



Werkzeuge der Informatik

Einführung in Unix/Linux

G. Zachmann
Clausthal University, Germany
zach@in.tu-clausthal.de



Literatur



- Bücher über Unix gibt es wie Sand am Meer ...
- Z.B.:
 - Jerry Peek, Tim O'Reilly & Mike Loukides: *UNIX Power Tools*. O'Reilly & Associates.
 - Michael Kofler: *Linux - Installation, Konfiguration, Anwendung*. Addison-Wesley.
 - Daniel J. Barrett: *Linux kurz und gut*. O'Reilly, September 2004.
- Parallel bzw. ergänzend dazu **Online-Literatur** auf der Web-Seite! (und noch viel mehr im Netz)

Weiterführender Kurs

- Vom GWDG in Göttingen:
 - Videoaufzeichnungen der letzten Veranstaltung
 - Siehe: <http://www.uni-math.gwdg.de/linuxuebung/>
 - Schon recht fortgeschritten
- Manchmal auch in unserem Rechenzentrum:
 - Termine siehe www.rz.tu-clausthal.de

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 3

Heimarbeit ... (wer hat kein Linux?)

- Irgendeine Distro kaufen oder vom RZ beziehen und installieren:
 - Z.B.: Kubuntu, Mandrake, ...
 - <http://ftp.tu-clausthal.de/ftp/linux/> oder <ftp://ftp.tu-clausthal.de/pub/linux/>
- Knoppix-CD vom RZ ziehen:
 - Keine Installation nötig
 - <ftp://ftp.tu-clausthal.de/pub/linux/knoppix/>
 - Achtung: Files sichern vor dem Abschalten!
- In beiden Fällen: ISO ziehen und CD brennen
- Cygwin
 - www.cygwin.com
 - Achtung: Execs laufen nicht auf den Linux-PCs im Pool

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 4

... und remote an der Uni

- Account am IfI:
 - Jede Gruppe bekommt in der ersten Übung einen generischen Account
 - Die Accounts werden nach dem Semester gelöscht (Daten extern sichern!)
- Remote einloggen auf kaosus

```
ssh login.in.tu-clausthal.de -l account
```
- Daten hin- und herkopieren

```
scp source-dir account@login.in.tu-clausthal.de:/home/account/...
```

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 5

Remote Login

- Ist oft sehr praktisch! Z.B., wenn ...
 - ... auf dem aktuellen Rechner bestimmte Software nicht installiert ist
 - ... man einen anderen Rechner administrieren muß
 - z.B. Web-Seiten von auswärts editieren

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 6

Wer braucht UNIX?

"Unix ist zwar ein Mainframe-Betriebssystem (und damit obsolet) hat aber noch viele Anhänger."
Windows MSCE-Training-Guide Windows 2000 Server
Kapitel 2.6.3 "Zusammenspiel mit UNIX", Verlag Markt & Technik

- Programmierer
- Web-Server
- Distributed Computing
- Wer braucht UNIX *nicht* (unbedingt) ?
 - Sekretärinnen
 - Büro- und Business-Software (Word, Buchhaltung, Powerpoint, Lagerhaltung, ...)

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 8

Vorteile von UNIX

- Extrem ausgereift (besonders die kommerziellen Unices)
- Gut durchdachtes Konzept von Anfang an
 - "Alles ist ein File"
 - "Alles ist ein Prozeß"
- Von Anfang an Multi-User- und Multi-Task-fähig
- Relativ sicher
- Flexibler
- Performanter
- Wesentlich leichter zu administrieren (wenn die Lernkurve erst einmal durchschritten ist)
- Auf allen Plattformen verfügbar

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 9

Plattformen

- Sun (Solaris)
- HP (HP-UX)
- SGI (IRIX)
- IBM (AIX)
- Mac (OS-X)
- PC (Linux)
- PDA
- Set-top boxes
- Armbanduhr
- Auto
- ...

<http://www.linuxdevices.com/>
<http://www.research.ibm.com/WearableComputing/index.html>

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 10

Die Erfinder

- Ca. 1970:
 - Haben UNIX und C erfunden!

Ken Thompson and Dennis Ritchie
Your new heroes

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 11

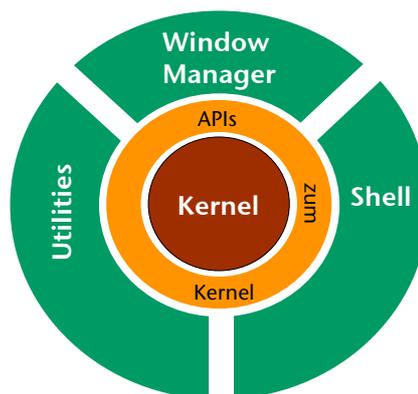
Was ist UNIX?

