

Hochschule Harz

Vorlesung

Grundlagen der Informatik 2

vi-Editor

**Dipl.-Inf., Dipl.-Ing. (FH) M. Wilhelm
Nils Nause (Student Informatik)**

Inhaltsverzeichnis

1 Befehle 3

1.1	Unix-Befehle	3
1.1.1	Weitere Befehle:	4
1.1.2	Komprimierung.....	5

2 Der Text-Editor vi / vim.....6

2.1	Aufrufen des Editors:.....	6
2.2	Text bearbeiten	7
2.3	Löschen von Zeichen.....	7
2.4	Zwischenspeicher	7
2.5	Suchen	8
2.6	Sprung.....	8
2.7	Datei-Operationen	8
2.8	Dateirechte.....	8

1 Befehle

1.1 Unix-Befehle

cat	Anzeige des Inhaltes einer Datei
cat /proc/meminfo	Info zum System
cat /proc/cpuinfo	Info zum System
	Parameter zum cat
-n	mit Nummer am Anfang
-v	Sonderzeichen als ^X....
-e	Ausgabe des Dollarzeichens am Ende jeder Zeile (mit -v)
-t	Tabs als ^T, Formfeed als ^L
cd „blank“	cd home
cd ~	cd home
chmod u+x datei	Änderung des Execute Attribut (Starten ohne Prefix bash)
chmod u+w datei	Änderung des Schreibrechtes
chmod u-r datei	Änderung des Leserechtes
chmod go-r datei	Änderung des Leserechtes bei der Gruppe
clear	Löschen des Bildschirms
df	Anzeige des freien Festplattenspeichers
du	Anzeige des benutzten Festplattenspeichers
grep	Suchen von Zeichenfolgen in Dateien
bash script	Starten des Scriptes „script“
bash	Aufruf der Bashshell
ls	Auflisten der Dateien / Verzeichnisse
-l	Komplettliste (Longformat)
-R	Rekursiv
-a	Auflisten aller Einträge (.)
-A	Auflisten aller Einträge (ohne . und ..)
-p	Jedes Verzeichnis erhält ein /
-i	Ausgabe der i-node Nummer
-s	Ausgabe der Blockgröße (Size)
-l	Eine Spalte
-b	Ausgabe nicht druckbarer Zeichen in Oktaldarstellung
mkdir	Anlegen eines Verzeichnisses
more	Seitenweise anzeigen einer Datei etc. ls more oder cat datei1 more
ps	Anzeige des Prozessesstatus
pwd	Print working Directory (Aktuelle Verzeichnis)
rmdir	Löschen eines leeren Verzeichnisses
rm -drf [verzeichnis]	Rekursives löschen nicht leerer Verzeichnisse

<code>rpm -ivh</code>	Installation eines Paketes (Optionen <code>-i</code> install, <code>-v</code> verbose, <code>-h</code> hash
<code>sort</code>	Sortieren einer Datei
<code>wc</code>	Word Count
<code>who</code>	Anzeige der aktuellen Benutzer
<code>whoami</code>	Anzeige des aktuellen Benutzers

1.1.1 Weitere Befehle:

<code>shutdown now -r</code>	<code>-r</code> reboot after shutdown: PC wird nach dem Runterfahren neu gestartet. Der häufigste Fehler bei Verwendung des shutdown-Befehls ist, dass keine Zeitangabe gemacht wird (hier: now)
<code>ls -R > beliebig.txt</code>	Erzeugt eine Datei beliebig.txt, in der das Ergebnis der Auflistung inkl. Unterverzeichnissen zu finden ist.
<code>locate index.html</code>	Alle Dateien namens "index.html" werden aufgelistet. Es können auch Platzhalter verwendet werden, wie "*.pdf"
<code>find ordner/ -type f -name "index.htm"</code>	Alle Dateien namens "index.htm" werden aufgelistet.
<code>cp /home/user/*.* /floppy/</code>	Dateien von Festplattenverzeichnis werden auf Diskette kopiert

1.1.2 Komprimierung

Archive mit Inhalt von /etc und /home erstellen:

```
tar cvf test.tar /etc/ /home/  
tar cvf - /etc /home | gzip > test.tar.gz  
tar cvf - /etc /home | bzip2 > test.tar.bz2
```

Mit GNU tar ist folgende Kurzform möglich, um gepackte Archive zu erstellen:

```
tar czvf test.tar.gz /etc/ /home/  
tar cjvf test.tar.bz2 /etc/ /home/
```

Archive entpacken:

```
tar xvf test.tar  
gunzip < test.tar.gz | tar xvf -  
bunzip2 < test.tar.bz2 | tar xvf -  
zcat test.tar.gz | tar xvf -
```

Mit GNU tar ist folgende Kurzform möglich:

```
tar xzvf test.tar.gz  
tar xjvf test.tar.bz2
```

Archivinhalt ansehen:

```
tar tvf test.tar  
gunzip < test.tar.gz | tar tf -  
bunzip2 < test.tar.bz2 | tar tf -  
zcat test.tar.gz | tar tf -
```

Mit GNU tar ist folgende Kurzform möglich:

```
tar tzvf test.tar.gz  
tar tjvf test.tar.bz2
```

Archive von einem Verzeichnis packen und über 'pipes' in ein anderes Verzeichnis entpacken

```
tar cvf - 'directory' | ( cd 'directory/'; tar xpvf -)
```

