Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz A
Einführung, Virtualisierung, Linux

Zur Vorlesung (1)

Service / Web-Seite: http://fom.hgesser.de

- Folien und Praktikumsaufgaben
- Vorlesungs-Videos („*test, test* “)
- Probeklausur gegen Semesterende

Hilfreiche Vorkenntnisse:

- **Linux-Shell** – Benutzung der Standard-Shell *bash* unter Linux
  ➞ Bash-Crashkurs

Zur Vorlesung (2)

Praktikum:

Systemadministration unter Linux

Prüfung und Benotung

1. Lernfortschrittskontrolle (LFK)
2. Klausur über 120 Minuten

Fragen:

- direkt in der Vorlesung (Handzeichen)
- oder danach
- oder per E-Mail

Über den Dozenten

Hans-Georg Eßer

- Dipl.-Math. (RWTH Aachen, 1997)
  Dipl.-Inform. (RWTH Aachen, 2005)
- Chefredakteur Linux-Zeitschrift (seit 2000) und Autor diverser Computerbücher
- LPI-zertifiziert (LPIC-1 und LPIC-2)
- seit 2006 Dozent (HS München, FH Nürnberg, FOM): Betriebssysteme, Rechnerarchitektur, IT-Infrastrukturen, Informatik-Grundlagen, Systemprogrammierung
- Seit 2010 Doktorand (Univ. Erlangen-Nürnberg)

Einführung und Motivation

Ein praktischer Kurs

- Installation
  - Verständnis für Partitionierung, Dateisysteme, Boot-Vorgang
- Wartung
  - Nutzen von Shell-Tools (Linux)
  - Software-Installation
  - Einrichtung von Geräten
  - Benutzer-Verwaltung, Sicherheit
  - Netzwerk-Konfiguration, Netzwerk-Dienste

LPI-Zertifizierung (1)

LPI Certified Junior Level Linux Professional

- Work at the Linux command line
- Perform easy maintenance tasks: help out users, add users to a larger system, backup & restore, shutdown & reboot
- Install and configure a workstation (including X) and connect it to a LAN, or a stand-alone PC via modem to the Internet.

LPI Certified Advanced Level Linux Professional

- Administer a small to medium-sized site
- Plan, implement, maintain, keep consistent, secure, and troubleshoot a small mixed (MS, Linux) network, including a LAN server (samba), Internet Gateway (firewall, proxy, mail, news), Internet Server (webserver, FTP server)
- Supervise assistants
- Advise management on automation and purchases

Quelle: www.lpi.org

LPI-Zertifizierung (2)

LPIC–3

LPI Certified Senior Level Linux Professional

- hauptsächlich: LDAP
- dazu: verschiedene Spezialisierungen
  - LPI 302: Mixed Environments
  - LPI 303: Security
  - LPI 304: Virtualization and High Availability
  - es kommen vielleicht noch weitere

Im Rahmen dieser Vorlesung:
teilweise (!) Vorbereitung
auf die LPIC-1-Prüfungen
101 und 102

LPI-Zertifizierung (3)

- Für die Zertifizierung: Stoff aus dieser Veranstaltung reicht nicht
- insbesondere: „passives Konsumieren“ reicht nicht
- Prüfungen sind so gestaltet, dass Administratoren sie leicht bestehen können

  → üben, üben, üben :)

- Kosten: 145 € pro Prüfung (LPIC-1 = 2 Prüfungen)

Linux-Administration

- keine grafischen Werkzeuge – auch wenn es welche gibt
- also: nicht YaST & Co., sondern Kommando-zeilentools, Konfigurationsdateien, Shell-Skripte
- verstehen, was im Hintergrund abläuft

Windows-Administration

- Windows setzt auch bei der Administration überwiegend auf grafische Tools
- „GUI-Administration“ leichter (schneller) zu erlernen, bietet aber weniger Möglichkeiten, wenn etwas schief geht
- Windows ist im Rahmen dieser Vorlesung kein Thema

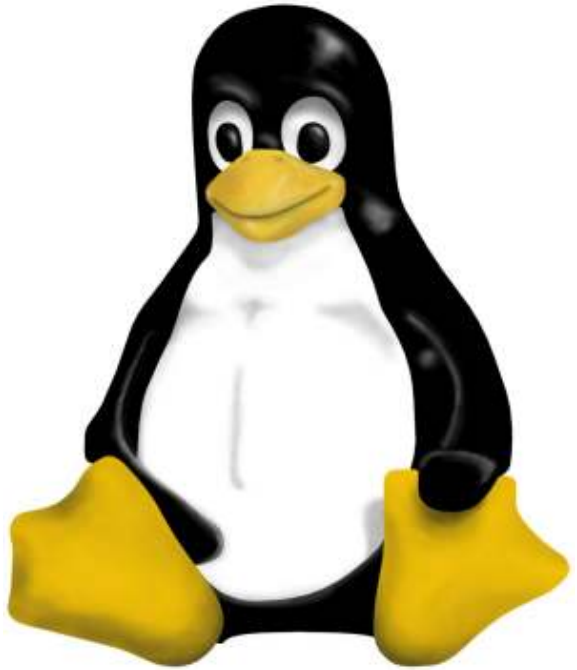
Gliederung

Gliederung

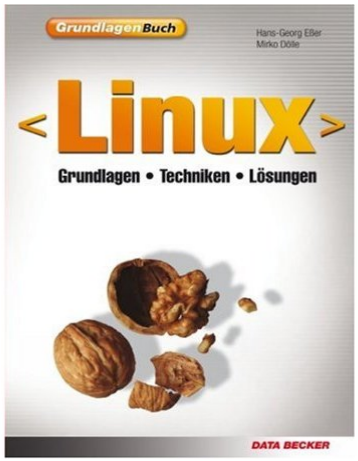
1. Einleitung
2. Virtualisierung
3. Allgemeine Grundlagen
   - Partitionen, RAID
3. Linux-Administration

Linux

- Etabliertes Standardsystem für sehr viele Plattformen (PC Desktop / Server, Embedded etc.)
- vor allem auf Servern weit verbreitet
- Image eines virtuellen Linux-PCs für VMware / VirtualBox



Literatur



Grundlagenbuch Linux
Grundlagen, Techniken, Lösungen
(Eßer, Dölle)
Data Becker, 2007
→ als PDF-Dokument im Campus-System

Linux 2011
Debian, Fedora, openSUSE, Ubuntu
(Kofler)
Addison-Wesley, 2010
49,80 €; E-Book: 39,80 €

2. Virtualisierung

Virtualisierung

- Idee: Ein Betriebssystem läuft nicht direkt auf der Hardware, sondern als Gast in einer virtuellen Maschine – unter einem Host-Betriebssystem
- Verschiedene Arten der Virtualisierung
  - Full Virtualization (z. B. VMware, VirtualBox)
  - Hypervisor ohne darunter liegendes Betriebssystem (z. B. VMware ESX Server)
  - Paravirtualisierung (z. B. Xen)

Virtualisierung: Warum? (1)

- Ungenutzte Rechenkapazitäten besser ausnutzen (vgl.: Multi-Processing)
- einheitliche Hardware: BS auf virtueller Maschine muss nicht an konkrete Hardware angepasst werden
- Server-Bereich:
  - Konsolidierung – wenige große Maschinen statt vielen kleinen
  - vereinfachtes Backup aller virt. Maschinen

Virtualisierung: Warum? (2)

- Einfachere Bereitstellung einer neuen Maschine, einfaches Duplizieren
- Snapshots erlauben Rückkehr zu funktionierendem Zustand einer Maschine
- Sicherheit: Voreinander zu schützende Anwendungen besser in separaten VMs als auf einem logischen Rechner laufen lassen
- Für Entwickler: Test auf verschiedenen Plattformen, ohne dafür jeweils 1 PC zu benötigen
- Legacy-Support (alte BS)

Virtualisierung: Warum? (3)

Warum im Rahmen dieser Vorlesung?

- einheitliche Hardware-Umgebung für ein zu testendes Linux-System (Debian)
- Systemverwaltung ohne das Risiko, Ihre Notebooks „kaputt“ zu konfigurieren

Virtualisierung: Konzepte

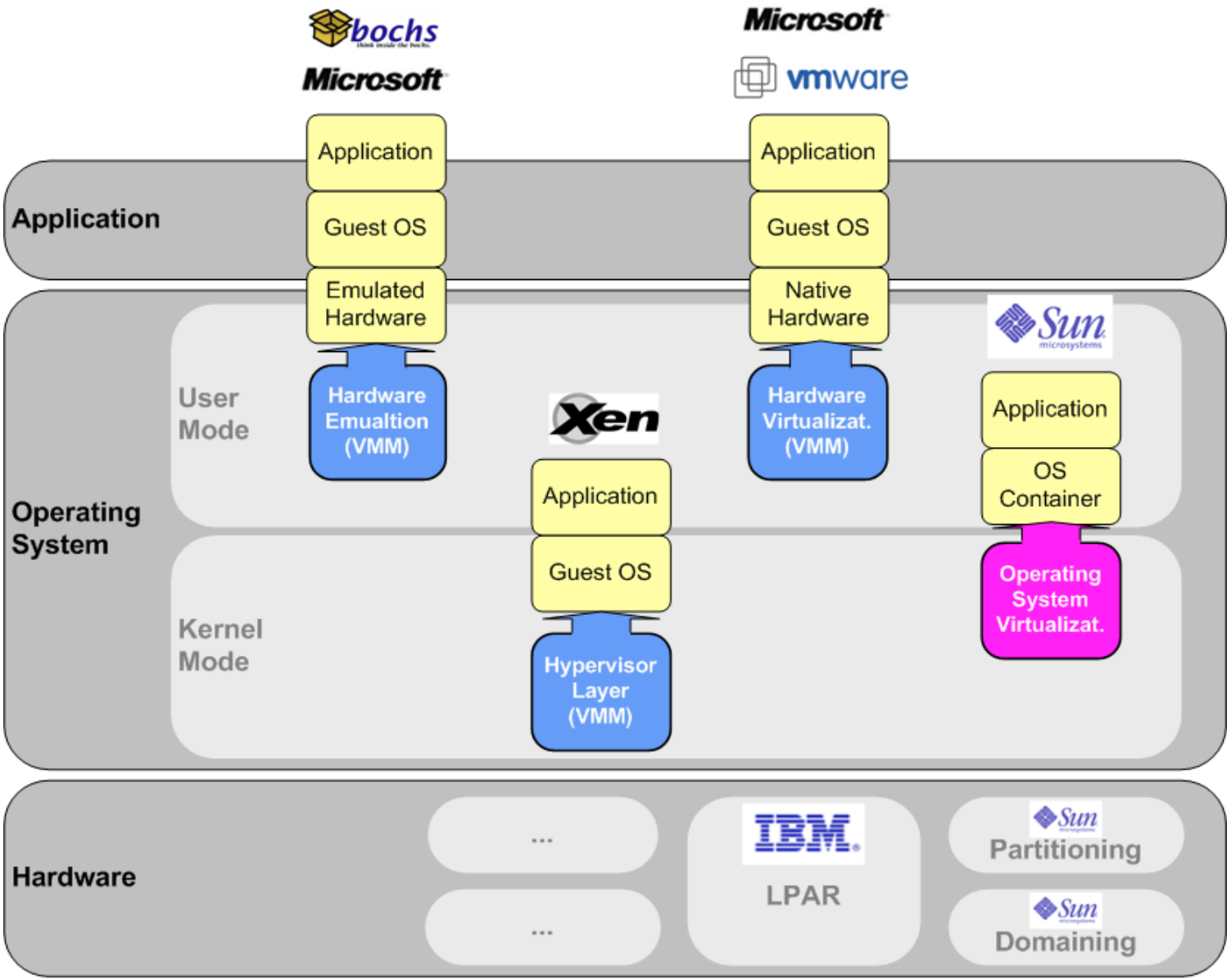


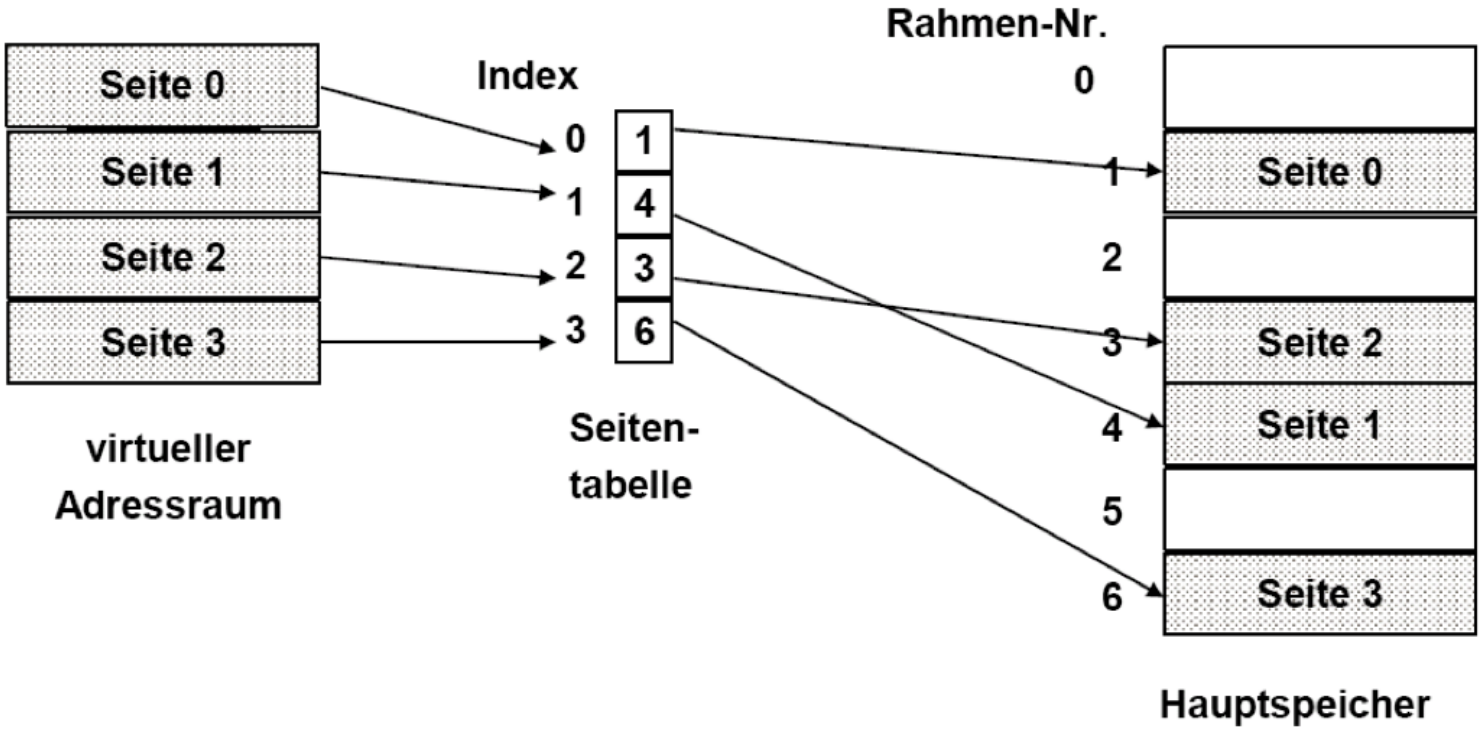
Bild: Daniel Hirschbach, Lizenz: CC-by-sa 2.0/de
(Quelle: Wikipedia, Virtualisierung)

Virtualisierung: Probleme

- Einfache User-Mode-Instruktionen (arithmetische und logische Instruktionen, Speicherzugriff bei virtuellem Speicher/Paging) sind direkt ausführbar
- Andere Instruktionen müssen abgefangen werden (im User Mode: Software-Interrupt, OS Call; im System Mode: Hardware-Zugriffe)
- „virtualisierbare“ vs. „nicht-virtualisierbare“ Instruktionen

Paging in der VM (1)

- Paging, bekannt aus BS-Theorie (3. Sem.):
  - Für jeden Prozess verwaltet das BS eine Seitentabelle, die virtuelle Seiten auf physikalische Seitenrahmen abbildet



  - Beim Prozesswechsel wird das Seitentabellenregister (Page Table Pointer) in der CPU aktualisiert