- Ein Betriebssystem
- Eine Sammlung von nützlichen Tools
- Eine (Computer-)Kultur



UNIX Komponenten

- Kernel: Herz des OS, managt Hardware & Programme
- Shell: eine Applikation, nimmt Kommandos entgegen und führt sie aus (CLI)
- Utilities: viele kleine (und große) Tools zur täglichen Arbeit, z.B. Files kopieren, ASCII-Texte editieren, ...



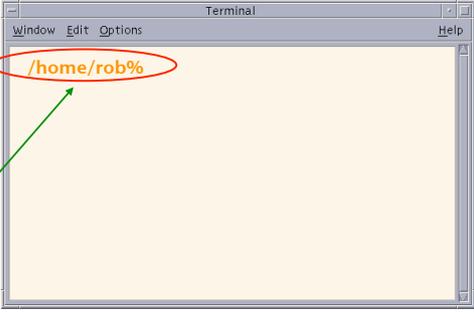
Deutsches UNIX

- Große Unsitte
 - Wegen Terminologie
- Also: **Englisch einstellen!**
 - In der tcsh: **setenv LANG en** und **setenv KDE_LANG de**
 - In der bash: **export LANG=en** und **export KDE_LANG=de**
 - Und/oder auf dem Login-Screen Englisch einstellen
 - Oder: KDE Control Center → Regional & Accessibility → Country/Region & Language

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 14

Erstes Einloggen

- Wie bekommt man eine Shell / (Terminal-)Fenster?
 - An der "Konsole" ("console")
 - Remote (ssh, rlogin, telnet)
- Login/passwd sind case-sensitive!
- Wieviele Shells kann man haben?
 - Beliebig viele ...
- Das Prompt



The image shows a terminal window titled "Terminal" with a menu bar containing "Window", "Edit", "Options", and "Help". The terminal content displays the prompt `/home/rob%`. A green arrow points from the text "Terminal-Fenster" to the title bar of the terminal window. Another green arrow points from the text "Prompt von der Shell" to the prompt text `/home/rob%`, which is also circled in red.

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 15

Das User-Interface

- Ist immer noch die Kommandozeile (CLI = command line interface)
- Für Programmierer ist CLI sehr viel effizienter!
- Lernkurve ist natürlich länger ("steiler")

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 16

Aufbau einer Kommandozeile

Kommando Optionen Parameter

```
/home/rob% ls -l data
-rw-rw-r-- 1 rob student 343 Dec 5 13:51 data
```

- Optionen (options, flags): ändern Verhalten
- Parameter: i.a. Files, auf denen Kommando operiert

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 17

Editieren der Kommandozeile

- In der Zeile:

Taste	Funktion
Tab	File- / Command-Completion
Ctrl-B / Ctrl-F	Wortweise vor / zurück springen
Ctrl-W	Voriges Wort löschen
Ctrl-U / Ctrl-K	Zeile bis zum Anfang / Ende löschen
Ctrl-A / Ctrl-E	An Ende / Anfang springen

- In der History:

Taste	Funktion
Cursor-Up / -Down	In der History rauf / runter
Ctrl-P / Ctrl-N	Match in der History nach oben / unten suchen

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 18

Kommandowiederholung

Kommando	Bedeutung
!!	Letztes Kommando wiederholen
! <i>string</i>	Kommando, das mit ' <i>string</i> ' beginnt, wiederholen
! <i>17</i>	Kommando mit Nummer 17 i.d. History wiederholen
^ <i>a</i> ^ <i>b</i>	Letztes Kommando wiederholen, dabei das erste Vorkommen von ' <i>a</i> ' durch ' <i>b</i> ' ersetzen

- History anzeigen: **history** (alias **h**)

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 19



More input? Hilfe zur Selbsthilfe

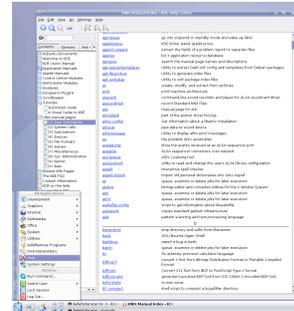


4 Arten von Informationsquellen:

- Man Pages
- HTML-Seiten
- Unter KDE: "K"-Menü → Help, dann z.B. "UNIX manual pages"

