

Kleiner Einstieg in die UNIX-Welt

UNIX-Shells

Neben der graphischen Benutzeroberfläche (GUI) existiert immer noch die klassisch originäre Benutzeroberfläche bei der der Programmaufruf durch Eingabe des Programmnamens über die Tastatur erfolgt. Dazu öffnet man ein Konsolfenster. Darin startet automatisch ein Program - die sogenannte Shell. Sie stellt unter UNIX die **Schnittstelle** zwischen dem Anwender und dem eigentlichen UNIX-Betriebssystem dar. Die Shell wartet auf Eingaben (=UNIX-Kommandos) des Benutzers - angezeigt durch den **Prompt**. Ist die Eingabe vollständig, wird durch Drücken der **Return-Taste** die Eingabe abgeschlossen. Nun interpretiert die Shell die Zeichenfolge, d.h. untersucht sie auf Fehler und gibt anschließend eine Fehlermeldung aus oder ruft das angegebene Programm auf.

Die Shell bietet auch eine Reihe von Zusatzfunktionen an, die den Umgang mit dem UNIX-System erleichtern. Da wären die Verwendung von Platzhaltern (Metazeichen) bei Dateinamen, um mehrere ähnliche Dateien auf einmal ansprechen zu können, die Umleitung der Ein- und Ausgabe von Programmen aus bzw. in Dateien und den Start von Programmen mit langer Laufzeit im Hintergrund, um zwischenzeitlich mit anderen Programmen zu arbeiten. Außerdem stehen eine Reihe von eingebauten Kommandos als Ergänzung zu den normalen UNIX-Befehlen zur Verfügung, um richtige Shell-Programme (Shell-Skripts genannt) zu schreiben, über die Programmabläufe mit Konstrukten genau wie bei Programmiersprachen gesteuert werden können.

Die bekanntesten Shells sind die Bourne-Shell (**sh**), die C-Shell (**csh**), die Korn-Shell (**ksh**), die **bash** und die **tcsh**. Letztere ist im SCI voreingestellt, weil bei ihr recht bequem frühere Eingabezeilen noch einmal verwendet und ggf. sogar verändert werden können (editierbar), ohne die komplette Zeile neu eingeben zu müssen. Zudem können Abkürzungen für häufig benutzte Kommandos eingeführt werden.

Kleine Sammlung von UNIX-Kommandos

-- *Einfache Kommandos* --

date	# Datum und Uhrzeit
who	# Wer ist eingeloggt?
who am i	# Gibt nur den eigenen Namen aus
sort textdatei	# Sortiert eine Textdatei zeilenweise alphabetisch
wc textdatei	# Gibt die Anzahl der vorhandenen Zeilen, Wörter # und Zeichen aus

--*Navigieren im Dateisystem* --

pwd	# Wo bin ich im Dateibaum?
------------	----------------------------

```

cd # Gehe direkt ins Heimat-Verzeichnis (homedirectory)
cd .. # Gehe ins Vorgänger-Verzeichnis
cd ../bin # Relativer Weg (path) ins Verzeichnis bin
cd /users1/m_muster/bin # Absoluter Weg (path) ins Verzeichnis bin
# beim User m_muster

cd ~m_muster/bin # Gehe direkt ins Verzeichnis bin beim User m_muster
cd ~vorlesungen # Gehe im homedirectory ins Verzeichnis Vorlesungen
cd - # Gehe ins Ausgangs-Verzeichnis zurück