Beispiel:

```
tar -C sourcedir cvf - . | tar -C targetdir xpvf -
```

und dies via SSH übers Netzwerk

```
ssh user@host "tar cfv - /ursprung1/ /ursprung2/" | (cd /ziel/; tar xpvf -)
```

2 Der Text-Editor vi / vim

Der Editor vi ist Bestandteil nahezu aller UNIX-Systeme. Der Texteditor vi kennt nicht den Komfort aus Windows-basierten Editoren. Er hat zwei Modi:

- Den Befehlsmodus
- Den Eingabemodus

Der Befehlsmodus ermöglicht das Ausführen von Kommandos wie Cursor verschieben, Eingabemodus ändern, Suchen, Schreiben auf Festplatte). Der vi-Editor wurde entwickelt, um auf alten Terminals zu laufen. Dementsprechend gibt es keine Menüs oder Mausbefehle, die es Ihnen ermöglicht, den Text zu ändern. Alle Befehle werden über die Tastatur eingegeben. Diese Übersicht gibt eine kurze Übersicht über die wichtigsten Befehle.

Hinweis:

Bei vi-Befehlen ist die Groß- und Kleinschreibung zu beachten.

2.1 Aufrufen des Editors:

Der Editor wird mit dem Befehl „vi dateienamen“ aufgerufen.

Beispiel:

vi test1

[illegible]

Existiert die angegebene Datei, so wird diese in den Editor geladen. Bei einer neuen Datei ist die Anzeige leer, und in der untersten Zeile ist der Dateinamen angegeben.

Am Anfang ist man immer im Befehlsmodus.

Jeder Befehl fängt mit dem „ESC“-Zeichen an.

2.2 Text bearbeiten

Befehl	Beschreibung
ESC i	Fügt Text vor dem aktuellen Zeichen ein
ESC l	Fügt Text am Zeilenanfang ein
ESC a	Fügt Text nach dem aktuellen Zeichen ein
ESC A	Fügt Text am Zeilenende hinzu
ESC o	Fügt eine Zeile unter der aktuellen Zeile ein
ESC O	Fügt eine Zeile oberhalb der aktuellen ein
ESC R	Eingabe von Text im Überschreibemodus

Nach der Eingabe eines Textes können Sie mit folgenden Befehlen die Cursorposition ändern.

2.3 Löschen von Zeichen

Befehl	Beschreibung
ESC x	Löscht ein Zeichen
ESC dw	Löscht bis Wortende
ESC db	Löscht bis zum Wortanfang
ESC dnw	Löscht die nächsten n Wörter
ESC dd	Löscht die Zeile
ESC ndd	Löscht n Zeilen
ESC d0	Löscht bis zum Zeilenanfang
ESC D	Löscht bis zum Zeilenende
ESC dg	Löscht bis zur letzten Zeile
ESC d1g	Löscht bis zur ersten Zeile

Es existiert eine Anzahl von Zwischenspeichern, die man zum Kopieren und Einfügen von Texten verwenden kann. Der Standardpuffer, der immer verfügbar ist, und die Puffer a-z können beliebig verwendet werden.

2.4 Zwischenspeicher

Befehl	Beschreibung
ESC yy	Übernimmt eine Textzeile in den Standardpuffer
ESC nyy	Übernimmt die nächsten n Zeilen in den Standardpuffer
ESC yw	Übernimmt das nächste Wort in den Standardpuffer
ESC ynw	Übernimmt die nächsten n Wörter in den Standardpuffer
ESC dd	Löscht die aktuelle Zeile und speichert diese in den Standardpuffer
ESC ndd	Löscht die nächsten n Zeilen und speichert diese in den Standardpuffer
ESC p	Wenn der Standardpuffer eine oder mehrere Zeilen enthält, werden diese unterhalb der aktuellen Zeile eingefügt
ESC P	Wenn der Standardpuffer eine oder mehrere Zeilen enthält, werden diese oberhalb der aktuellen Zeile eingefügt

2.5 Suchen

Befehl	Beschreibung
ESC /=	Sucht das erste = ab Position
ESC n	Nächste Suche
n	Nächste Suche
:%s/ab/cd/	Ersetzt die Buchstaben ab durch cd

2.6 Sprung

Befehl	Beschreibung
ESC +100	Sprung zur 100. Zeile
ESC :n	Cursor in Zeile <i>n</i>
ESC G	Cursor ans Ende des Dokumentes

2.7 Datei-Operationen

Befehl	Beschreibung
ESC zz	Speichert und beendet
ESC :w name	Schreibt die Datei
ESC :w	Schreibt die Datei
ESC :w!	Überschreibt eine Readonly-Datei
ESC :x	Speichert die Änderung und beendet
ESC :q!	Ende ohne zu speichern
ESC :set number	Zeilennummerierung anschalten
ESC:30,60w datei	Schreibt Zeile 30 bis 60 in eine neue <i>datei</i>
ESC:30,60w>>datei	Schreibt Zeile 30 bis 60 an Ende von <i>datei</i>

2.8 Dateirechte

Befehl	Beschreibung
chmod 755 *	Datei ausführbar für root, alle anderen nur r/w
chmod u-r *	Schreibschutz löschen
chmod u+w *	Schreibrechts setzen
chmod go-r *	Schreibschutz löschen
chmod go+w *	Schreibrechts setzen