Paging in der VM (2)

- Was tun bei einem virtualisiertem BS? (1/3)
  - Annahme: Host und Gast arbeiten beide mit Paging und verwenden dieselbe Seitengröße
  - Speicherzugriff im User-Mode (aus einer Anwendung in der virtuellen Maschine) ist eine nicht-privilegierte Operation – es muss also eine Seitentabelle geben, die Seiten auf die echten Seitenrahmen (auf dem Hostsystem) abbildet
  - Gastsystem darf diese Tabelle nicht selbst erzeugen (liegt außerhalb der VM) und nicht selbst das echte CPU-Register verändern

Paging in der VM (3)

- Was tun bei einem virtualisiertem BS? (2/3)
  - Eigentliche zweistufige Übersetzung nötig:
    - Zuordnung
      virtuelle Seite (Gast) → physik. Frame (Gast)
      (durch „virtuelle“ MMU, mit Seitentabelle im Gast)
    - Zuordnung
      physik. Frame (Gast) → physik. Frame (Host)

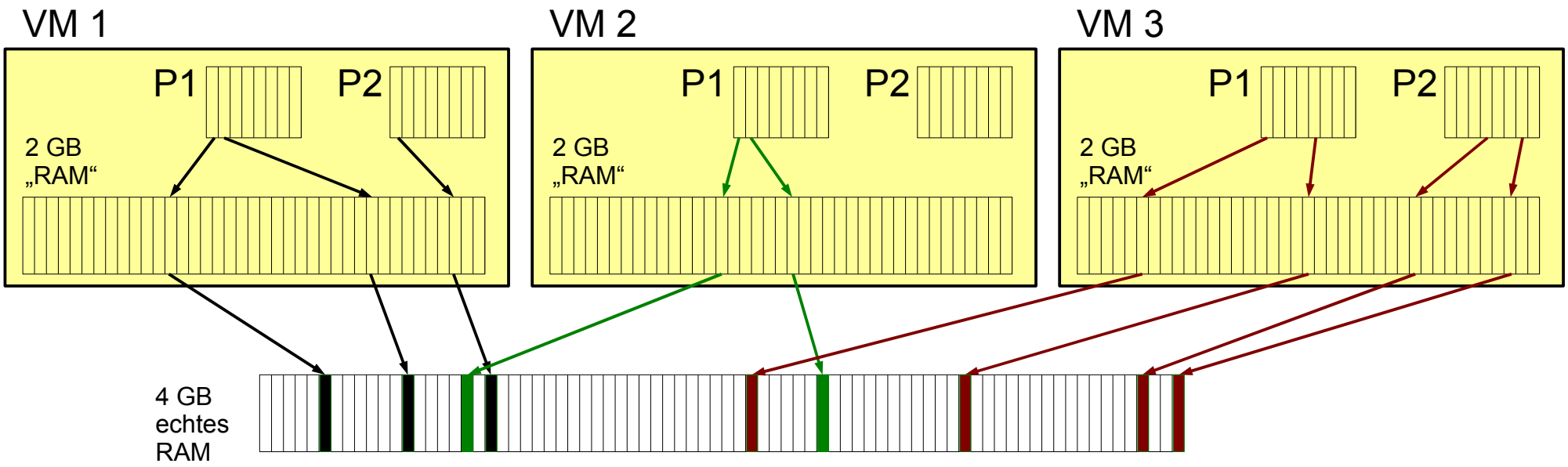
  VMM müsste also jeden Speicherzugriff abfangen und „umbiegen“, d. h. in Software die Funktion der MMU nachbilden

Paging in der VM (4)

- Was tun bei einem virtualisiertem BS? (3/3)
  - Lösung: zwei Seitentabellen – eine virtuelle Page Table (im Gast) und eine zugehörige „Shadow Page Table“ (im Host)
  - Wenn das Gast-BS beim Context Switch die Seitentabelle austauschen will (also das PT-Register schreibt), wird dies abgefangen; dann wird im PT-Register die Adresse der passenden Shadow Page Table eingetragen

Paging in der VM (5)

- Beispiel:
  - Host: 4 GByte RAM
  - 3 Gäste, jeweils 2 GByte virtuelles RAM
  - je zwei Prozesse pro Gastsystem



Paging in der VM (6)

Ablauf beim Context Switch im Gast-BS:

- Gast-BS führt Scheduler aus und wählt einen neuen Prozess aus
- Gast-BS setzt PT-Register auf neue Seitentabelle für diesen Prozess
- VMM fängt diesen Aufruf ab und sucht passende Shadow Page Table
- VMM setzt PT-Register und gibt Kontrolle an Gast-BS zurück
- Gast-BS vollendet Context Switch und gibt Kontrolle an neuen Prozess

Emulation vs. Virtualisierung

- Emulation (von Hardware) ist ein anderes Konzept und nicht mit Virtualisierung zu verwechseln
- Emulator ahmt Hardware inkl. CPU vollständig nach
- Instruktionen werden also im Emulator nicht ausgeführt, sondern „interpretiert“
- Emulatoren können auch Hardware mit abweichenden CPUs emulieren (z. B. C64-Emu)

Emulation über Bibliotheken

- Beispiel WINE („WINE Is Not an Emulator“)
- WINE ersetzt auf Linux-Systemen den Programm-Loader und diverse Bibliotheken (DLLs), die Software unter Windows erwartet.
- WINE ist also kein Hardware-Emulator und stellt auch keine virtuelle Maschine bereit.
- Windows-Programme laufen dank WINE fast „nativ“ auf dem Linux-System.

VirtualBox und Debian Linux

Praxisteil

- Installation VirtualBox auf Ihrem Notebook / Netbook
- Anlegen einer virtuellen Maschine (5 GByte Plattenplatz, 512 MByte RAM)
- Installation von Debian Linux 5 in der virtuellen Maschine
- erste Experimente mit dem installierten Linux

Einführung in die Linux-Shell

Praxisteil

- Arbeiten mit der Shell
- Verzeichnisnavigation, -Listings
- Dateien kopieren, umbenennen, verschieben
- Verzeichnisse erstellen, löschen etc.
- Dateien öffnen
- Der Editor „vi“

Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz B
Dateien und Verzeichnisse, Editor vi

Einführung in die Linux-Shell

Praxisteil

- Arbeiten mit der Shell
- Verzeichnisnavigation, -Listings
- Dateien kopieren, umbenennen, verschieben
- Verzeichnisse erstellen, löschen etc.
- Dateien öffnen
- Der Editor „vi“

Shell-Prompt (1)

- Shell zeigt durch **Prompt** an, dass sie bereit ist, einen Befehl entgegen zu nehmen
- Prompts können verschieden aussehen:
  - … $ _
    … > _ : Anwender-Prompt, nicht-privilegiert
  - … # _ : Root-Prompt, für den Administrator

Shell-Prompt (2)

- Vor dem `$`, `>`, `#` meist Hinweise auf Benutzer, Rechner, Arbeitsverzeichnis

```
[esser@macbookpro:BS-Praxis]$
root@quad:~#
```

  - `esser`, `root`: Benutzername; individuell
  - `macbookpro`, `quad`: Rechnername
  - `BS-Praxis`, `~`: Arbeitsverzeichnis, je nach Prompt-Einstellung auch in voller Länge (z. B. /home/esser/Daten/FOM/SS2011/BS-Praxis)
  - `~` = „Home-Verzeichnis“ des Benutzers

Befehlseingabe (1)

- Am Prompt Befehl eingeben und mit [Eingabe] abschicken
- Shell versucht, (in der Regel) erstes Wort als Kommandoname zu interpretieren:
  - Alias? (→ später)
  - Shell-interne Funktion? (→ später)
  - eingebautes Shell-Kommando? (z. B. `cd`)
  - externes Programm? (Suche in Pfad)

Befehlseingabe (2)

- Beispiel: Aktuelles **Arbeitsverzeichnis** anzeigen (`pwd` = **p**rint **w**orking **d**irectory)

```
[esser@quad:~]$ pwd
/home/esser
[esser@quad:~]$ _
```

- Nach Abarbeiten des Befehls (oft: mit einer „Antwort“) erscheint wieder der Prompt – Shell ist bereit für nächstes Kommando

Befehlseingabe (3)

- Mehrere Befehle auf einmal abschicken: mit Semikolon `;` voneinander trennen

```
[esser@quad:~]$ pwd; pwd
/home/esser
/home/esser
[esser@quad:~]$ _
```

Befehlseingabe (4)

- **Inhaltsverzeichnis** anzeigen: `ls` (**list**)
- bezieht sich immer auf das aktuelle Arbeits-verzeichnis (Alternative: Ort als Parameter angeben)

```
[esser@quad:~]$ ls
bahn-2011-02-22.pdf    bh-win-04-kret.pdf
buch_kap08.pdf         bv-anleitung.pdf
bz2.pdf
[esser@quad:~]$ ls /tmp
cvcd  kde-esser  ksocket-esser  orbit-esser
ssh-vrUNLb1418  virt_1111
[esser@quad:~]$ _
```

Befehlseingabe (5)

- Inhalt mit mehr Informationen: `ls -l`

```
[esser@quad:~]$ ls -l
-rw------- 1 esser users  29525 Feb 21  2011 bahn-2011-02-22.pdf
-rw-r--r-- 1 esser users 745520 Apr 10  2004 bh-win-04-kret.pdf
-rw-r--r-- 1 esser users 856657 Oct 21  2005 buch_kap08.pdf
-rw-r--r-- 1 esser esser 738570 Mar 17 20:29 bv-anleitung.pdf
-rw-r--r-- 1 esser users 123032 Sep 22  2003 bz2.pdf
[esser@quad:~]$ _
```

- Ausgabe enthält zusätzlich:
  - Zugriffsrechte (`-rw-r--r--` etc.) → später
  - Dateibesitzer und Gruppe (`esser`, `users`) → später
  - Größe und Datum/Zeit der letzten Änderung

Befehlseingabe (6)

- Leere Datei erzeugen (für Experimente): `touch`

```
[esser@quad:~]$ touch Testdatei
[esser@quad:~]$ ls -l Testdatei
-rw-r--r-- 1 esser esser 0 Apr  7 13:58 Testdatei
[esser@quad:~]$ _
```

- Datei hat Größe 0

Befehlseingabe (7)

- Fehlermeldungen: Unbekanntes Kommando

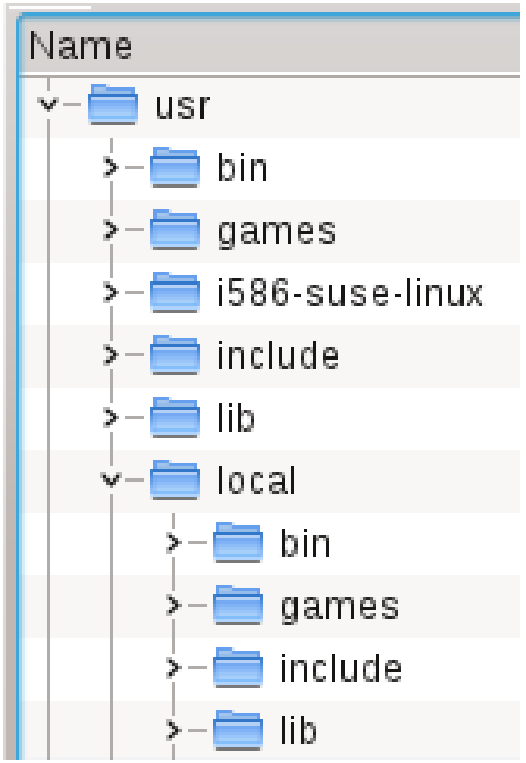
```
[esser@quad:~]$ fom
No command 'fom' found, did you mean:
 Command 'fim' from package 'fim' (universe)
 Command 'gom' from package 'gom' (universe)
 Command 'fop' from package 'fop' (universe)
 Command 'fdm' from package 'fdm' (universe)
 Command 'fpm' from package 'fpm2' (universe)
 [...]
fom: command not found
[esser@quad:~]$ _
```

- Meldung kann auch deutschsprachig sein

Dateiverwaltung (1)

Grundlagen (1)

- Linux kennt keine „Laufwerksbuchstaben“ (`C:`, `D:` etc.)
- Wurzelverzeichnis heißt `/`
- Pfadtrenner: auch `/` – d. h.: `/usr/local/bin` ist das Verzeichnis `bin` im Verzeichnis `local` im Verzeichnis `usr`. (wie bei Webadressen)



Dateiverwaltung (2)

Grundlagen (2)

- Weitere Datenträger erscheinen in Unterordnern
  - Beispiel: DVD mit Dateien zum Kurs hat Volume-Name `BS-ESSER`
  - Datei `test.txt` auf oberster DVD-Verzeichnisebene ist als `/media/BS-ESSER/test.txt` erreichbar (Windows: `e:\test.txt`)
  - Datei `Software/index.html` der DVD entsprechend als `/media/BS-ESSER/Software/index.html` (Windows: `e:\Software\index.html`)

Dateiverwaltung (3)

Grundlagen (3)

- Für private Nutzerdaten hat jeder Anwender ein eigenes **Home-Verzeichnis**, das i. d. R. unterhalb von `/home` liegt, z. B. `/home/esser`.
- Die Tilde `~` ist immer eine Abkürzung für das Home-Verzeichnis
  - funktioniert auch in zusammengesetzen Pfaden
  - `~/Daten/brief.txt` statt `/home/esser/Daten/brief.txt`

Dateiverwaltung (4)

Grundlagen (4)

- Ausnahme: Das Home-Verzeichnis des Systemadministrators `root` ist nicht `/home/root`, sondern `/root`
- Der Trick mit der Tilde `~` funktioniert aber auch für `root`
- Warum? `/home` könnte auf einer separaten Partition liegen und bei einem Fehlstart nicht verfügbar sein

Dateiverwaltung (5)

Grundlagen (5)

- Zwei Spezialverzeichnisse in jedem Ordner
  - `..` ist das Verzeichnis eine Ebene tiefer (von `/usr/local/bin` aus ist `..` also `/usr/local`)
  - `.` ist das aktuelle Verzeichnis
- Pfade kann man **absolut** und **relativ** zusammen bauen
  - absoluter Pfad beginnt mit `/`
  - relativer Pfad nicht; er gilt immer ab dem aktuellen Arbeitsverzeichnis

Dateiverwaltung (6)

Verzeichnisnavigation

- Kommando `cd` (**c**hange **d**irectory) wechselt in ein anderes Verzeichnis
- Zielverzeichnis als Argument von `cd` angeben – wahlweis mit relativem oder absolutem Pfad

```
[esser@quad:~]$ pwd
/home/esser
[esser@quad:~]$ cd /home ; pwd
/home
[esser@quad:home]$ cd .. ; pwd
/
[esser@quad:/]$ _
```

Dateiverwaltung (7)

Datei kopieren

- Kommando `cp` (**cop**y) kopiert eine Datei
- Reihenfolge: `cp` *`Original Kopie`*

```
[esser@quad:tmp]$ ls -l
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 test.dat
[esser@quad:tmp]$ cp test.dat kopie.dat
[esser@quad:tmp]$ ls -l
-rw-r--r--  1 esser  wheel  1501 Apr  8 12:17 kopie.dat
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 test.dat
[esser@quad:tmp]$ _
```

! Kopie erhält aktuelles Datum/Zeit

Dateiverwaltung (8)

Datei umbenennen

- Kommando `mv` (**mv**e) benennt eine Datei um
- Reihenfolge: `mv` *`AltName NeuName`*

```
[esser@quad:tmp]$ ls -l
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 test.dat
[esser@quad:tmp]$ mv test.dat neu.dat
[esser@quad:tmp]$ ls -l
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 neu.dat
[esser@quad:tmp]$ _
```

! Umbennen ändert Datum/Zeit nicht

Dateiverwaltung (9)

Datei verschieben

- Kommando `mv` (move) verschiebt eine Datei
- Reihenfolge: `mv` *`AltName NeuerOrdner/`*

```
[esser@quad:tmp]$ ls -l
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 test.dat
[esser@quad:tmp]$ mv test.dat /home/esser/
[esser@quad:tmp]$ ls -l
[esser@quad:tmp]$ ls -l /home/esser/
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 test.dat
   [...]
[esser@quad:tmp]$ _
```

! Verschieben ändert Datum/Zeit nicht

Dateiverwaltung (10)

Datei löschen

- Kommando `rm` (**rem**ove) löscht eine Datei

```
[esser@quad:tmp]$ ls -l
-rw-r--r--  1 esser  wheel  1501 Apr  5 11:37 test.dat
[esser@quad:tmp]$ rm test.dat
[esser@quad:tmp]$ ls -l
[esser@quad:tmp]$ _
```

Dateiverwaltung (11)

Mehrere Dateien

- Einige Befehle akzeptieren mehrere Argumente, z. B.
  - `mv` (beim Verschieben in anderen Ordner)
  - `rm`
- Beispiele:

```
[esser@quad:tmp]$ mv datei1.txt datei2.txt Ordner/
[esser@quad:tmp]$ rm datei3.txt datei4.txt datei5.txt
[esser@quad:tmp]$ _
```

Dateiverwaltung (12)

Wildcards (*, ?)