Man Pages:

- `man cmd` : zeigt Man-Page zu `cmd` an (Programm oder Funktion)
- `man -k keyword` – Alle Man-Pages nach keyword durchsuchen (nur die Titel-Zeile jeder Man-Page)
- `man -K keyword` – Alle Man-Pages nach keyword durchsuchen (komplette Seite)
- Start-Menü → Help



Format of each man page



Name	Name und 1-zeilige Beschreibung
Syntax	
Description	Ausführliche Beschreibung
Options	
Files	Liste von Files wichtig für diesen Befehl
Return values	
Diagnostics	Mögliche Fehlermeldungen und Ursachen
Bugs	Bekannte Bugs und Unzulänglichkeiten
See also	Verwandte Befehle und Infos



HTML-Seiten

- Hauptproblem: diese zu finden
- Normalerweise in `/usr/share/docs` oder `/usr/local/share/`
- Hilfsmittel: **locate**
- Dann:

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 22



Grundregeln unter UNIX

- Don't Panic!
- RTFM! ("read the f*ing manual")
- Probieren geht über studieren ...

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 23

UNIX-Konzepte

- Einige wenige Grundkonzepte:
 - Alles ist ein File (Programm, Daten, Speicher, ...)
 - Alles ist ein Prozeß (OS, laufendes Programm, Editor, Shell, ...)
 - Viele kleine Utilities, die kombiniert werden können
 - ...

The diagram illustrates the UNIX kernel as a central brown circle. Surrounding it are four green rectangular boxes, each containing a concept and its details:

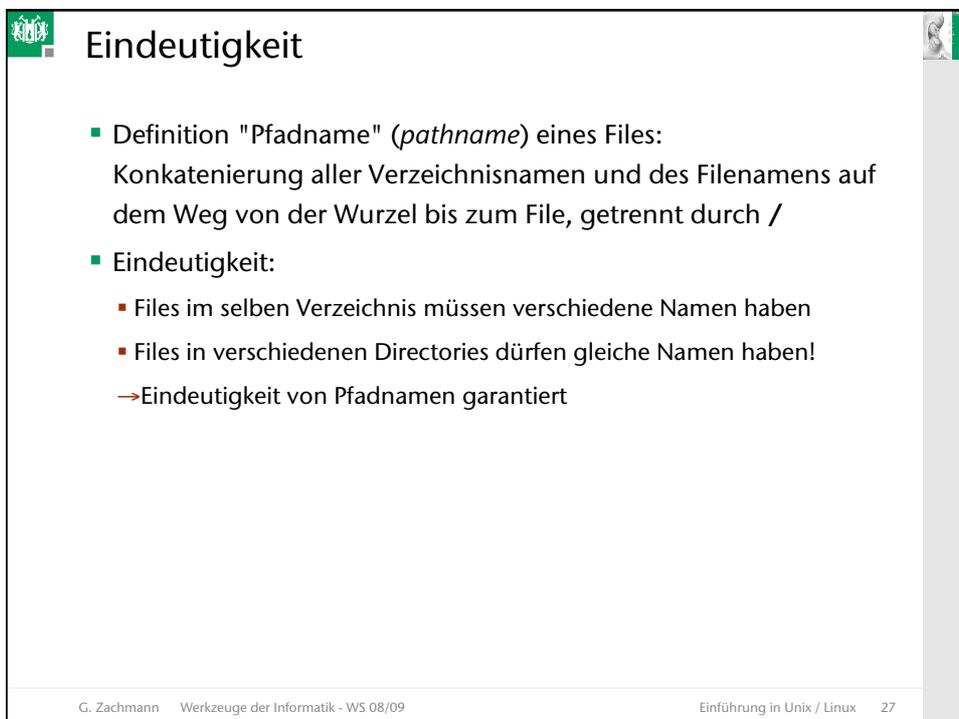
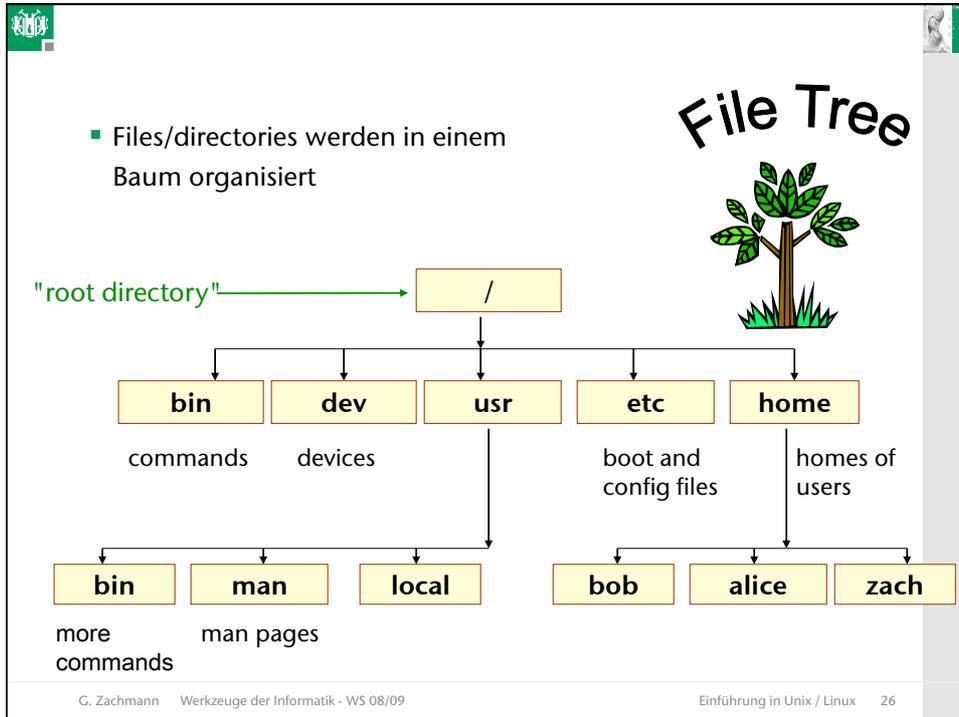
- Processes** (time sharing, protected address space)
- Interprocess comm.** (signals, pipes sockets, ...)
- Virtual memory** (swapping, paging, mapping)
- The filesystem** (files, directories, devices, pipes, namespace, ...)

