

Arbeiten mit der Shell Teil 2

Linux-Kurs der Unix-AG

Klaus Denker

5. November 2012



cat

- ▶ cat gibt den Inhalt von Dateien aus
- ▶ Wichtige Optionen:
 - ▶ -A: nicht darstellbare Zeichen (Tabs, Zeilenumbrüche, etc.) werden durch Ersatzzeichen dargestellt
- ▶ Argumente: eine oder mehrere Dateien

less

- ▶ less ist ein Pager: gibt soviel Inhalt einer Datei aus wie auf den Bildschirm passt und ermöglicht es zu scrollen
- ▶ Scrollen mit den Pfeiltasten (eine Zeile) oder Bild-Auf/-Ab (eine Seite)
- ▶ Pos1 und Ende springt zum Anfang bzw. zum Ende
- ▶ Suchen mit /muster (nach unten) oder ?muster (nach oben)
- ▶ Weitersuchen mit n (gleiche Richtung) oder N (andere Richtung)
- ▶ Beenden mit q

head

- ▶ head gibt den Anfang einer Datei aus
- ▶ Wichtige Optionen:
 - ▶ -n Zahl: gibt so viele Zeilen aus wie angegeben; Standard sind 10 Zeilen

tail

- ▶ tail gibt das Ende einer Datei aus
- ▶ Wichtige Optionen:
 - ▶ -n Zahl: gibt soviele Zeilen aus wie angegeben; Standard sind 10 Zeilen
 - ▶ -f: wartet darauf, dass neue Zeilen in die Datei geschrieben werden und gibt diese dann aus

Globs

- ▶ wenn man alle Dateien in einem Verzeichnis gleichzeitig bearbeiten will (z. B. alle Kopieren oder Löschen), ist es mühsam alle einzeln aufzulisten
- ▶ Lösung: Globbs (auch Wildcards genannt)
- ▶ spezielle Zeichen, die auf mehrere Zeichen passen
- ▶ ?: ein beliebiges Zeichen (außer .)
- ▶ *: beliebig viele Zeichen (außer .)
- ▶ [...]: die gewünschten Zeichen können angegeben werden (z. B. [avx] oder [a-z])

Sonderzeichen in Dateinamen

- ▶ bestimmte Zeichen haben für die Shell eine Sonderbedeutung (wie Globs oder das Leerzeichen)
- ▶ können nicht einfach so als Dateinamen verwendet werden
- ▶ Lösung: Sonderzeichen „escapen“: \ vor das Sonderzeichen stellen
- ▶ oder: den Dateinamen in Anführungszeichen setzen
- ▶ `ls Komischer\ Date\*name` oder
`ls "Komischer Date*name"`
- ▶ trotzdem nicht zu empfehlen!

Ein-/Ausgabekanäle

- ▶ Ein- und Ausgaben von Shell Kommandos sind Text
- ▶ Ohne spezielle Parameter benutzen die meisten Kommandos die Standard-Kanäle

Nummer	Bezeichnung	Name	Gerät
0	Standard-Eingabe	stdin	Tastatur
1	Standard-Ausgabe	stdout	Konsole
2	Standard-Fehlerausgabe	stderr	Konsole

Pipes

- ▶ Pipes | leiten die Ausgabe (stdout) von einem Programm als Eingabe (stdin) in ein anderes Programm
- ▶ Grundprinzip: Einfache Befehle werden verknüpft um komplexe Dinge zu tun

Beispiele

- ▶ `ls -la | grep bash`, zeigt die Informationen aller Dateien im aktuellen Verzeichnis an, die bash enthalten
- ▶ `find . | less`, zeigt alle Dateien unterhalb des aktuellen Verzeichnisses seitenweise an

Ein-/Ausgabe in Dateien

- ▶ Die Kanäle können in Dateien umgeleitet werden
- ▶ Dazu dienen die Operatoren `<` und `>`
- ▶ `<` liest die Eingabe `stdin` aus einer Datei
- ▶ `>` bzw. `>>` schreibt die Ausgabe `stdout` in eine Datei

