i>ls -l</i> " <b>unalias</b> <i>kürzel</i>	<b>alias</b> <i>kürzel</i> <i>Befehl</i> <b>unalias</b> <i>kürzel</i>