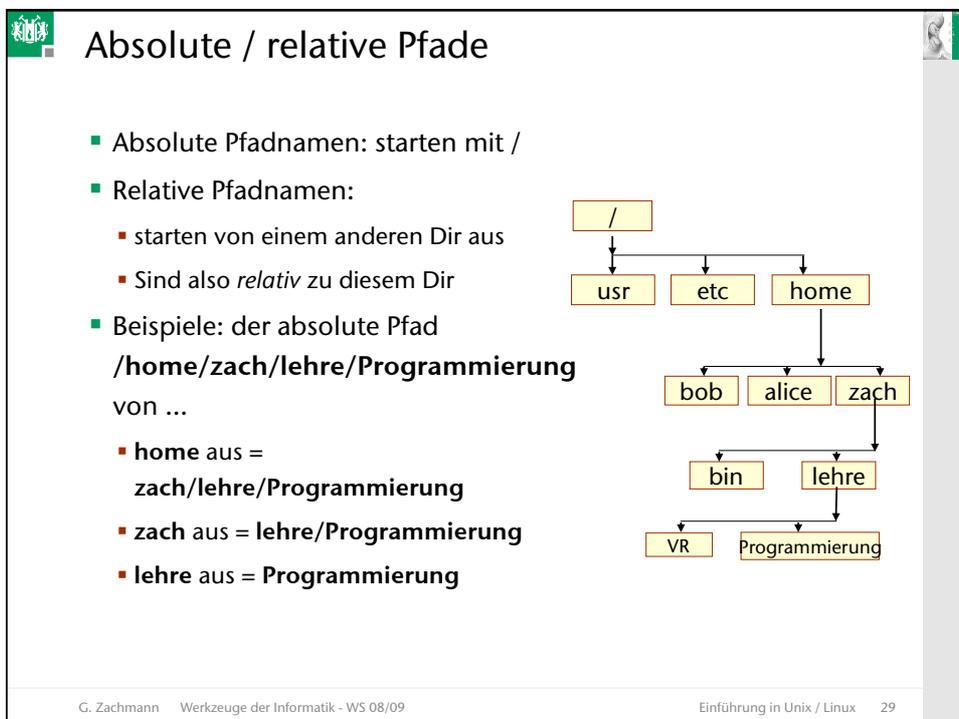
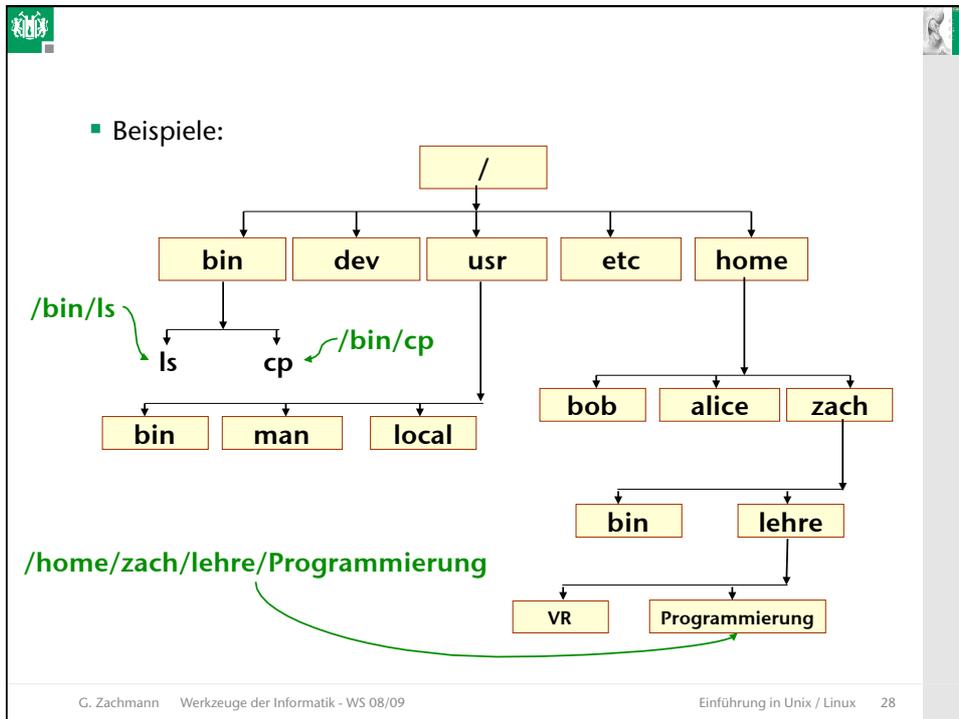
G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 24