- Bei Befehlen, die mehrere Argumente akzeptieren, können Sie auch Wildcards verwenden:
  - `*` steht für beliebig viele (auch 0) beliebige Zeichen
  - `?` steht für genau ein beliebiges Zeichen
- Beispiele:

```
[esser@quad:~]$ ls -l ?????.pdf
-rw-r--r--  1 esser  staff   79737 Apr  2 01:18 RegA4.pdf
-rw-r--r--  1 esser  staff  132246 Apr  4 18:02 paper.pdf
[esser@quad:~]$ rm /tmp/*
[esser@quad:~]$ _
```

Befehle testen

- Löschbefehl mit Wildcards zu gewagt?
  → vorher mit `echo` testen:

```
[esser@quad:Downloads]$ echo rm *.zip
rm Logo_a5_tif.zip Uebung1.zip c32dwenu.zip
ct.90.01.200-209.zip ct.90.12.130-141.zip
ct.91.02.285-293.zip ct.91.12.024-025-1.zip
ct.91.12.024-025.zip ct.92.08.052-061.zip
ix.94.03.010-011.zip ix.94.07.068-071.zip
[esser@quad:Downloads]$ rm *.zip
[esser@quad:Downloads]$ _
```

Wildcard-Auflösung

- Das letzte Beispiel verrät etwas über das Auflösen der Wildcards
  - Wenn Sie `rm *.zip` eingeben, startet die Shell *nicht* `rm` mit dem Argument „`*.zip`“
  - Die Shell sucht im aktuellen Verzeichnis alle passenden Dateien und macht jeden Dateinamen zu einem Argument für den `rm`-Aufruf.
  - Es wird also
    `rm Logo_a5_tif.zip Uebung1.zip c32dwenu.zip ct.90.01.200-209.zip ...`
    aufgerufen.

Verzeichnisse (1)

Mit Verzeichnissen können Sie ähnliche Dinge tun wie mit Dateien

- Verzeichnis erstellen
- (leeres!) Verzeichnis löschen
- Verzeichnis umbenennen oder verschieben
- Verzeichnis rekursiv (mit allen enthaltenen Dateien und Unterordnern) löschen

Verzeichnisse (2)

Verzeichnis erstellen

- Kommando `mkdir` (**make dir**ectory) erzeugt ein neues (leeres) Unterverzeichnis

```
[esser@quad:tmp]$ ls -l
[esser@quad:tmp]$ mkdir unter
[esser@quad:tmp]$ ls -l
drwxr-xr-x  2 esser  wheel  68 Apr  8 14:28 unter
[esser@quad:tmp]$ cd unter
[esser@quad:unter]$ ls -l
[esser@quad:unter]$ cd ..
[esser@quad:tmp]$ _
```

! Kurzform `md` für `mkdir` nicht immer vorhanden → vermeiden

Verzeichnisse (3)

Verzeichnis löschen

- Kommando `rmdir` (**rem**ove **dir**ectory) löscht ein leeres (!) Unterverzeichnis

```
[esser@quad:tmp]$ touch unter/datei
[esser@quad:tmp]$ rmdir unter
rmdir: unter: Verzeichnis nicht leer
[esser@quad:tmp]$ rm unter/datei
[esser@quad:tmp]$ rmdir unter
[esser@quad:tmp]$ _
```

! Kurzform `rd` für `rmdir` nicht immer vorhanden → vermeiden

Verzeichnisse (4)

Verzeichnis umbenennen / verschieben

- funktioniert wie das Umbenennen / Verschieben von Dateien
- gleicher Befehl: `mv`, wieder zwei Varianten:
  - `mv Verzeichnis NeuerName`
  - `mv Verzeichnis AndererOrdner/`

Verzeichnisse (5)

Verzeichnis rekursiv löschen

- Kommando `rm` (**rem**ove) hat eine Option `-r` zum **r**ekursiven Löschen:

```
[esser@quad:tmp]$ mkdir a; mkdir a/b; mkdir a/b/c
[esser@quad:tmp]$ touch a/b/c/datei
[esser@quad:tmp]$ rmdir a
rmdir: a: Verzeichnis nicht leer
[esser@quad:tmp]$ rm -r a
[esser@quad:tmp]$ _
```

! Vorsicht beim rekursiven Löschen: „Was weg ist, ist weg“

~~Undelete~~

- Undelete = Löschen rückgängig machen
  - gibt es unter Linux nicht
  - Wiederherstellung von gelöschten Dateien mit Profi-Tools möglich, wenn Computer nach dem Löschen sofort ausgeschaltet wurde
  - solche Tools stellen aber sehr viele Dateien wieder her → enormer Aufwand, anschließend die gesuchte Datei zu finden; u. a. sind die Dateinamen dauerhaft verloren

→ vor `rm -r` … mehrfach prüfen …

Optionen und Argumente

- **Argumente:** z. B. Dateinamen; beziehen sich oft auf Objekte, die manipuliert werden sollen
- **Optionen:** verändern das Verhalten eines Befehls
  - bei den meisten Befehlen zwei Varianten:
  - kurze Optionen: `-a`, `-b`, `-c`, ...
    → lassen sich kombinieren: `-abc` = `-a -b -c`
  - lange Optionen: `--ignore`, `--force`, `--all` etc.
  - Beispiel: `-r` bei `rm`

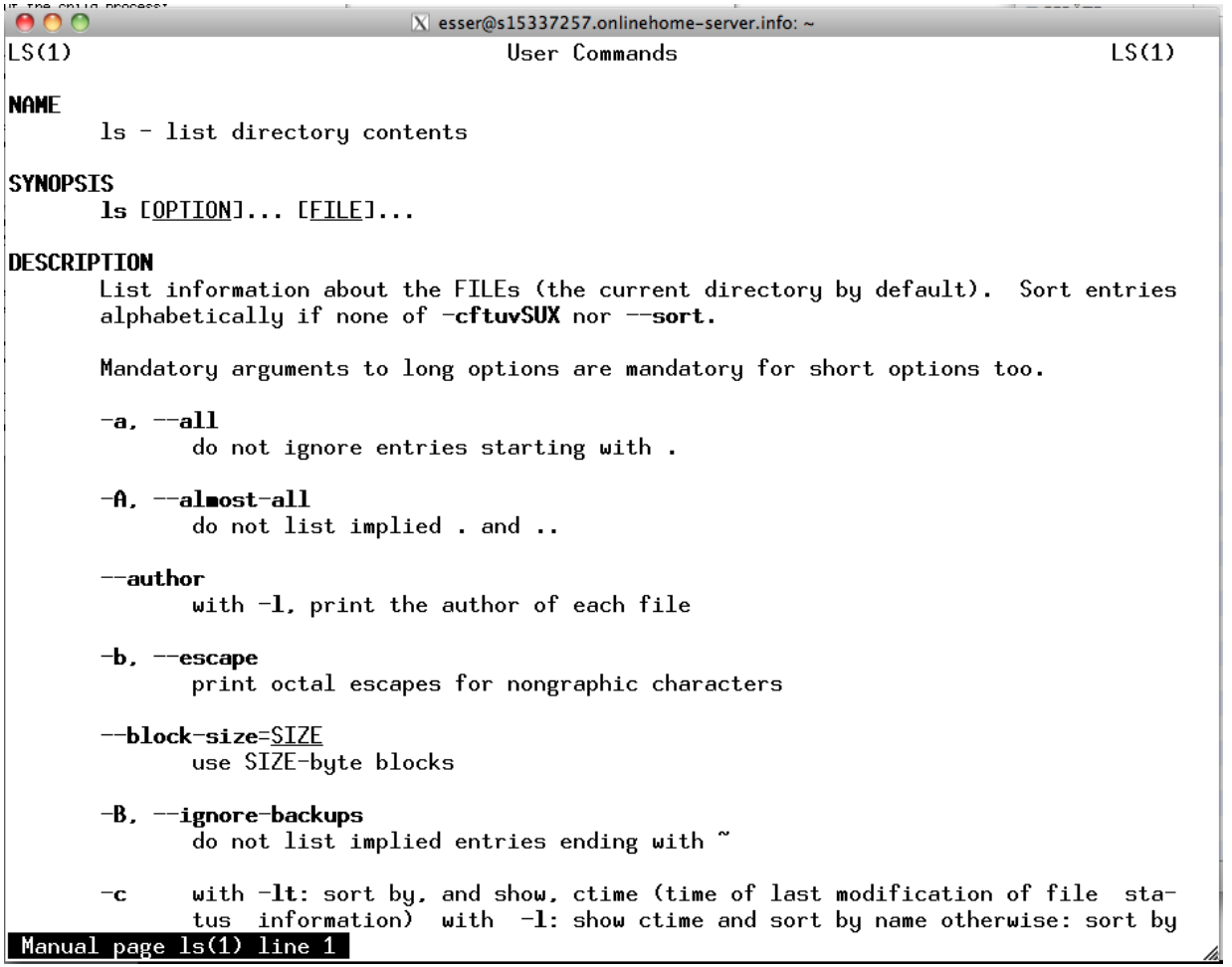
Hilfe: Handbuch

- Zu den meisten Kommandos gibt es eine sog. Manpage, die Sie über

  `man kommando`

  abrufen
- Beispiel:

  `man ls`



Der Editor vi (1)

- Standard-Editor auf allen Unix-Systemen (und damit auch Linux): `vi` (**vi**sual editor)
- gewöhnungsbedürftige Bedienung
- zwei Betriebsarten
  - **Befehlsmodus** (nach Start aktiviert; Normalmodus)
  - **Bearbeitungsmodus**
- `vi` aus Versehen gestartet? Verlassen ohne Speichern von Änderungen mit

  [Esc] `:q!`

Der Editor vi (2)

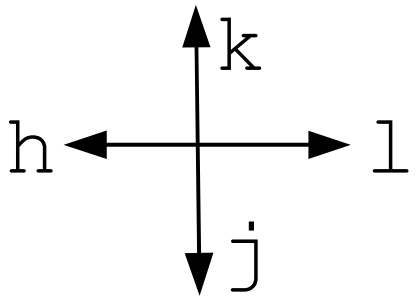
- Warum Umgang mit `vi` lernen?
  - auf jedem – noch so minimalistischen – Unix-System ist ein `vi` installiert (kleines Programm):

```
[esser@quad:~]$ ls -l /usr/bin/vi /usr/bin/emacs
-rwxr-xr-x 1 root root 5502096 Nov  9  2008 /usr/bin/emacs
-rwxr-xr-x 1 root root  630340 Oct 17  2008 /usr/bin/vi
```

  - läuft im Terminal → hilfreich bei Remote-Zugriff
  - Bei Problemen (Plattenfehler, nicht alle Datei-systeme verfügbar) sind andere Editoren evtl. nicht erreichbar, `vi` vielleicht doch → gilt leider nicht mehr für aktuelle Linux-Versionen
  - Thema ist LPI-prüfungsrelevant

Der Editor vi (3)

- Wechseln in den Bearbeitungsmodus: `i`, `I`, `a`, `A`
  - `i`: Text vor dem Cursor einfügen
  - `a`: Text nach dem Cursor einfügen
  - `I`: Text am Zeilenanfang einfügen
  - `A`: Text am Zeilenende einfügen
- Bearbeitungsmodus verlassen: [Esc]
- Navigieren im Text:
  Cursortasten oder:



Der Editor vi (4)

- Zeichen / Text löschen:
  - im Bearbeitungsmodus mit [Rückschritt] und [Entf], wie aus anderen Editoren bekannt
  - im Befehlsmodus mehrere Möglichkeiten:
  - `x` löscht Zeichen unter Cursor
  - `X` löscht Zeichen links von Cursor
  - `dw` löscht ab Cursor-Position bis Anfang des nächstens Wortes
  - `dd` löscht aktuelle Zeile
  - vorab Zahl: Mehrfachausführung (`15dd`: 15 Zeilen)

Der Editor vi (5)

- Speichern und beenden
  - Immer zuerst in den Befehlsmodus
    → im Zweifelsfall einmal [Esc] drücken
  - Speichern: `:w`
  - Speichern (erzwingen): `:w!`
  - Beenden (klappt nur, wenn Text seit letztem Speichern nicht verändert wurde): `:q`
  - Beenden erzwingen (ohne speichern): `:q!`
  - Speichern und beenden: `:wq` (oder: `ZZ` ohne „:“)

Der Editor vi (6)

- Suche im Text
  - Vorwärtssuche: `/` und Suchbegriff, dann [Eingabe]
  - Sprung zum nächsten Treffer: `n` (next)
  - Rückwärtssuche: `?` und Suchbegriff, dann [Eingabe]
  - Sprung zum nächsten Treffer: `n`
  - Wechsel zwischen Vorwärts- und Rückwärtssuche: einfach `/` bzw. `?` , dann Eingabe und mit `n` weiter (in neuer Richtung) suchen

Der Editor vi (7)

- Rückgängig machen / wiederherstellen
  - Letzte Änderung rückgängig machen: `u` (undo)
  - geht auch mehrfach: `u`, `u`, `u`, …
  - … und mit Mehrfachausführung: `3u` macht die letzten drei Änderungen rückgängig
  - Einen Undo-Schritt aufheben: [Strg]+`r` `:redo`
  - mehrfaches Redo: z. B. `3` [Strg]+`r`

Der Editor vi (8)

- Copy & Paste: Kopieren …
  - `yw` (ab Cursorposition bis Wortende)
  - `y$` (ab Cursorposition bis Zeilenende)
  - `yy` (ganze Zeile)
  - `3yy` (drei Zeilen ab der aktuellen)
- … und Einfügen
  - `P` (fügt Inhalt des Puffers an Cursorposition ein)
- Cut & Paste
  - Löschen mit `dd`, `dw` etc.; dann einfügen mit `P`

Der Editor vi (9)

- Copy & Paste mit der Maus
  - Wenn Sie die grafische Oberfläche verwenden, geht es auch mit der Maus:
  - Kopieren: Mauszeiger auf 1. Zeichen, klicken (und gedrückt halten), zum letzten Zeichen ziehen, loslassen
  - Einfügen: Cursor zu Ziel bewegen, dann (im Einfügemodus!) die mittlere Maustaste drücken
  - Bei beiden Schritten muss man je nach `vi`-Version evtl. die [Umschalt]-Taste drücken

Der Editor vi (10)

- Datei im Editor öffnen:

  ```
  [esser@quad:~] vi Dateiname
  ```

- zweite Datei an Cursorposition hinzuladen:

  ```
  :read Dateiname
  ```

  (im Befehlsmodus!)