```

--Änderungen im Dateisystem --

```

mkdir neuerOrdner # Lege hier (im workingdirectory) ein neues Verzeichnis an
mkdir ~heuerOrdner # Lege im homedirectory ein neues Verzeichnis an
touch textdatei # Lege im workingdirectory eine Textdatei an
ls -l # Liste im workingdirectory Dateien (l=)ausführlich auf
ls -F # Auflistung, wobei Verzeichnisse und ausführbare Daten
# gekennzeichnet sind
ls -a # Liste im workingdirectory alle Dateien auf
cp original kopie # Kopiere im workingdirectory eine Datei
cp D1 D2 D3 V2 # Kopiere die Dateien D1 bis D3 ins Verzeichnis V2
cp D* V2 # Kopiere alle Dateien beginnend mit D ins Verzeichnis V2
cp DA V2/AD # Kopiere Datei ins Verzeichnis V2 mit neuem Namen AD
cp -r VA VB # Kopiere gesamtes Verzeichnis VA in Verzeichnis VB
mv D1 DA # Benenne D1 um in DA
mv D1 VA # Verschiebe D1 ins Verzeichnis VA
mv D* VA # Verschiebe alles mit Anfang D ins Verzeichnis VA
ln -s /usr/bin/cat zeige # Lege einen Verweis (link) auf das Kommando cat an
# d.h. kreiere zusätzlichen Namen für das cat-Kommando

rm D1 # Lösche Datei D1
rm -i D1 # Lösche Datei D1 erst nach Bestätigung
rmdir V1 # Lösche leeres Verzeichnis V1
rm -r V1 # Lösche Verzeichnis V1 samt Inhalt
chmod 640 D1 # Lege Zugriffsrechte für D1 wie folgt fest:
# für Eigentümer: 6 (= 4(lesend) + 2(schreibend))
# für Gruppe: 4(lesend) und Welt 0(keine)

chmod 510 D1 # Lege Zugriffsrechte für D1 wie folgt fest:
# für Eigentümer: 5 (= 4(lesend) + 1(ausführend))
# für Gruppe: 1(ausführend) und Welt 0(keine)

```

--Dateiinhalte betrachten --

cat <i>LiesMich</i>	# Ausgabe der Datei auf den Bildschirm
more <i>LiesMich</i>	# Seitenweise Ausgabe der Datei auf den Bildschirm
	# Eingabe von h (=help) und Eingabe von q (=quit)
less <i>LiesMich</i>	# Seitenweise Ausgabe der Datei auf den Bildschirm
	# Eingabe von h (=help) und Eingabe von q (=quit)
head <i>LiesMich</i>	# Ausgabe der ersten 10 Zeilen der Datei
tail <i>LiesMich</i>	# Ausgabe der letzten 10 Zeilen der Datei
file <i>LiesMich</i>	# Bestimme Dateityp wie z.B. english text

--Ausdrucken --

lp -d scin <i>textdatei</i>	# Drucke einseitig aus
lp -d scid <i>textdatei</i>	# Drucke doppelseitig aus
lpstat -o	# zeigt Auslastung (Warteschlange) der Drucker an
cancel scid-779	# entfernt Druckjob 779 aus der Warteschlange
	# nur eigene Druckjobs können entfernt werden!

--Benutzerinformationen --

whoami	# Unter welcher Benutzerkennung bin ich eingeloggt?
hostname	# Auf welchem Rechner bin ich eingeloggt?
quota -v	# Wieviel Speicherplatz habe ich belegt?
lpquota	# Welchen Stand hat mein Druckkonto?
ps -u <i>MeinLoginName</i>	# Welche Prozesse habe ich laufen?
kill -9 <i>PID</i>	# Schieße Prozess mit der ProzessID PID ab.
yppasswd	# Ich ändere mein Paßwort für alle SCI-Rechner!

--Systemdokumentationen --

man -k <i>Stichwort</i>	# Benennt Kommandos, für die das Stichwort relevant ist!
man <i>Kommando</i>	# Gibt die Manpage für das gewählte Kommando aus.

--Vernetzte Rechner --

ssh -l <i>user Rechner</i>	# Einloggen auf einem anderen Rechner per Secure Shell!
ssh <i>user@Rechner</i>	# Einloggen auf einem anderen Rechner per Secure Shell!
sftp <i>Rechner</i>	# Dateitransfer mittels Secure FTP!
scp <i>D1 user@Rechner:/wohin</i>	# Kopieren mittels Secure Copy!

finger *user@Rechner* # Information zu einem Benutzer auf einem anderen Rechner!

--Web-Browser --

firefox #
opera #
galeon # (aus Gnome)
konqueror # (aus KDE)

--eMail-Clients --

thunderbird #
opera #
kmail # (aus KDE)
evolution # (aus Gnome)

--Text-Editoren --

xedit *oder nedit* # mit GUI
emacs *oder xemacs* #
gedit *oder kate* # (aus Gnome / aus KDE)
vim *oder nano* *oder vi* # ohne GUI

--pdf-Reader --

acroread #
evince # (aus Gnome)
okular # (aus KDE)
xpdf #

--Professionelle Textverarbeitung (Office-Pakete) --

soffice # Openoffice