Das Filesystem

- Directories ("Folders") und Files
- File enthält sequentielle Folge von Zeichen (Bytes)
- Interpretation ist Sache des benutzenden Programms:
 - Text, Zahlen, Programm, Speicherauszug, ...
- Jeder File hat einen Namen:
 - Case-sensitive! (UNIX allg.)
 - Länge typ. bis zu 1024
 - Können beliebige Zeichen enthalten – besser nur alphanumerische Zeichen und Underscore!
- Directory ("Verzeichnis"):
 - Enthält Name von File und Verweis darauf
 - Spezieller File

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 25





Spezielle Verzeichnisse

- '.' Bezeichnet das aktuelle Verzeichnis
 - Bsp.: `/bin/ls = /bin/./ls = /bin/././ls ...`
- '..' Bezeichnet das Vater-Verzeichnis (*parent directory*)
 - Bsp.: `/usr/bin/w = /home/./usr/bin/w = /usr/man/./bin/w ...`
- Wird besonders wichtig im Zusammenhang mit dem CWD (*current working directory*)

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 30

Kommandos: File- und Verzeichnis-Manipulation

Kommando	Funktion
<code>rm file</code>	File löschen
<code>ls [dir]</code>	Verzeichnis / File anzeigen
<code>ls -l [dir]</code>	Mehr Infos zum Verzeichnis / File anzeigen
<code>ls -a [dir]</code>	Dot-Files (.*) anzeigen
<code>cp file1 ... dir</code>	Files kopieren
<code>cp file1 file2</code>	Kopie von File1 erzeugen und File2 nennen
<code>mv file1 ... dir</code>	Files verschieben
<code>mv file1 file2</code>	File umbenennen
<code>cat file1 file2 ... > file</code>	Files aneinanderhängen (konkatenerieren)
<code>mkdir dir</code>	Neues Verzeichnis erzeugen
<code>rmdir dir</code>	Verzeichnis löschen (muß leer sein)
<code>touch file</code>	Leeren File erzeugen

- Achtung: ES GIBT KEIN RECYCLE-BIN!!! ...

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 31

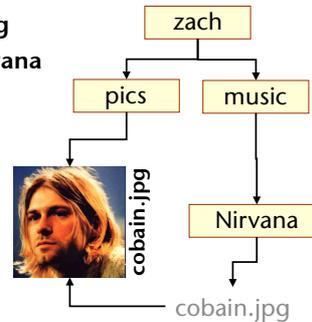
Symbolische Links

- Problem: File "gehört" genau einem Verzeichnis

- Beispiel: File `/home/zach/pics/cobain.jpg` soll auch im Dir. `/home/zach/music/Nirvana` sichtbar sein ...