Praktische Übungen

- Aufgabenblatt
  - Umgang mit Dateien und Verzeichnissen
  - Editor vi

Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz C
Topic 103: GNU and Unix commands

Überblick LPIC-1, Prüfung 101

Topic 101: System Architecture

- 101.1 Determine and configure hardware settings
- 101.2 Boot the system
- 101.3 Change runlevels and shutdown or reboot system

Topic 102: Linux Installation and Package Management

- 102.1 Design hard disk layout
- 102.2 Install a boot manager
- 102.3 Manage shared libraries
- 102.4 Use Debian package management
- 102.5 Use RPM and YUM package management

Topic 103: GNU and Unix Commands

- 103.1 Work on the command line
- 103.2 Process text streams using filters
- 103.3 Perform basic file management
- 103.4 Use streams, pipes and redirects
- 103.5 Create, monitor and kill processes
- 103.6 Modify process execution priorities
- 103.7 Search text files using regular expressions
- 103.8 Perform basic file editing operations using vi

Überblick LPIC-1, Prüfung 101

Topic 104: Devices, Linux Filesystems, Filesystem Hierarchy Standard

- 104.1 Create partitions and filesystems
- 104.2 Maintain the integrity of filesystems
- 104.3 Control mounting and unmounting of filesystems
- 104.4 Manage disk quotas
- 104.5 Manage file permissions and ownership
- 104.6 Create and change hard and symbolic links
- 104.7 Find system files and place files in the correct location

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Topic 103: GNU and Unix Commands

103.1 Work on the command line

Description: Candidates should be able to interact with shells and commands using the command line. The objective assumes the bash shell.

Key Knowledge Areas:

- Use single shell commands and one line command sequences to perform basic tasks on the command line.
- Use and modify the shell environment including defining, referencing and exporting environment variables.
- Use and edit command history.
- Invoke commands inside and outside the defined path.

The following is a partial list of the used files, terms and utilities: . , bash, echo, env, exec, export, pwd, set, unset, man, uname, history

Topic 103: GNU and Unix Commands

103.2 Process text streams using filters

Description: Candidates should should be able to apply filters to text streams.

Key Knowledge Areas

- Send text files and output streams through text utility filters to modify the output using standard UNIX commands found in the GNU textutils package.

The following is a partial list of the used files, terms and utilities:
cat, cut, expand, fmt, head, od, join, nl, paste, pr, sed, sort, split, tail, tr, unexpand, uniq, wc

Topic 103: GNU and Unix Commands

103.5 Create, monitor, and kill processes

Description: Candidates should be able to perform basic process management.

Key Knowledge Areas:

- Run jobs in the foreground and background.
- Signal a program to continue running after logout.
- Monitor active processes.
- Select and sort processes for display.
- Send signals to processes.

The following is a partial list of the used files, terms and utilities:
&, bg, fg, jobs, kill, nohup, ps, top, free, uptime, killall

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Topic 103: GNU and Unix Commands

103.6 Modify process execution properties

Description: Candidates should should be able to manage process execution priorities.

Key Knowledge Areas:

- Know the default priority of a job that is created.
- Run a program with higher or lower priority than the default.
- Change the priority of a running process.

The following is a partial list of the used files, terms and utilities:
nice, ps, renice, top

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

103.1 Arbeiten in der Shell

Aus Einführung und Übungsblatt 1 bereits bekannt:

- pwd: aktuelles (Arbeits-) Verzeichnis anzeigen
- cd: Verzeichniswechsel
- .. : nächst höheres Verzeichnis
- ls: Verzeichnisinhalt anzeigen
- cp: Datei kopieren
- vi: Text-Editor
- mkdir: Verzeichnis erzeugen
- rmdir: Verzeichnis löschen
- rm: Datei löschen
- rm -r: Verzeichnis rekursiv löschen
- touch: Datei (leer) erzeugen; Zugriffsdatum aktualisieren
- less: Datei anzeigen
- grep: Suchen in Datei
- head, tail: Anfang und Ende einer Datei
- man: Hilfe anzeigen
- dmesg: Systemmeldungen ausgeben
- wc: word count
- shutdown: System runter fahren

103.1: Shell-Variablen (1)

- Die Shell (und auch andere Programme) nutzen **Umgebungsvariablen** (für Optionen, Einstellungen etc.)
- „set“ gibt eine Liste aller in dieser Shell gesetzten Variablen aus

```
$ set
BASH=/bin/bash
BASH_VERSION='3.2.48(1)-release'
COLUMNS=156
COMMAND_MODE=unix2003
DIRSTACK=()
DISPLAY=/tmp/launch-Lujw2L/org.x:0
EUID=501
GROUPS=()
HISTFILE=/home/esser/.bash_history
HISTFILESIZE=500
HISTSIZE=500
HOME=/home/esser
HOSTNAME=macbookpro.fritz.box
...
```

103.1: Shell-Variablen (2)

- Einzelne Variablen geben Sie mit „echo“ und einem Dollar-Zeichen ($) vor dem Variablennamen aus

```
$ echo $SHELL
/bin/bash
$ _
```

- zum Ändern / Setzen schreiben Sie „var=wert“:

```
$ TESTVAR=fom
$ echo $TESTVAR
fom
$ set | grep TEST
TESTVAR=fom
$ _
```

- Sie können Variablen auch **exportieren**:

```
$ export TESTVAR
$ _
```

→ nächste Folie

103.1: Shell-Variablen (3)

- Exportieren?

  Wert einer Variablen gilt nur lokal in der laufenden Shell.

- Exportierte Variablen gelten auch in aus der Shell heraus gestarteten Programmen

```
$ A=eins; B=zwei; export A
$ echo "A=$A B=$B"
A=eins B=zwei
$ bash              # neue Shell starten; das ist ein neues Programm!
$ echo "A=$A B=$B"
A=eins B=
$ exit              # diese zweite Shell verlassen, zurück zur ersten
$ echo "A=$A B=$B"
A=eins B=zwei
```

!

103.1: Shell-Variablen (4)

- Liste aller exportierten Variablen gibt „export“ ohne Argument aus – allerdings in ungewöhnlicher Syntax

```
$ export
declare -x A="1"
declare -x Apple_PubSub_Socket_Render="/tmp/launch-CYfDhh/Render"
declare -x COMMAND_MODE="unix2003"
declare -x DISPLAY="/tmp/launch-Lujw2L/org.x:0"
declare -x HOME="/Users/esser"
declare -x INFOPATH="/sw/share/info:/sw/info:/usr/share/info"
declare -x LOGNAME="esser"
...
```

- (Hintergrund: „declare -x VAR“ exportiert ebenfalls die Variable VAR, ist also dasselbe wie „export VAR“)

103.1: History (1)

- Shell merkt sich die eingegebenen Befehle („History“)
- Komplette Ausgabe mit „history“:

```
$ history
1  df -h
2  ll
3  /opt/seamonkey/seamonkey
4  dmesg|tail
5  ping hgesser.de
6  google-chrome
7  killall kded4
```

- Wie viele Einträge? Normal 500:

```
$ echo $HISTSIZE
500
```

103.1: History (2)

- Neben Ausgabe der kompletten History gibt es auch eine intelligente Suche nach alten Kommandos: [Strg-R]

```
$  # Suche nach dem letzten echo-Aufruf
$ ^R
(reverse-i-search)`ech': echo $HISTFILESIZE
```

- mit [Eingabe] ausführen
- weitere [Strg-R] liefern ältere Treffer
- Außerdem: Mit [Pfeil hoch], [Pfeil runter] durch alte Befehle blättern
- gefundenes Kommando kann übernommen und überarbeitet werden

103.2: Filter für Text-Streams

- Idee beim Filter:
  - Standardeingabe in Standardausgabe verwandeln
  - Ketten aus Filtern zusammen bauen:
  - `prog1 | filter1 | filter2 | filter3` ...
  - mit Eingabedatei:
    `prog1 < eingabe | filter1 | ...`
- cat, cut, expand, fmt, head, od, join, nl, paste, pr, sed, sort, split, tail, tr, unexpand, uniq, wc

103.2: cat

- cat steht für con**cat**enate (aneinanderfügen)
- gibt mehrere Dateien unmittelbar hintereinander aus
- auf Wunsch auch nur eine Datei
  → Mini-Dateibetrachter
- Spezialoptionen:
  - -n (Zeilennummern)
  - -T (Tabs als ^I anzeigen)
  - … und einige weitere (siehe: man cat)

103.2: cut

- cut kann spaltenweise Text ausschneiden – Spalten sind wahlweise definierbar über
  - Zeichenpositionen
  - Trennzeichen (die logische Spalten voneinander trennen)

c: character; zeichenbasiert

d: delimiter; Trennzeichen

```
$ cat test.txt
1234 678901 234
abc def ghijklmn
r2d2 12 99
1 2 3
Langer Testeintrag
```

```
$ cut -c3-8 test.txt
34 678
c def
d2 12
2 3
nger T
```

```
$ cut -d" " -f2,3 test.txt
678901 234
def ghijklmn
12 99
2 3
Testeintrag
```

f: field (Feld)

103.2: fmt

- fmt (**for**mat) bricht Textdateien um

keine Umbrüche

```
$ cat test.txt
Das ist mal ein Beispiel fuer einen Satz. Das ist mal ein Beispiel fue
r einen Satz. Das ist mal ein Beispiel fuer einen Satz. Das ist mal ei
n Beispiel fuer einen Satz. Das ist mal ein Beispiel fuer einen Satz.
Das ist mal ein Beispiel fuer einen Satz. Das ist mal ein Beispiel fue
r einen Satz. Das ist mal ein Beispiel fuer einen Satz. Das ist mal ei
n Beispiel fuer einen Satz. Das ist mal ein Beispiel fuer einen Satz.

$ fmt test.txt
Das ist mal ein Beispiel fuer einen Satz. Das ist mal ein Beispiel
fuer einen Satz. Das ist mal ein Beispiel fuer einen Satz. Das ist
mal ein Beispiel fuer einen Satz. Das ist mal ein Beispiel fuer
einen Satz. Das ist mal ein Beispiel fuer einen Satz. Das ist mal
ein Beispiel fuer einen Satz. Das ist mal ein Beispiel fuer einen
Satz. Das ist mal ein Beispiel fuer einen Satz. Das ist mal ein
Beispiel fuer einen Satz.
```

Zeilen-umbrüche

- Parameter -w75: Breite 75 (width)

103.2: split

- split kann große Dateien in mehrere Dateien mit angegebener Maximalgröße aufteilen
- (cat fügt diese anschließend wieder zusammen)

```
$ split ZM_ePaper_18_11.pdf -b1440k ZM_ePaper_18_11.pdf.
$ ls -l ZM*
-rw-r--r-- 1 esser esser 10551293 2011-04-29 06:58 ZM_ePaper_18_11.pdf
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.aa
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.ab
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.ac
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.ad
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.ae
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.af
-rw-r--r-- 1 esser esser  1474560 2011-04-29 14:46 ZM_ePaper_18_11.pdf.ag
-rw-r--r-- 1 esser esser   229373 2011-04-29 14:46 ZM_ePaper_18_11.pdf.ah
$ cat ZM_ePaper_18_11.pdf.* > ZM_Kopie.pdf
$ ls -l ZM_Kopie.pdf
-rw-r--r-- 1 esser esser 10551293 2011-04-29 14:48 ZM_Kopie.pdf
$ diff ZM_ePaper_18_11.pdf ZM_Kopie.pdf
$ _
```

103.2: sort

- sort ist ein komplexes Sortier-Tool, das
  - Sortierung nach *n*-ter Spalte
  - alphabetische und numerische Sortierung

  unterstützt
- Einfache Beispiele:

```
$ cat test3.txt
13 Autos
5 LKW
24 Fahrraeder
2 Baeume
Wohnung
Haus
Hotel
Strasse
Allee
```

```
$ sort test3.txt
13 Autos
2 Baeume
24 Fahrraeder
5 LKW
Allee
Haus
Hotel
Strasse
Wohnung
```

```
$ sort -n test3.txt
Allee
Haus
Hotel
Strasse
Wohnung
2 Baeume
5 LKW
13 Autos
24 Fahrraeder
```

103.2: uniq

- uniq (**uniq**ue, einmalig) fasst mehrere identische (aufeinander folgende) Zeilen zu einer zusammen; entfernt also Doppler
- Alternative: Beim Sortieren mit sort kann man über die Option -u (**u**nique) direkt Doppler entfernen;
  - statt `sort datei | uniq` also besser `sort -u datei`

103.2: grep

- grep (**g**lobal/**r**egular **e**xpression/**p**rint) zeigt nur die Zeilen einer Datei, die einen Suchbegriff enthalten – oder nicht enthalten (Option -v)

```
$ wc -l /etc/passwd
57 /etc/passwd
$ grep esser /etc/passwd
esser:x:1000:1000:Hans-Georg Esser,,,:/home/esser:/bin/bash
$ grep /bin/bash /etc/passwd
root:x:0:0:root:/root:/bin/bash
esser:x:1000:1000:Hans-Georg Esser,,,:/home/esser:/bin/bash
$ grep -v /bin/bash /etc/passwd | head -n5
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
```

103.2: sed (1/2)

- sed (**S**tream **Ed**itor) führt (u. a.) Suchen-/Ersetzen-Funktionen in einem Text durch

```
$ cat test4.txt
Das Wort ist ein Wort, und mehrere
Woerter sind der Plural von Wort.
Ohne Woerter oder Worte gibt es
keinen Satz - wir sind wortlos.

$ sed 's/Wort/Bild/' test4.txt
Das Bild ist ein Wort, und mehrere
Woerter sind der Plural von Bild.
Ohne Woerter oder Bilde gibt es
keinen Satz - wir sind wortlos.

$ sed 's/Wort/FOM/g' test4.txt
Das FOM ist ein FOM, und mehrere
Woerter sind der Plural von FOM.
Ohne Woerter oder FOMe gibt es
keinen Satz - wir sind wortlos.
```

```
$ sed 's/Wort/FOM/gi' test4.txt
Das FOM ist ein FOM, und mehrere
Woerter sind der Plural von FOM.
Ohne Woerter oder FOMe gibt es
keinen Satz - wir sind FOMlos.
```

s: substitute (**s**/.../.../gi)
g: global (s/.../.../**g**i)
i: ignore case (s/.../.../g**i**)

Die i-Option gibt es nicht in jeder sed-Version!

103.2: sed (2/2)

- sed-Optionen:
  - -i: in-place-editing, verändert die angegebene Datei; am besten mit Angabe eines Suffix für eine Backup-Datei:
    z. B. `sed -i.bak 's/Wort/Bild/g' test4.txt`
    legt erst Sicherheitskopie test4.txt.bak an und verändert dann test4.txt
  - -e: zum Kombinieren mehrerer Ersetzungen; z. B. `sed -e 's/1/eins/g' -e 's/2/zwei/g' test.txt`
  - weitere Optionen → Manpage

103.2: Reguläre Ausdrücke

- Idee: Allgemeinere Suchbegriffe, vergleichbar mit Wildcards (*, ?) bei Dateinamen
- Muster:
  - . – ein beliebiges Zeichen
  - [abcd] – eines der Zeichen a, b, c, d
  - [2-8] – eines der Zeichen 2, 3, 4, 5, 6, 7, 8
  - ^ – Zeilenanfang
  - $ – Zeilenende
  - ? – vorheriger Ausdruck darf vorkommen, muss aber nicht
  - * – vorheriger Ausdruck kann beliebig oft (auch 0 mal) vorkommen

```
$ cat test5.txt
Haus
Die Hotels
Hotels am Wasser
Bau-Haus-Objekt
Diese Zeile nicht
```

```
$ grep 'H.*s' test5.txt
Haus
Die Hotels
Hotels am Wasser
Bau-Haus-Objekt
```

```
$ sed 's/H.*s/HAUS/g' test5.txt
HAUS
Die HAUS
HAUSer
Bau-HAUS-Objekt
Diese Zeile nicht
```

103.2: Reguläre Ausdrücke

- Beispiele für reguläre Ausdrücke

  (live, in der Shell...)