Ein-/Ausgabe in Dateien, Beispiele

- ▶ `ls -la > liste.txt`, erzeugt eine Datei `liste.txt` mit der Ausgabe von `ls -l` als Inhalt. Wenn die Datei schon existiert, wird der vorherige Inhalt gelöscht
- ▶ `ls -la >> liste.txt`, macht fast dasselbe. Wenn die Datei schon existiert, bleibt sie erhalten und die Ausgabe von `ls -la` wird am Ende angehängt
- ▶ `grep bash < liste.txt`, liest den Inhalt von `liste.txt` und gibt die Zeilen aus, die `bash` enthalten

Kanäle umleiten

- ▶ Die Kanäle können aufeinander umgeleitet werden
- ▶ Um Kanal 2 stderr auf Kanal 1 stdout umzuleiten wird der Operator `>&` benutzt, meist `2>&1`
- ▶ `2>&1` wird verwendet um die Fehlerausgabe von Programmen mit Pipes oder in Dateien weiterzuleiten
- ▶ Die Reihenfolge ist wichtig

Kanäle umleiten, Beispiele, tee

- ▶ `make > make.log 2>&1`, schreibt alle Ausgaben von make in die Datei `make.log`
- ▶ `make 2>&1 > make.log`, schreibt nur stdout von make in `make.log`
- ▶ Mit `tee` können Zwischenergebnisse zusätzlich in Dateien geschrieben werden: `ls -la | tee liste.txt | less`

sort

- ▶ sort sortiert Text abhängig von der Spracheinstellung LC_COLLATE des Systems
- ▶ -d (dictionary) ignoriert alles ausser Buchstaben, Ziffern und Leerzeichen
- ▶ -i (ignore) ignoriert nicht druckbare Zeichen
- ▶ -b (blank) ignoriert führende Leerzeichen

sort

- ▶ -u (unique) vereint mehrfach vorkommende Zeilen
- ▶ -n (numeric) sortiert nach dem Zahlenwert
- ▶ -k <x> (key) sortiert nach der x-ten Spalte
- ▶ -r (reverse) dreht die Sortierreihenfolge um
- ▶ Beispiel: `ls -l | sort -n -k 5 -r`, gibt die Dateien im aktuellen Verzeichnis sortiert nach der Dateigröße aus

cut

- ▶ cut dient dazu Spalten aus Texten herauszuschneiden
- ▶ -c <LISTE> schneidet die in LISTE angegebenen Zeichen heraus
- ▶ -f <LISTE> schneidet die in LISTE angegebenen Felder heraus
- ▶ -d <TRENnzeichen> zerschneidet die Eingabe in Felder an den TRENnzeichen

cut, Beispiele

- ▶ `cut -c 2-5 /etc/passwd`, gibt die Zeichen 2-5 aller Benutzer des Systems aus
- ▶ `cut -d : -f 5 /etc/passwd`, gibt die langen Namen aller Benutzer des Systems aus

- ▶ wc zählt Zeilen, Worte und Zeichen
- ▶ -c nur Zeichen werden gezählt
- ▶ -l nur Zeilen werden gezählt
- ▶ -w nur Worte werden gezählt

wc, Beispiele

- ▶ `ls -a | grep bash | wc -l`, gibt die Anzahl der Dateien mit bash im Dateinamen an
- ▶ `wc -l /etc/passwd`, gibt die Anzahl der Benutzer des Systems aus
- ▶ `cut -d : -f 4 /etc/passwd | sort -n -u | wc -l`, zählt die Anzahl der primären Benutzergruppen des Systems

Alle Befehle

Befehl	Optionen
cat	-A
less	(/muster, ?muster, n, N, q)
head	-n
tail	-n, -f
sort	-d, -i, -b, -u, -n, -k, -r
cut	-c, -f, -d
wc	-c, -l, -w

Globs

?, *, [...]