- Lösung: *symbolic links (symlinks)*

- Bsp.: `music/Nirvana/cobain.jpg` ist ein Symlink nach `.././pics/cobain.jpg`



Kommando	Funktion
<code>ln -s file1 file2</code>	Erzeugt symbolischen Link von File2 nach File1 (Eselsbrücke: ln -s statt cp)
<code>rm symlink</code>	Löscht den Symbolic Link, nicht den File worauf dieser zeigt

Das Current Working Directory

- Die Shell merkt sich ein *Current Working Directory (CWD, PWD)*
 - Bei mehreren offenen Terminal-Fenstern (= Shells) merkt sich jede Shell ihr **eigenes** CWD
- Alle **relativen** Pfade werden von der Shell **relativ zu diesem CWD** interpretiert

- Für die Fortgeschrittenen:
 - Eigentlich hat jeder Prozeß sein eigenes CWD
 - (Auch die Shell ist ein ganz normaler Prozeß)
 - Die Interpretation eines relativen Pfades relativ zum CWD geschieht durch den Unix Kernel

Kommandos: Moving Around

Utility	Funktion
<code>cd dir</code>	Ins Verzeichnis <code>dir</code> wechseln (rel. oder abs. Pfad)
<code>cd -</code>	Ins vorige Verzeichnis zurück wechseln
<code>cd</code>	Ins Home wechseln
<code>pwd</code>	Aktuelles Verzeichnis (current working directory) anzeigen

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 35

Home Sweet Home

- Jeder User hat ein *Home*
 - Z.B. `/home/zach`
 - Enthält normalerweise alle Daten des Users
 - Alle Konfigurationsfiles aller Programme ("Dot-Files", z.B. `.login`) (riesiger Vorteil gegenüber Registry!)
- Beim Einloggen "startet man im Home" (d.h., CWD = `~`)
- Normalerweise auf einem Fileserver
- Ist auf jeder Maschine gleich zugreifbar
- Schreibweise: `~`

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 36

Users & Groups

- Daten eines Users:
 - Username (login, oft gleich wie email)
 - UID = ID des Usernames (`id` Kommando)
 - GID = group ID (evtl. mehrere)
 - Ein Home
 - Wird i.A. LAN-weit verwaltet
- Gruppen:
 - Jeder User gehört zu mindestens einer Gruppe
 - LAN-weit oder lokal

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 37

File Permissions

- 3 Personengruppen: Owner (=User), Group, World (Other)
- File gehört genau 1 User
- File ist assoziiert zu genau 1 Group
- Für jede der 3 Gruppen einen Satz File-Permissions:
read, write, execute

```
Terminal
Window Edit Options Help
/home/rob% ls -l file
-rw-r----- 1 rob student 343 Dec 5 13:51 file
```

File-
typ

Owner-
Permissions

Group-
Permissions

World-
Permissions

Owner

Group

G. Zachmann Werkzeuge der Informatik - WS 08/09 Einführung in Unix / Linux 38

- Filetyp-Flag:
 - Kein Permissionflag!
 - Zeigt Filetyp an:
 - - = normaler File
 - **d** = Directory
 - **l** = Symlink
 - ... einige seltenere Spezial-Flags
- Bedeutung der Permissions

Perm.	File	Directory
r (read)	Read a file	List files in ...
w (write)	Write a file	Create / move / remove a file in ...
x (execute)	Execute a file (shell script or binary)	Access a file in ...

 - Weitere, sehr praktische Flags (set-GID, set-UID, sticky, ...)

G. Zachmann Werkzeuge der Informatik - WS 08/09
Einführung in Unix / Linux 39

Permissions modifizieren

- Syntax von **chmod** ("change mode"):

chmod <level><op><perm> filename

level = String aus: u, g, o, a (user, group, other, all)
op = ein Zeichen aus +, -, = (gets, loses, equals)
perm = String aus: r, w, x, ... (read, write, execute, ...)
- Beispiele:


```
% chmod u+x foobar
% chmod u+rx,g-w foobar
% chmod g=u temp/
% chmod u=rx,g=rx,o= shared/
```

G. Zachmann Werkzeuge der Informatik - WS 08/09
Einführung in Unix / Linux 40