103.5: Vorder-/Hintergrund (1)

- In der Shell gestartete Anwendungen laufen standardmäßig im **Vordergrund** – d. h.,
  - die Shell ist blockiert, solange das Programm läuft,
  - und es nutzt das aktuelle Terminal (-Fenster) für Ein- und Ausgabe
- Alternativ kann ein Programm im **Hintergrund** laufen:
  - die Shell kann dann sofort weiter genutzt werden (weitere Kommando eingeben),
  - keine Eingabe möglich, aber Ausgabe (auch ins aktuelle Terminal; besser umleiten

103.5: Vorder-/Hintergrund (2)

- Typische Vordergrund-Programme
  - Kommandos, die eine Anfrage sofort beantworten
  - Text-Editoren
  - Compiler
- Typische Hintergrund-Programme
  - manuell gestartete Server (Dienste)
  - unter X Window: grafische Anwendungen (die kein Terminal brauchen, sondern ein eigenes Fenster öffnen)

103.5: Vorder-/Hintergrund (3)

- Programm im Vordergrund starten:
  einfach den Namen eingeben
  Bsp.: `ls -l`
- Programm im Hintergrund starten:
  kaufmännisches Und (&, ampersand) anhängen
  Bsp.: `/usr/sbin/apache2 &`
- Wechsel von Vordergrund in Hintergrund:
  - Programm mit [Strg-Z] unterbrechen
  - Programm mit `bg` in den Hintergrund schicken
- Wechsel von Hinter- in Vordergrund: `fg`

103.5: Job-Verwaltung (1)

- Programme, die aus einer laufenden Shell heraus gestartet wurden, heißen **Jobs** dieser Shell
- Anzeige mit: `jobs`

```
[esser@macbookpro:~]$ jobs
[esser@macbookpro:~]$ nedit &
[1] 77787
[esser@macbookpro:~]$ vi /tmp/test.txt
^Z
[2]+  Stopped                 vi /tmp/test.txt
[esser@macbookpro:~]$ find / > /tmp/ergebnisse.txt &
[3] 77792
[esser@macbookpro:~]$ jobs
[1]   Running                 nedit &
[2]+  Stopped                 vi /tmp/test.txt
[3]-  Running                 find / > /tmp/ergebnisse.txt &
[esser@macbookpro:~]$
```

103.5: Job-Verwaltung (2)

Zustand des Jobs:
- running: aktiv / bereit
- stopped: mit ^Z oder kill -STOP angehalten
- terminated: beendet, wird nur 1x angezeigt

```
[esser@macbookpro:~]$ jobs
[1]   Running         nedit &
[2]+  Stopped         vi /tmp/test.txt
[3]-  Running         find / > ...&
```

Kommandos

1,2,3, …: Job-Nummer

\+ : „current job“ = letzter Job, der
- im Vordergrund gestartet und dann unterbrochen
- oder im Hintergrund gestartet wurde

viele Kommandos (`fg`, `bg`, …) ohne Parameter beziehen sich auf den current job

– : „previous job“ = vorletzter Job mit obiger Eigenschaft

103.5: Job-Verwaltung (3)

- Jobs gezielt ansprechen: %n (mit n = Job-Nummer)

```
[esser@macbookpro:~]$ jobs
[1]   Running                 nedit /tmp/1 &
[2]   Running                 nedit /tmp/2 &
[3]   Running                 nedit /tmp/3 &
[4]-  Running                 nedit /tmp/4 &
[5]+  Running                 nedit /tmp/5 &
[esser@macbookpro:~]$ kill %3
[esser@macbookpro:~]$ jobs
[1]   Running                 nedit /tmp/1 &
[2]   Running                 nedit /tmp/2 &
[3]   Terminated              nedit /tmp/3
[4]-  Running                 nedit /tmp/4 &
[5]+  Running                 nedit /tmp/5 &
[esser@macbookpro:~]$ jobs
[1]   Running                 nedit /tmp/1 &
[2]   Running                 nedit /tmp/2 &
[4]-  Running                 nedit /tmp/4 &
[5]+  Running                 nedit /tmp/5 &
```

103.5: Job-Verwaltung (4)

Kommandos zur Job-Verwaltung

- `bg %n`: in den Hintergrund bringen
- `fg %n`: in den Vordergrund bringen
- `kill %n`: beenden
- `kill -`*`SIGNALNAME`* `%n`: Signal schicken, siehe nächste Folie
- `disown %n`: Verbindung mit der Shell lösen; `disown -a`: für alle Jobs
- `wait %n`: Warten, bis Job beendet ist

103.5: Job-Verwaltung (5)

Signale (mit Signalnummer)

- `TERM, 15`: terminieren, beenden (mit „Aufräumen“); Standardsignal
- `KILL, 9`: sofort abbrechen (ohne Aufräumen)
- `STOP, 19`: unterbrechen (entspricht ^Z)
- `CONT, 18`: continue, fortsetzen; hebt `STOP` auf
- `HUP, 1`: hang-up, bei vielen Server-Programmen: Konfiguration neu einlesen (traditionell: Verbindung zum Terminal unterbrochen)
- Liste aller Signale: `kill -l`

103.5: Jobs vs. Prozesse

- Die Bezeichnung **Job** bezieht sich immer auf die aktuelle Shell-Sitzung
- Jobs, die Sie in verschiedenen Shells starten, haben nichts miteinander zu tun
- Allgemeinerer Begriff: **Prozess**
- Tool für die Prozessanzeige: `ps`
- Die (Gesamt-) Prozessliste (`ps auxw`) enthält alle Prozesse auf dem Linux-System

103.5: Prozesse (1)

- `ps` (ohne Optionen) zeigt alle Prozesse an, die zur aktuellen Shell-Sitzung gehören – das sind dieselben wie in der Ausgabe von `jobs`:

```
[esser@quadamd:~]$ jobs
[1]+  Angehalten    vi /tmp/test4
```

```
[esser@quadamd:~]$ ps
  PID TTY          TIME CMD
27967 pts/0    00:00:00 bash
28160 pts/0    00:00:00 vi
28168 pts/0    00:00:00 ps
```

- über Optionen (ohne „–“) lässt sich die Ausgabe von `ps` anpassen, z. B. `ps auxw`:
  - a: alle Prozesse (die ein Terminal haben)
  - u: „user oriented format“
  - x: auch Prozesse ohne Terminal
  - w: „wide“: Befehlszeilen nicht abschneiden

103.5: Prozesse (2)

```
[esser@quadamd:~]$ ps auw
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      1122  0.0  0.0   1872   580 tty4     Ss+  Apr17   0:00 /sbin/getty -8 38400 tty4
root      1127  0.0  0.0   1872   580 tty5     Ss+  Apr17   0:00 /sbin/getty -8 38400 tty5
root      1150  0.0  0.0   1872   576 tty2     Ss+  Apr17   0:00 /sbin/getty -8 38400 tty2
root      1151  0.0  0.0   1872   576 tty3     Ss+  Apr17   0:00 /sbin/getty -8 38400 tty3
root      1153  0.0  0.0   1872   580 tty6     Ss+  Apr17   0:00 /sbin/getty -8 38400 tty6
root      1776  0.0  0.0   5248  2156 tty1     Ss   Apr17   0:00 /bin/login --
esser     1941  0.0  0.1   9272  4816 tty1     S    Apr17   0:01 -bash
esser     2956  0.0  0.0   1912   540 tty1     S+   Apr17   0:00 /bin/sh /usr/bin/startx
esser     2973  0.0  0.0   3128   708 tty1     S+   Apr17   0:00 xinit /home/esser/.xinitrc -- /etc/X11/x
root      2974  0.0  1.8  85616 75712 tty8     Ss+  Apr17  14:56 /usr/bin/X -nolisten tcp :0 -auth /tmp/s
esser     2977  0.0  0.0   1912   564 tty1     S    Apr17   0:00 /bin/sh /home/esser/.xinitrc
esser     3037  0.0  0.0   3456   560 tty1     S    Apr17   0:00 dbus-launch --sh-syntax --exit-with-sess
esser     3045  0.0  0.0   1700    64 tty1     S    Apr17   0:00 /usr/lib/kde4/libexec/start_kdeinit +kcm
esser     3174  0.0  0.0   1832   244 tty1     S    Apr17   0:00 kwrapper4 ksmserver
esser     3454  0.0  0.0   6552  2040 pts/2    Ss   Apr17   0:00 /bin/bash
root      3775  0.0  0.0   8560  2112 pts/2    S    Apr17   0:00 sudo su
root      3776  0.0  0.0   8316  1764 pts/2    S    Apr17   0:00 su
root      3784  0.0  0.0   6656  2172 pts/2    S+   Apr17   0:00 bash
esser    10674  1.4  7.0 443588 289940 pts/6   Sl   Apr30 131:17 /usr/lib/opera/opera
esser    10694  0.0  0.1  21764  7552 pts/6    S    Apr30   0:08 /usr/lib/opera//operapluginwrapper 101 1
esser    10695  0.0  0.0   3040   548 pts/6    S    Apr30   0:02 /usr/lib/opera//operapluginceaner 10674
esser    10699  1.3  0.0      0     0 pts/6    Z    Apr30 119:12 [gtk-gnash] <defunct>
esser    12198  0.0  0.0   6552  1828 pts/4    Ss   Apr17   0:00 /bin/bash
root     12583  0.0  0.0   8560  2116 pts/4    S    Apr17   0:00 sudo su
root     13077  0.0  0.0   8316  1768 pts/4    S    Apr17   0:00 su
root     13097  0.0  0.0   6612  2052 pts/4    S+   Apr17   0:00 bash
esser    13653  0.0  0.0   6768  2140 pts/3    Ss   Apr17   0:00 /bin/bash
esser    18905  0.0  0.1   9128  4712 pts/6    Ss+  Apr22   0:02 /bin/bash
esser    27587  0.0  0.0   5256  2144 pts/3    S+   19:40   0:00 ssh backup
esser    28613  0.0  0.1  11744  7264 pts/5    Ss+  22:45   0:00 -bash
esser    29091  0.0  0.1  12156  7704 pts/0    Ss   22:58   0:00 -bash
esser    29307  0.0  0.0   5856  1172 pts/0    R+   23:42   0:00 ps auw
```

103.5: Prozesse (3)

- Spalten in der Ausgabe von `ps auw`:
  - `USER`: Benutzer, dem der Prozess gehört
  - `PID`: Prozess-ID
  - `%CPU`: CPU-Nutzung in Prozent (Verhältnis Rechenzeit / Lebenszeit)
  - `%MEM`: RSS / RAM-Größe in Prozent
  - `VSZ`: Größe des virtuellen Speichers (in KByte)
  - `RSS`: Resident Set Size, aktuell genutzter Speicher (KByte)
  - `TTY`: Terminal
  - `STAT`: Prozess-Status
  - `START`: Startzeit des Prozesses (ggf. Datum)
  - `TIME`: bisherige Gesamtlaufzeit
  - `COMMAND`: Kommando (Aufruf)

103.5: Prozesse (4)

- Signale an beliebige Prozesse schicken
  - wie vorher: Kommando `kill`
  - aber: nicht `kill %`n (n=Job-ID), sondern `kill p` (p = PID)
  - auch hier Angabe eines Signals möglich
- `killall` *`Name`*: alle Prozesse beenden, deren ausführbares Programm *`Name`* heißt
- mit `killall` auch (wie bei `kill`) andere Signale an alle Prozesse mit passendem Namen schicken

Studium neben dem Beruf | FOM Hochschule für Oekonomie & Management University of Applied Sciences

103.5: Prozesse (5): pstree

- Darstellung der Prozessliste auch in Baumansicht möglich: `pstree`
- Jeder Prozess hat einen Vaterprozess
- identische Teilbäume nur 1x
- Option `-p`: Prozess-IDs anzeigen

```
[esser@quadamd:~]$ pstree
init-+-NetworkManager-+-dhclient
     |                `-2*[{NetworkManager}]
     |-acpid
     |-akonadi_control-+-2*[akonadi_contact]
     |                 |-3*[akonadi_ical_re]
     |                 |-akonadi_maildir
     |                 |-akonadi_maildis
     |                 |-akonadi_nepomuk
     |                 |-akonadi_vcard_r
     |                 |-akonadiserver-+-mysqld---23*[{mysqld}]
     |                 |               `-15*[{akonadiserver}]
     |                 `-3*[{akonadi_contro}]
     |-atd
     |-avahi-daemon---avahi-daemon
     |-console-kit-dae---64*[{console-kit-da}]
     |-cron
     |-cupsd
[...]
     |-knotify4---6*[{knotify4}]
     |-konsole-+-2*[bash---sudo---su---bash]
     |         |-bash---ssh
     |         |-bash---opera-+-operapluginclea
     |         |              |-operapluginwrap---gtk-gnash
     |         |              `-6*[{opera}]
     |         `-2*[{konsole}]
     |-krunner---11*[{krunner}]
     |-kuiserver
     |-kwalletd
     |-login---bash---startx---xinit-+-.xinitrc---kwrapper4
     |                               `-Xorg
     |-upstart-socket-
     |-upstart-udev-br
     |-vpnagentd
     `-wpa_supplicant
```

103.5: Hang-up, No Hang-up

- Wenn Sie sich in der Konsole abmelden (`exit`) oder unter X Window ein Terminalfenster schließen, erhalten alle in der Shell laufenden Jobs das HUP-Signal (Hang-up).
- Die Standardreaktion auf HUP ist: beenden
- Abmelden / Fenster schließen beendet also alle darin gestarteten Programme
- Auswege:
  - Programme mit `nohup` starten oder
  - Prozess mit `disown` von der Shell lösen

103.5: nohup

- nohup hat zwei Funktionen:
  - der gestartete Prozess ignoriert HUP-Signale
  - Ausgaben des Prozesses (auf die Standard-ausgabe) erscheinen nicht im Terminal, sondern werden in die Datei `nohup.out` geschrieben

```
[esser@macbookpro:~]$ nedit /tmp/1 &
[1] 79142
[esser@macbookpro:~]$ nohup nedit /tmp/2 &
[2] 79144
appending output to nohup.out
```

103.5: top (1)

- Prozesse nach CPU-Auslastung sortiert anzeigen: `top`
- Anzeige wird regelmäßig aktualisiert

```
top - 00:07:30 up 19 days,  6:15,  7 users,  load average: 0.00, 0.02, 0.05
Tasks: 194 total,   2 running, 191 sleeping,   0 stopped,   1 zombie
Cpu(s):  1.2%us,  0.7%sy,  0.0%ni, 98.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   4120180k total,  2353392k used,  1766788k free,   560756k buffers
Swap:  4191936k total,        0k used,  4191936k free,   566868k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 3177 esser     20   0  302m  45m  20m S    2  1.1 260:37.29 knotify4
10674 esser     20   0  433m 283m  27m S    2  7.0 131:47.89 opera
 3432 esser     20   0  893m 155m  33m S    1  3.9 253:02.73 seamonkey-bin
 1093 messageb  20   0  4524 2664  840 R    0  0.1  16:55.65 dbus-daemon
 1576 root      20   0 12072 7272 3408 S    0  0.2   8:50.93 python
 2076 root      20   0  5560  972  692 S    0  0.0   5:07.52 udisks-daemon
 3238 esser     20   0  367m  64m  35m S    0  1.6   1:13.69 krunner
29348 esser     20   0  2632 1180  852 R    0  0.0   0:00.03 top
    1 root      20   0  3028 1892 1236 S    0  0.0   0:02.06 init
    2 root      20   0     0    0    0 S    0  0.0   0:00.73 kthreadd
    3 root      20   0     0    0    0 S    0  0.0   1:36.46 ksoftirqd/0
    6 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/0
   17 root       0 -20     0    0    0 S    0  0.0   0:00.00 cpuset
   18 root       0 -20     0    0    0 S    0  0.0   0:00.00 khelper
   19 root       0 -20     0    0    0 S    0  0.0   0:00.00 netns
   21 root      20   0     0    0    0 S    0  0.0   0:02.28 sync_supers
   22 root      20   0     0    0    0 S    0  0.0   0:00.05 bdi-default
   23 root       0 -20     0    0    0 S    0  0.0   0:00.00 kintegrityd
   24 root       0 -20     0    0    0 S    0  0.0   0:00.00 kblockd
   25 root       0 -20     0    0    0 S    0  0.0   0:00.00 kacpid
   26 root       0 -20     0    0    0 S    0  0.0   0:00.00 kacpi_notify
   27 root       0 -20     0    0    0 S    0  0.0   0:00.00 kacpi_hotplug
```

103.5: top (2)

- Sortierung in `top` anpassbar (Sortierspalte ändern mit < und >)
- Über der Prozessliste: Informationen zur Gesamtauslastung des Systems
- umschaltbar auf Anzeige/CPU bzw. /Kern: 1

```
top - 00:14:22 up 19 days,  6:22,  7 users,  load average: 0.05, 0.03, 0.05
Tasks: 194 total,   2 running, 191 sleeping,   0 stopped,   1 zombie
Cpu0  :  0.7%us,  0.3%sy,  0.0%ni, 99.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu1  :  1.7%us,  1.0%sy,  0.0%ni, 97.3%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu2  :  0.0%us,  0.3%sy,  0.0%ni, 99.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Cpu3  :  3.6%us,  0.7%sy,  0.0%ni, 95.8%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   4120180k total,  2353400k used,  1766780k free,   560948k buffers
Swap:  4191936k total,        0k used,  4191936k free,   566868k cached
```

103.5: free, uptime

- Weitere Systeminformationen:
  - `free` (freien Speicher anzeigen)

```
[esser@quadamd:~]$ free
             total       used       free     shared    buffers     cached
Mem:       4120180    2347264    1772916          0     561488     566896
-/+ buffers/cache:    1218880    2901300
Swap:      4191936          0    4191936
```

  - `uptime` (wie lange läuft das System schon?)

```
[esser@quadamd:~]$ uptime
 00:34:08 up 19 days,  6:42,  6 users,  load average: 0.06, 0.07, 0.05
```

103.6: Prozess-Priorität (1)

- Jeder Linux-Prozess hat eine **Priorität**. Diese bestimmt, welchen Anteil an Rechenzeit der Prozess erhält.
- Priorität ist ein Wert zwischen -20 und 19.
- Konvention: hohe Priorität = kleiner Wert
  (also: -20 = maximale Prior., 19 = minimale Prior.)
- unter Linux/Unix auch als **nice value** („Nettigkeit“) bezeichnet: 19 = extrem nett, -20 = gar nicht nett
- Bei Programmstart Priorität mit `nice` setzen

103.6: Prozess-Priorität (2)

- `nice` mit Priorität (Option) und auszuführendem Kommando (folgende Argumente) aufrufen, z. B.

```
[esser@quadamd:~]$ nice -5 program &

[esser@quadamd:~]$ ps -eo user,pid,ni,cmd
USER        PID  NI CMD
...
root      28299   0 [kworker/2:0]
root      28300   0 [kworker/0:1]
esser     28301   5 program
esser     28303   0 ps -eo user,pid,ni,cmd
```

- negative Nice-Werte kann nur Administrator *root* setzen:

```
[esser@quadamd:~]$ nice --10 vi
nice: kann Priorität nicht setzen: Keine Berechtigung
```

103.6: Prozess-Priorität (3)

- Alternative Syntax für bessere Lesbarkeit:
  `nice -n Wert` (statt `nice -Wert`)
- vor allem für negative Werte intuitiver:
  `nice -n -10` (statt `nice --10`)

```
[esser@quadamd:~]$ su
Passwort:
root@quadamd:~# nice -n -10 program &
[1] 28373
root@quadamd:~# ps -eo user,pid,ni,cmd
USER       PID  NI CMD
[...]
root     28311   0 su
root     28319   0 bash
root     28373 -10 program
root     28375   0 ps -eo user,pid,ni,cmd
```

103.6: Prozess-Priorität (4)

- Genauer: Nice-Wert in `nice`-Aufruf ist relativ zum „aktuellen Nice-Level“ (Standard: 0)
- angebener Wert wird zum Nice-Wert addiert:

```
[esser@quadamd:~]$ nice
0
[esser@quadamd:~]$ nice -n 5 bash
[esser@quadamd:~]$ nice
5
[esser@quadamd:~]$ nice -n 10 bash
[esser@quadamd:~]$ nice
15
[esser@quadamd:~]$ _
```

103.6: Prozess-Priorität (5)

- Nice-Wert für laufendes Programm ändern: `renice`
- Wert <0 setzen darf nur *root*
- in alten Linux-Versionen galt auch: aktuellen Wert verringern darf nur *root*)

```
[esser@quadamd:~]$ program &
[5] 28937
[esser@quadamd:~]$ ps -eo user,pid,ni,cmd
USER       PID  NI CMD
esser    28937   0 program
[esser@quadamd:~]$ renice 5 28937
28937: Alte Priorität: 0, neue Priorität: 5
[esser@quadamd:~]$ ps -eo user,pid,ni,cmd
USER       PID  NI CMD
esser    28937   5 program
[esser@quadamd:~]$ renice 0 28937
28937: Alte Priorität: 5, neue Priorität: 0
[esser@quadamd:~]$ renice -10 28937
renice: 28937: setpriority: Keine Berechtigung
```

Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz D
Topic 102: Package Management
102.4 / 102.5 Debian; RPM/YUM

Topic 102: Linux Installation and Package Managment

102.4 Use Debian package management

Description: Candidates should be able to perform package management using the Debian package tools.

Key Knowledge Areas:

- Install, upgrade and uninstall Debian binary packages.
- Find packages containing specific files or libraries which may or may not be installed.
- Obtain package information like version, content, dependencies, package integrity and installation status (whether or not the package is installed).

The following is a partial list of the used files, terms and utilities:
/etc/apt/sources.list, dpkg, dpkg-reconfigure, apt-get, apt-cache, aptitude

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Topic 102: Linux Installation and Package Managment

102.5 Use RPM and YUM package management

Description: Candidates should be able to perform package management using RPM and YUM tools.

Key Knowledge Areas:

- Install, re-install, upgrade and remove packages using RPM and YUM.
- Obtain information on RPM packages such as version, status, dependencies, integrity and signatures.
- Determine what files a package provides, as well as find which package a specific file comes from.

The following is a partial list of the used files, terms and utilities:
rpm, rpm2cpio, /etc/yum.conf, /etc/yum.repos.d/, yum, yumdownloader

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Grundlagen Paketverwaltung (1)

- Verschiedene Linux-Distributionen verwenden unterschiedliche Software-Paketformate:
  - *.*deb* – Debian-Paketformat (z. B. Debian, Ubuntu, Knoppix)
  - *.*rpm* – Red-Hat-Paketformat (z. B. OpenSuse, Fedora / Red Hat, Mandriva)
  - weitere (weniger verbreitete Formate), z. B. *.*txz* (Slackware, *xz*-komprimiertes *tar*-Archiv)
- Pakete enthalten die eigentliche Software und „Anleitungen“ für (De-) Installation

Grundlagen Paketverwaltung (2)

- Die beiden „großen“ Paketformate sind
  - das Debian-Format (*.*deb*) und
  - das RPM-Format (*.*rpm*, Red Hat Package Manager)
- Software-Verwaltung mit *deb/rpm*-Paketen findet immer auf zwei Ebenen statt:
  - auf Ebene einzelner Pakete; Tools:
    - `dpkg` (Debian Package Manager)
    - `rpm` (Red Hat Package Manager)
  - auf Ebene von Repositories; Tools:
    - APT (Advanced Packaging Tool / Debian)
    - YUM (Yellowdog Updater, Modified / Red Hat, Fedora)

Grundlagen Paketverwaltung (3)

- Funktionen auf Paketebene
  - Paket installieren / deinstallieren / aktualisieren
  - Paketinhalt anzeigen
  - erkennt → Abhängigkeiten und → Konflikte, verweigert evtl. (De-) Installation oder Update
  - zu installierter Datei das zugehörige Paket finden
  - Installation / Update benötigt immer eine konkrete Paketdatei

Grundlagen Paketverwaltung (4)

- Funktionen auf Repository-Ebene (1)
  - Ein Repository (Repo) ist eine Paketquelle, z. B.
    - Installations-CD/DVD
    - Verzeichnis auf einem Web- oder FTP-Server
    - lokales Verzeichnis
  - Repos haben ein Inhaltsverzeichnis mit den wichtigsten Metadaten zu allen vorhandenen Paketen
  - Repos für jede Distributions-Version separat

Grundlagen Paketverwaltung (5)

- Funktionen auf Repository-Ebene (2)
  - Anlegen / Aktualisieren / Entfernen von Repos (→ parallele Nutzung mehrerer Repos möglich)
  - Suche nach Paketen in allen eingetragenen Repos
  - Installation mit automatischem Auflösen von Abhängigkeiten
  - Komplett-Upgrade auf neue Version einer Linux-Distribution
  - Drittanbieter-Repos (z. B. für Multimedia-Pakete, Treiber vom Hardware-Hersteller oder proprietäre Software)

Paketebene (1)

- Paket installieren:

```
[root@redhat:~]# ls
testpaket-1.2.3.i386.rpm
[root@redhat:~]# rpm -ivh testpaket-1.2.3.i386.rpm
Preparing...                ########################################### [100%]
   1:testpaket              ########################################### [100%]
[root@redhat:~]# _
```

Optionen -vh erzeugen die Ausgabe

```
[root@debian:~]# ls
joe_3.5-1.1_amd64.deb
[root@debian:~]# dpkg -i joe_3.5-1.1_amd64.deb
Wähle vormals abgewähltes Paket joe. (Lese Datenbank ...
113154 Dateien und Verzeichnisse sind derzeit installiert.)
Entpacke joe (aus .../archives/joe_3.5-1.1_amd64.deb) ...
Richte joe ein (3.5-1.1) ...
[root@debian:~]# _
```

Paketebene (2)

- Paket aktualisieren (Update):

```
[root@redhat:~]# rpm -ivh dhcpcd-3.2.3-66.69.1.i586.rpm
Preparing...   ########################################### [100%]
        file /sbin/dhcpcd from install of dhcpcd-3.2.3-
66.69.1.i586 conflicts with file from package dhcpcd-
3.2.3-65.1.i586

[root@redhat:~]# rpm -Uvh dhcpcd-3.2.3-66.69.1.i586.rpm
Preparing...   ########################################### [100%]
   1:dhcpcd    ########################################### [100%]
```

```
[root@debian:~]# dpkg -i joe_3.5-1.1_amd64.deb
```

(Update bei Debian auch über Installationsoption `-i`)

Paketebene (3)

- Paket löschen: benötigt nur Paketnamen, nicht die Versionsnummer

```
[root@redhat:~]# rpm -e testpaket
```

e = erase

```
[root@debian:~]# dpkg -r joe
(Lese Datenbank ... 113175 Dateien und
Verzeichnisse sind derzeit installiert.)
Entferne joe ...
```

r = remove

oder

```
[root@debian:~]# dpkg -P joe
(Lese Datenbank ... 113175 Dateien und
Verzeichnisse sind derzeit installiert.)
Entferne joe ...
Lösche Konfigurationsdateien von joe ...
```

P = purge

Paketebene (4)

- Paketinhalte:

```
[root@redhat:~]# rpm -ql rpm
/bin/rpm
/etc/init.d/rpmconfigcheck
/etc/rpm
/usr/bin/gendiff
/usr/bin/rpm2cpio
/usr/bin/rpmbuild
/usr/bin/rpmdb
...
```

ql = query, list

```
[root@debian:~]# dpkg -L dpkg
/.
/var
/var/lib
/var/lib/dpkg
/var/lib/dpkg/updates
/var/lib/dpkg/parts
/var/lib/dpkg/info
 joe ...
```

L = list

Paketebene (5)

- Paketinformationen (rpm):

```
[root@redhat:~]# rpm -qi rpm
Name        : rpm                          Relocations: (not relocatable)
Version     : 4.8.0                             Vendor: openSUSE
Release     : 27.1                          Build Date: Do 17 Feb 2011 19:09:21 CET
Install Date: Di 15 Mär 2011 11:01:20 CET   Build Host: build24
Group       : System/Packages               Source RPM: rpm-4.8.0-27.1.src.rpm
Size        : 3674658                          License: GPLv2+
Signature   : RSA/8, Do 17 Feb 2011 19:09:42 CET, Key ID b88b2fd43dbdc284
Packager    : http://bugs.opensuse.org
Summary     : The RPM Package Manager
Description :
RPM Package Manager is the main tool for managing the software packages
of the SuSE Linux distribution.

RPM can be used to install and remove software packages. With rpm, it
is easy to update packages.  RPM keeps track of all these manipulations
in a central database.  This way it is possible to get an overview of
all installed packages.  RPM also supports database queries.
```

qi = query, info

Paketebene (6)

- Paketinformationen ():

s = status

```
[root@debian:~]# dpkg -s dpkg
Package: dpkg
Essential: yes
Status: install ok installed
Priority: required
Section: admin
Installed-Size: 7276
Origin: debian
Maintainer: Dpkg Developers <debian-dpkg@lists.debian.org>
Architecture: i386
Version: 1.14.31
Replaces: manpages-de (<= 0.4-3), manpages-pl (<= 20051117-1)
Pre-Depends: libc6 (>= 2.7-1), coreutils (>= 5.93-1), lzma
Suggests: apt
Conflicts: apt (<< 0.7.7), aptitude (<< 0.4.7-1), dpkg-dev (<< 1.14.16)
Conffiles:
 /etc/logrotate.d/dpkg 501f8c90b83c7ea180868ca82e1e82d1
 /etc/dpkg/origins/debian 731423fa8ba067262f8ef37882d1e742
 /etc/dpkg/dpkg.cfg f4413ffb515f8f753624ae3bb365b81b
Description: Debian package management system
 This package provides the low-level infrastructure for handling the
 installation and removal of Debian software packages.
 .
 For Debian package development tools, install dpkg-dev.
Homepage: http://wiki.debian.org/Teams/Dpkg
```

Paketebene (7)

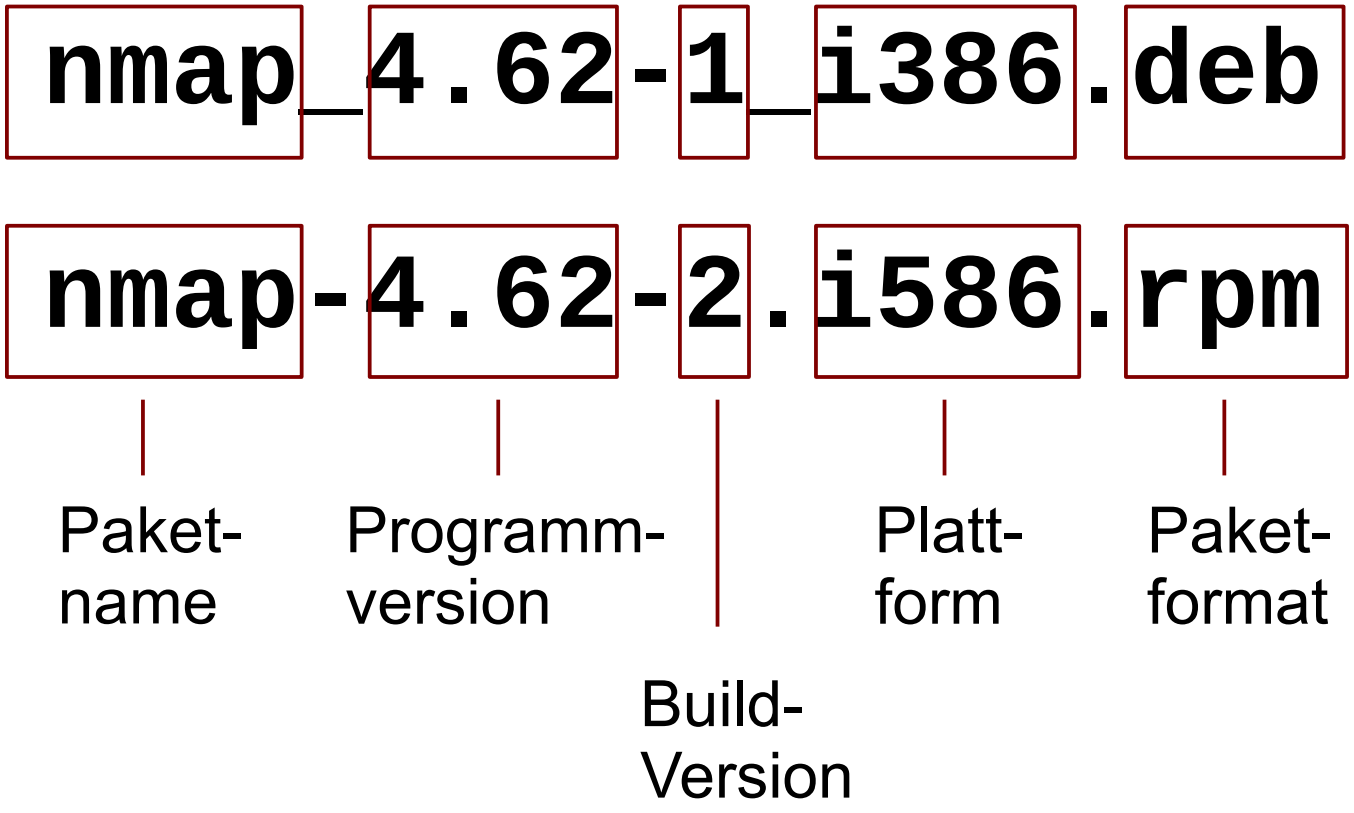
Paketkonvertierung

- `alien` wandelt RPM- und DEB-Archiv ineinander um

```
[root@debian:~]# alien --to-rpm nmap_4.62-1_i386.deb
Warning: Skipping conversion of scripts in package nmap: postinst prerm
Warning: Use the --scripts parameter to include the scripts.
nmap-4.62-2.i386.rpm generated
[root@debian:~]# ls -l
-rw-r--r-- 1 root root 1054002 May 13 20:55 nmap-4.62-2.i386.rpm
-rw-r--r-- 1 root root 1048536 May 13 20:54 nmap_4.62-1_i386.deb
[root@debian:~]# rpm -qip nmap-4.62-2.i386.rpm
Name        : nmap                         Relocations: (not relocatable)
Version     : 4.62                              Vendor: (none)
Release     : 2                          Build Date: Fri May 13 20:55:33 2011
Install Date: (not installed)            Build Host: hgesser.com
Group       : Converted/net              Source RPM: nmap-4.62-2.src.rpm
Size        : 3461471                       License: see /usr/share/doc/nmap/copyri
Signature   : (none)
Summary     : The Network Mapper
Description :
Nmap is a utility for network exploration or security auditing. It
supports ping scanning (determine which hosts are up), many port
...
```

Paketebene (8)

Aufbau der Paketnamen



Plattformen:

i386: Intel i386
i486, i586, i686: ...
sparc: Sun SPARC
ppc: IBM, Apple
ppc64: IBM PowerPC
axp: DEC Alpha
ia64: Intel Itanium
x86_64: PC, 64 Bit
noarch: übergreifend

```
# dpkg-architecture -L
armel
lpia
i386
ia64
alpha
amd64
armeb
arm
...
```

Paketebene (9)

Vollständige Paketliste:

```
# rpm -qa
gpg-pubkey-3dbdc284-4be1884d
bundle-lang-gnome-en-11.4-5.13.1.noarch
translation-update-11.3-7.1.noarch
yast2-trans-stats-2.19.0-4.1.noarch
openSUSE-release-ftp-11.4-1.9.i586
apparmor-docs-2.5.1.r1445-52.55.1.i586
lsscsi-0.23-6.1.i586
libspeex1-1.1.999_1.2rc1-9.1.i586
...
```

-qa: query, all

```
# dpkg -l
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Cfg-files/Unpacked/Failed-cfg/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Hold/Reinst-required/X=both-problems (Status,Err: uppercase=bad)
||/ Name                    Version            Description
+++-=======================-==================-=======================================
ii  adduser                 3.110              add and remove users and groups
ii  alien                   8.72               convert and install rpm and other package
ii  apache2                 2.2.9-10+lenny9    Apache HTTP Server metapackage
ii  apache2-mpm-prefork     2.2.9-10+lenny9    Apache HTTP Server - traditional non-thre
ii  apache2-utils           2.2.9-10+lenny9    utility programs for webservers
ii  apache2.2-common        2.2.9-10+lenny9    Apache HTTP Server common files
ii  apt                     0.7.20.2+lenny2    Advanced front-end for dpkg
...
```

-l: list

Paketebene (10)

- `rpm` und `dpkg` werden heute nur noch selten direkt benutzt
- stattdessen Paketverwaltung mit „höheren“ Tools wie YUM und APT

Abhängigkeiten (1)

- Ein Paket hat **Abhängigkeiten** (engl. **dependencies**), wenn weitere Pakete oder bestimmte Dateien zwingend installiert sein müssen, damit es funktioniert
- Diese zusätzlich benötigten Pakete / Dateien nennt man die Abhängigkeiten
- Beispiele:
  - Paket `mc` (Midnight Commander) benötigt die Bibliothek `libc.so.6`
  - Paket `openssh-askpass` benötigt Paket `openssh`

Abhängigkeiten (2)

- Abhängigkeit von Paket meist (relativ) leicht lösbar:
  - Paket suchen und installieren
  - richtige Versionsnummer, Linux-Distribution (und -Version), Plattform beachten!
- Abhängigkeit von Datei (i.d.R. Bibliothek) oft schwieriger zu beheben
  - „Welches Paket enthält die Datei `x.y.so.1`?“

Abhängigkeiten (3)

- Installation bei **nicht erfüllten Abhängigkeiten** lässt sich trotzdem erzwingen:

```
[root@redhat:~]# rpm -i --nodeps paket.rpm
```

```
[root@debian:~]# dpkg -i --force-depends paket.deb
```

- Ratsam ist das aber nicht; Software wird meist nicht (oder nicht korrekt) arbeiten

Abhängigkeiten (4)

- Informationen zu Abhängigkeiten:

```
[root@redhat:~]# rpm -qp --requires
mtools-4.0.15-5.6.1.i586.rpm
/bin/sh
rpmlib(PayloadFilesHavePrefix) <= 4.0-1
rpmlib(CompressedFileNames) <= 3.0.4-1
libc.so.6
libc.so.6(GLIBC_2.0)
rpmlib(PayloadIsLzma) <= 4.4.6-1
...
```

-qp: query package

```
[root@debian:~]# dpkg -f nmap_4.62-1_i386.deb depends
libc6 (>= 2.7-1), libgcc1 (>= 1:4.1.1),
libpcap0.8 (>= 0.9.3-1), libpcre3 (>= 7.4),
libssl0.9.8 (>= 0.9.8f-5), libstdc++6 (>= 4.2.1)
```

-f: field

Achtung Syntax: Nach `-f` erst Paketname, dann Feldname `depends`

Konflikte (1)

- Zwei Pakete können miteinander in **Konflikt** stehen, d. h.: Es darf nur eines von beiden installiert sein.
- Konflikt tritt meist auf, wenn man versucht, ein neues Paket zu installieren, und bereits ein damit in Konflikt stehendes installiert ist.
- Beispiele:
  - zwei Mail-Server, zwei DHCP-Server (feste Ports)

Konflikte (2)

- Installation bei **Konflikten** lässt sich trotzdem erzwingen:

```
[root@redhat:~]# rpm -i --force paket.rpm
```

```
[root@debian:~]# dpkg -i --force-conflicts paket.deb
```

- Auch das Ignorieren von Konflikten ist nicht ratsam.

Repository-Ebene (1)

- Repository:
  - Sammlung von Software-Paketen
  - für genau eine Distributionsversion (z. B. OpenSuse 11.4 oder Ubuntu 10.10)
  - mit Inhaltsbeschreibungen aller Pakete
  - und Abhängigkeitsinformationen zu allen Paketen
- Installations-Tool kann Abhängigkeiten damit automatisch auflösen, nötige Zusatzpakete auswählen und diese gleich mit-installieren

Repository-Ebene (2)

- Paketverwaltung mit Repos:
  - Verwaltung der Repos
    - Eintragen neuer Repos und Löschen von Repos
    - Aktualisieren vorhandener Repos
  - „eigentliche“ Paketverwaltung
    - Installation, Update, Deinstallation von Paketen über die Repo-Tools

APT (1)

- Debian: APT
  - Advanced Packaging Tool
  - neben Debian auch von „Debian-basierten“ Distributionen genutzt, z. B. Knoppix, Ubuntu
  - Repository-Konfiguration in `/etc/apt/sources.list`
  - Paketverwaltung mit den Befehlen `apt-get` und `apt-cache`

APT (2)

- Beispiel-Konfigurationsdatei `/etc/apt/sources.list`:

```
deb cdrom:[Debian GNU/Linux 4.0]/ etch contrib main
deb http://ftp.gwdg.de/pub/linux/debian/debian/ etch main non-free
deb-src http://ftp.gwdg.de/pub/linux/debian/debian/ etch main
```

- Jede Zeile besteht aus vier Teilen:
  - `deb` oder `deb-src`: Handelt es sich um eine Binär- oder Source-Paketquelle?
  - URI zur Quelle, z. B. mit `http://`, `ftp://` oder `cdrom:[Volume-Name]`
  - Distributionsversion (`etch`, `sarge`, `squeeze` etc.; alternativ: `stable`, `unstable`, `testing`)
  - einzubindende (Unter-) Repositories: `main`, `contrib`, `non-free` etc.

APT (3)

- Neues Repo anlegen (Web):
  - Zusätzliche Zeile in `sources.list` eintragen
  - `apt-get update` ausführen

```
server:~ # apt-get update
Hole:1 http://security.debian.org etch/updates Release.gpg [189B]
Hole:2 http://security.debian.org etch/updates Release [22,5kB]
Hole:3 http://http.us.debian.org etch Release.gpg [378B]
Hole:4 http://http.us.debian.org etch Release [58,2kB]
Hole:5 http://security.debian.org etch/updates/main Packages [91,5kB]
Hole:6 http://http.us.debian.org etch/main Packages [4223kB]
Hole:7 http://security.debian.org etch/updates/contrib Packages [14B]
Hole:8 http://security.debian.org etch/updates/non-free Packages [14B]
Hole:9 http://http.us.debian.org etch/contrib Packages [50,6kB]
Hole:10 http://http.us.debian.org etch/non-free Packages [63,9kB]
Es wurden 4510kB in 15s geholt (291kB/s)
Paketlisten werden gelesen... Fertig
```

APT (4)

- Neues Repo (CD/DVD) anlegen: `apt-cdrom`

```
server:~ # apt-cdrom add
Verwendeter CD-ROM-Einbindungspunkt: /media/cdrom0/
Identifizieren ... [e1947a0c703f32d960fbfce6c7961521-2]
Durchsuchen des Mediums nach Index-Dateien ...
1 Paketindizes, 0 Quellindizes, 0 Übersetzungsindizes und 0
Signaturen gefunden
Dieses Medium heißt:
»Debian GNU/Linux 6.0.1a _Squeeze_ - Official i386 xfce+lxde-CD
Binary-1 20110322-15:11«
Reading Package Indexes... Fertig
Schreiben der neuen Quellliste
Quelllisteneinträge für dieses Medium sind:
deb cdrom:[Debian GNU/Linux 6.0.1a _Squeeze_ - Official i386
xfce+lxde-CD Binary-1 20110322-15:11]/ squeeze main
Wiederholen Sie dieses Prozedere für die restlichen Disks Ihres
Satzes.
```

→ erzeugt auch neuen Eintrag in `/etc/apt/sources.list`

APT (5)

- Pakete suchen: `apt-cache search`

```
server:~ # apt-cache search kmenu
deskmenu - A root menu for X11 window managers
kmenuedit - menu editor for KDE
oroborus - A lightweight themeable windowmanager for X
tastymenu - replacement K-menu for KDE/Kicker
```

- Pakete installieren: `apt-get install`

```
server:~ # apt-get install lynx
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  lynx-cur
The following NEW packages will be installed:
  lynx lynx-cur
0 upgraded, 2 newly installed, 0 to remove and 6 not upgraded.
Need to get 2040kB of archives.
After this operation, 4518kB of additional disk space will be used.
Do you want to continue [Y/n]? Y
```

Abhängigkeiten

```
root@debian:/home/esser# apt-get install k3b
Paketlisten werden gelesen... Fertig
Abhängigkeitsbaum wird aufgebaut
Statusinformationen werden eingelesen... Fertig
Die folgenden zusätzlichen Pakete werden installiert:
  cdparanoia cdrdao dvd+rw-tools exiv2 genisoimage k3b-data kaboom kdebase-runtime kdebase-runtime-data kdelibs-bin
  kdelibs5-data kdelibs5-plugins kdoctools libao-common libao4 libasyncns0 libattica0 libclucene0ldbl libeggdbus-1-0
  libexiv2-9 libflac++6 libgomp1 libgraphicsmagick3 libilmbase6 libiodbc2 libjack-jackd2-0 libk3b6 libk3b6-extracodecs
  libkcddb4 libkde3support4 libkdecore5 libkdesu5 libkdeui5 libkdnssd4 libkfile4 libkhtml5 libkio5 libkjsapi4
  libkjsembed4 libkmediaplayer4 libknewstuff2-4 libknewstuff3-4 libknotifyconfig4 libkntlm4 libkparts4 libkpty4
  libkrosscore4 libktexteditor4 libkutils4 libmusicbrainz4c2a libmysqlclient16 libnepomuk4 libnepomukquery4a libopenexr6
  libphonon4 libplasma3 libpolkit-agent-1-0 libpolkit-gobject-1-0 libpolkit-qt-1-0 libpulse-mainloop-glib0 libpulse0
  libqca2 libqt4-dbus libqt4-designer libqt4-network libqt4-opengl libqt4-qt3support libqt4-script libqt4-sql libqt4-sql-
  mysql libqt4-svg libqt4-webkit libqt4-xml libqt4-xmlpatterns libreadline5 libsamplerate0 libsndfile1 libsolid4
  libsoprano4 libssh-4 libstreamanalyzer0 libstreams0 libthreadweaver4 libvirtodbc0 libvorbisfile3 libwmf0.2-7 libxcb-
  shape0 libxine1 libxine1-bin libxine1-console libxine1-ffmpeg libxine1-misc-plugins libxine1-plugins libxine1-x
  libxml2-utils libxss1 mysql-common odbcinst odbcinst1debian2 oxygen-icon-theme phonon phonon-backend-xine plasma-
  scriptengine-javascript shared-desktop-ontologies soprano-daemon ttf-dejavu ttf-dejavu-extra vcdimager virtuoso-minimal
  virtuoso-opensource-6.1-bin virtuoso-opensource-6.1-common wodim xdg-utils
Die folgenden NEUEN Pakete werden installiert:
  cdparanoia cdrdao dvd+rw-tools exiv2 genisoimage k3b k3b-data kaboom kdebase-runtime kdebase-runtime-data kdelibs-bin
  kdelibs5-data kdelibs5-plugins kdoctools libao-common libao4 libasyncns0 libattica0 libclucene0ldbl libeggdbus-1-0
  libexiv2-9 libflac++6 libgomp1 libgraphicsmagick3 libilmbase6 libiodbc2 libjack-jackd2-0 libk3b6 libk3b6-extracodecs
  libkcddb4 libkde3support4 libkdecore5 libkdesu5 libkdeui5 libkdnssd4 libkfile4 libkhtml5 libkio5 libkjsapi4
  libkjsembed4 libkmediaplayer4 libknewstuff2-4 libknewstuff3-4 libknotifyconfig4 libkntlm4 libkparts4 libkpty4
  libkrosscore4 libktexteditor4 libkutils4 libmusicbrainz4c2a libmysqlclient16 libnepomuk4 libnepomukquery4a libopenexr6
  libphonon4 libplasma3 libpolkit-agent-1-0 libpolkit-gobject-1-0 libpolkit-qt-1-0 libpulse-mainloop-glib0 libpulse0
  libqca2 libqt4-dbus libqt4-designer libqt4-network libqt4-opengl libqt4-qt3support libqt4-script libqt4-sql libqt4-sql-
  mysql libqt4-svg libqt4-webkit libqt4-xml libqt4-xmlpatterns libreadline5 libsamplerate0 libsndfile1 libsolid4
  libsoprano4 libssh-4 libstreamanalyzer0 libstreams0 libthreadweaver4 libvirtodbc0 libvorbisfile3 libwmf0.2-7 libxcb-
  shape0 libxine1 libxine1-bin libxine1-console libxine1-ffmpeg libxine1-misc-plugins libxine1-plugins libxine1-x
  libxml2-utils libxss1 mysql-common odbcinst odbcinst1debian2 oxygen-icon-theme phonon phonon-backend-xine plasma-
  scriptengine-javascript shared-desktop-ontologies soprano-daemon ttf-dejavu ttf-dejavu-extra vcdimager virtuoso-minimal
  virtuoso-opensource-6.1-bin virtuoso-opensource-6.1-common wodim xdg-utils
0 aktualisiert, 114 neu installiert, 0 zu entfernen und 27 nicht aktualisiert.
Es müssen noch 96,2 MB von 99,6 MB an Archiven heruntergeladen werden.
Nach dieser Operation werden 238 MB Plattenplatz zusätzlich benutzt.
Möchten Sie fortfahren [J/n]? J
```

APT (7)

- Pakete entfernen

```
server:~ # apt-get remove mc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  mc
0 upgraded, 0 newly installed, 1 to remove and 6 not upgraded.
After this operation, 6402kB disk space will be freed.
Do you want to continue [Y/n]?
```

remove: mit Entfernen der Konfigurationsdateien (wie `dpkg -r`)

```
server:~ # apt-get purge paketname
```

purge: mit Entfernen der Konfigurationsdateien (wie `dpkg -P`)

APT (8)

- Alle Pakete aktualisieren: `apt-get upgrade`

```
server:~ # apt-get upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be upgraded:
  bind9-host dhcp3-client dhcp3-common dnsutils libbind9-50
  libdns58 libisc50 libiscccc50 libisccfg50 liblwres50
  libmozjs1d libnss3-1d libtiff4 mysql-common xulrunner-1.9
15 upgraded, 0 newly installed, 0 to remove and 6 not
upgraded.
Need to get 9825kB of archives.
After this operation, 24.6kB of additional disk space will
be used.
Do you want to continue [Y/n]? Y
Get:1 http://update.onlinehome-server.info lenny/updates/main
dhcp3-client 3.1.1-6+lenny5 [225kB]
Get:2 http://update.onlinehome-server.info lenny/updates/main
dhcp3-common 3.1.1-6+lenny5 [291kB]
[...]
```

APT (9)

- Umstieg auf neue Distributionsversion:
  - Repository-Einträge in sources.list auf neue Version umstellen
  - Dann `apt-get dist-upgrade` ausführen
  - Dabei werden evtl. einige Pakete entfernt, für die es in der neuen Version keinen Ersatz gibt

APT (10)

Übung:

- Tragen Sie die Zeile

  `deb http://ftp.debian.org/debian/ squeeze main`

  in die APT-Konfigurationsdatei ein (falls noch nicht vorhanden – nicht mit einem ähnlich aussehenden Eintrag für Updates verwechseln!)

- Aktualisieren Sie die Paketlisten
- Installieren Sie über die APT-Tools den grafischen Editor `nedit`

Hinweis zum Netzwerk: Wenn das Netzwerk nicht funktioniert, geben Sie in der Shell `su` (danach das root-Passwort), `killall dhclient` und `dhclient eth0` ein – danach sollte es gehen. Evtl. ist eine Anmeldung des Rechners im Browser (fom.de) nötig.

Repositories für RPM-Systeme

- In der „RPM-Welt“ gibt es kein einheitliches APT-ähnliches Tool für die Verwaltung von Repositories:
  - Fedora / Red Hat: YUM
  - Mandriva: URPMI
  - Suse: Zypper
- Wir betrachten hier nur YUM
- Prinzipien sind dieselben wie bei APT; aber Eintragen der Repos funktioniert anders

YUM (1)

- Red Hat / Fedora: YUM
  - Yellow Dog Updater, Modified
    - Yellow Dog Linux ist/war eine Linux-Distribution für PowerPC-Prozessoren, z. B. die alten Apple-Macs vor der Intel-Umstellung)
    - YUM war zunächst deren Paketmanager
  - Repository-Konfiguration: einzelne Dateien in `/etc/yum.repos.d/`
  - Paketverwaltung mit dem Befehl `yum`

YUM (2)

- Beispiel-Konfigurationsdatei für ein Repo:

```
[livna]
name=rpm.livna.org for $releasever - $basearch
#baseurl=http://rpm.livna.org/repo/$releasever/$basearch/ http://ftp-
stud.fht-esslingen.de/pub/Mirrors/rpm.livna.org/repo/$releasever/
$basearch/
mirrorlist=http://rpm.livna.org/mirrorlist
failovermethod=roundrobin
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-livna
```

  - anders als bei APT: diese Dateien nicht ändern
  - einfach in den Ordner `/etc/yum.repos.d/` kopieren

YUM (3)

- Neues Repo anlegen:
  - Repo-Anbieter stellt spezielle RPM-Datei zur Verfügung, welche das Repo automatisch einrichtet
  - Installation z. B. mit

    `rpm -i http://server/pfad/repopaket.rpm`

    möglich
- Repos anzeigen: `yum repolist all`

```
server:~ # yum repolist all
Loading "installonlyn" plugin
repo id              repo name                                status
development          Fedora - Development                     disabled
development-debuginf Fedora - Development - Debug             disabled
development-source   Fedora - Development - Source            disabled
fedora               Fedora 7 - i386                          enabled
fedora-debuginfo     Fedora 7 - i386 - Debug                  disabled
```

- Pakete suchen: `yum search`

```
server:~ # yum search mplayer
[...]
mplayer.i386  1.0-0.75.20070513svn.l livna
Matched from:
mplayer
MPlayer is a movie player that plays most MPEG, VOB, AVI, OGG/OGM,
VIVO, ASF/WMA/WMV, QT/MOV/MP4, FLI, RM, [...]
```

- Pakete installieren: `yum install`

```
server:~ # yum install w3m
Setting up Install Process
Parsing package install arguments
Resolving Dependencies
--> Running transaction check
---> Package w3m.i386 0:0.5.2-1.fc7 set to be updated
--> Processing Dependency: libgc.so.1 for package: w3m
--> Restarting Dependency Resolution with new changes.
[...]
```

- Pakete entfernen

```
server:~ # yum remove glibc
[...]
Transaction Summary
================================================================================
Install     0 Package(s)
Update      0 Package(s)
Remove    989 Package(s)

Is this ok [y/N]? n


Exiting on user command.
Complete!
```

  - YUM schlägt vor, auch alle abhängigen Pakete automatisch mit zu entfernen

YUM (6)

- Alle Pakete aktualisieren: `yum update` (vgl.: `apt-get upgrade`)
- gleicher Effekt wie bei APT
- Distributions-Upgrade:
  - RPM-Pakete mit Repo-Informationen der neuen Version einspielen, z. B.:

```
rpm -Uvh ftp://download.fedora.redhat.com/pub/fedora/linux/releases/8/Fedora/i386/os/Packages/fedora-release-8-3.noarch.rpm ftp://download.fedora.redhat.com/pub/fedora/linux/releases/8/Fedora/i386/os/Packages/fedora-release-notes-8.0.0-3.noarch.rpm
```

  - dann `yum upgrade` ausführen

Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz E
Topic 104: Dateisysteme
104.1 – 104.3, 104.6

Topic 104: Devices, Linux Filesystems, Filesystem Hierarchy Standard

104.1 Create partitions and filesystems

Description: Candidates should be able to configure disk partitions and then create filesystems on media such as hard disks. This includes the handling of swap partitions.

Key Knowledge Areas:

- Use various mkfs commands to set up partitions and create various filesystems such as: ext2, ext3, xfs, reiserfs v3, vfat

The following is a partial list of the used files, terms and utilities:
fdisk, mkfs, mkswap

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Topic 104: Devices, Linux Filesystems, Filesystem Hierarchy Standard

104.2 Maintain the integrity of filesystems

Description: Candidates should be able to maintain a standard filesystem, as well as the extra data associated with a journaling filesystem.

Key Knowledge Areas:

- Verify the integrity of filesystems.
- Monitor free space and inodes.
- Repair simple filesystem problems.

The following is a partial list of the used files, terms and utilities:
du, df, fsck, e2fsck, mke2fs, debugfs, dumpe2fs, tune2fs,
xfs tools (such as xfs_metadump and xfs_info)

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Topic 104: Devices, Linux Filesystems, Filesystem Hierarchy Standard

104.3 Control mounting and unmounting of filesystems

Description: Candidates should be able to configure the mounting of a filesystem.

Key Knowledge Areas:

- Manually mount and unmount filesystems.
- Configure filesystem mounting on bootup.
- Configure user mountable removeable filesystems.

The following is a partial list of the used files, terms and utilities:
/etc/fstab, /media, mount, umount

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Topic 104: Devices, Linux Filesystems, Filesystem Hierarchy Standard

104.6 Create and change hard and symbolic links

Description: Candidates should be able to create and manage hard and symbolic links to a file.

Key Knowledge Areas:

- Create links.
- Identify hard and/or softlinks.
- Copying versus linking files.
- Use links to support system administration tasks.

The following is a partial list of the used files, terms and utilities:
ln

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

104.1: Partitionen (1)

- Linux (und andere BS) unterteilen Festplatten in Partitionen
  - traditionell: vier Partitionen
    - Anfang, Ende, Größe: in Partitionstabelle im MBR (Master Boot Record)
    - Bezeichnung: **primäre Partitionen**
  - falls mehr nötig: eine der vier Partitionen zur **erweiterten** Partition machen
  - darin: **logische Partitionen**

104.1: Partitionen (2)

- Windows vergibt für jede (Windows)-Partition einen Laufwerksbuchstaben (`C:`, `D:` etc.)
  - unabhängig von Status primär/logisch
  - Reihenfolge kann wechseln
- Linux verwendet Bezeichnungen, die sich aus
  - Typ der Platte (IDE, SCSI)
  - Gerätenummer
  - Partitionsnummer

  zusammensetzen (`sda1` = SCSI disk a, part. 1)

104.1: Partitionen (3)

- Festplatten
  - `sda`, `sdb`, `sdc`, ...: SCSI und moderne SATA
  - `hda`, `hdb`, `hdc`, ...: klassische IDE
- Partitionen
  - 1, 2, 3, 4: primäre Partitionen
  - 5, 6, 7, ...: logische Partitionen (dann muss mind. eine der primären Part. eine erweiterte sein)
- Zugriff über Gerätedateien:
  - `sda3` → `/dev/sda3`

104.1: Partitionen (4)

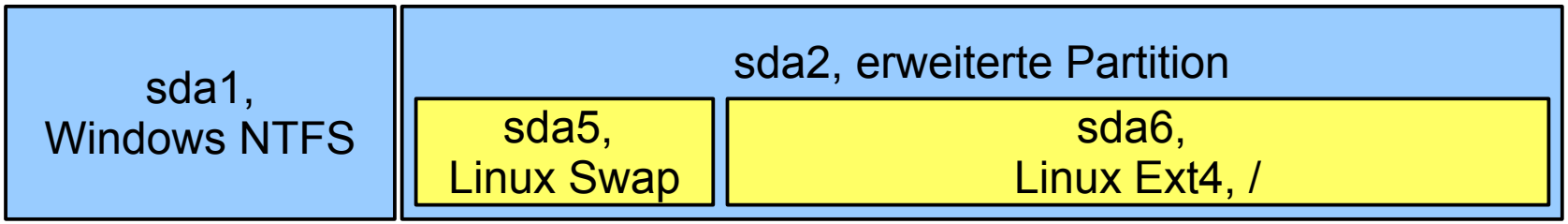
- Gerätedateien erzeugen moderne Linux-Versionen dynamisch:

```
esser@dissdevel:~$ ls -l /dev/sd*
brw-rw---- 1 root disk 8, 0  2. Jun 17:15 /dev/sda
brw-rw---- 1 root disk 8, 1  2. Jun 17:15 /dev/sda1
brw-rw---- 1 root disk 8, 2  2. Jun 17:15 /dev/sda2
brw-rw---- 1 root disk 8, 5  2. Jun 17:15 /dev/sda5
```

- in alten Linux-Versionen: große Mengen an passenden Gerätedateien statisch erzeugt

104.1: Partitionen (5)

- Typische Partitionierung



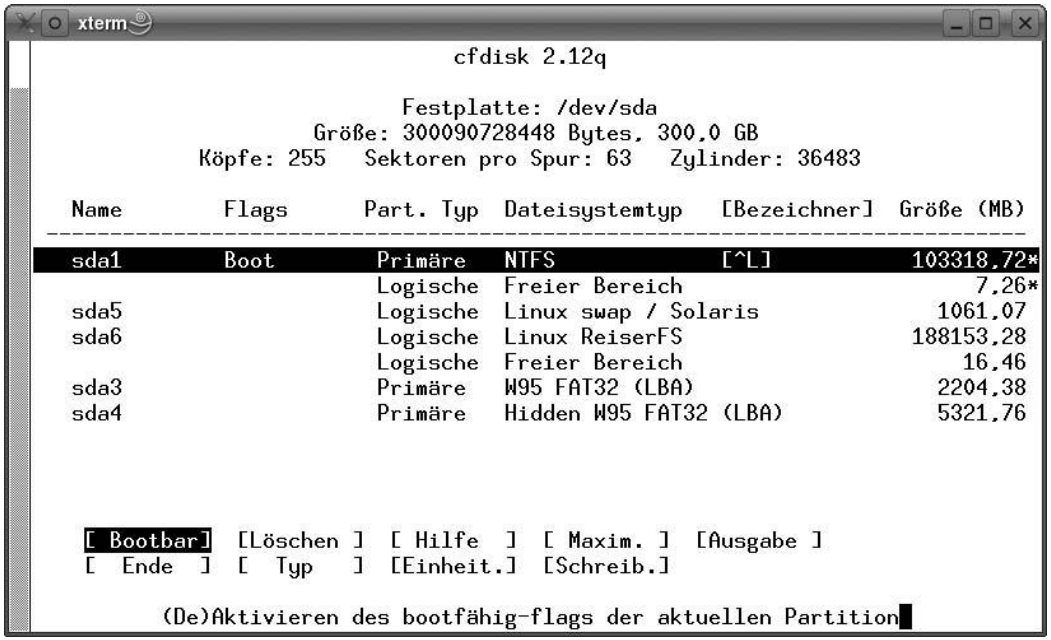
  - `sda1`: 1. primäre Partition: Windows, NTFS („Laufwerk C:“)
  - `sda2`: erweiterte Partition, enthält logische
  - `sda5`: 1. logische Partition: Linux, Swap
  - `sda6`: 2. logische Partition: Linux, Ext4

104.1: Partitionen (6)

- Arbeiten mit Gerätedateien
  - `head /dev/sda1`
    gibt Anfang der Partition `sda1` aus
  - `dd if=/dev/sda1 of=/tmp/image.dat`
    erzeugt 1:1-Kopie der Partition `sda1` in Datei,
    if=input file, of=output file
  - `fdisk /dev/sda`
    bearbeitet Partitionstabelle der Festplatte `sda`
  - `mkfs.ext3 /dev/sda7`
    formatiert Partition `sda7` mit Ext3-Dateisystem

104.1: Partitionen (7)

- Partitionieren unter Linux
  - `fdisk`: Standard-Tool
  - `cfdisk`: „grafisches“ Tool



  - `sfdisk`: für Skript-gesteuertes Partitionieren

104.1: fdisk (1)

Partitionsliste anzeigen

```
server:~# fdisk -l

Disk /dev/sda: 10.7 GB, 10694426624 bytes
255 heads, 63 sectors/track, 1300 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000ce798

   Device Boot   Start      End      Blocks   Id  System
/dev/sda1   *        1     1241     9965568   83  Linux
/dev/sda2         1241     1301      475137    5  Extended
/dev/sda5         1241     1301      475136   82  Linux swap / Solaris
```

Platte partitionieren

```
server:~# fdisk /dev/sda

Command (m for help): _
```

fdisk (2): Kommandoübersicht

- `p` – zeigt die Partitionstabelle (wie in `fdisk -l /dev/sda`).
- `n` – legt eine neue Partition an; fragt Partitionstyp, Nummer der Partition und Größe ab.
- `t` – Ändert den Typ einer Partition. Nach dem Aufruf des Kommandos erhalten Sie mit dem Kommando `L` eine Übersicht über die `fdisk` bekannten Partitionstypen.
- `d` – Löscht eine Partition.
- `w` – schreibt die von Ihnen überarbeitete Partitionstabelle. Danach beendet sich `fdisk`.
- `q` – Programm beendet sich, ohne die Partitionstabelle zu ändern.
- `m` – Menü, in dem alle Befehle aufgeführt sind, nur in Englisch und noch ein paar mehr.

104.1: fdisk (3)

Neue primäre Partition erzeugen

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (1241-1300, default 1241):
Using default value 1241
Last cylinder, +cylinders or +size{K,M,G} (1241-1300, default 1300):
Using default value 1300

Command (m for help): p

Disk /dev/sda: 10.7 GB, 10694426624 bytes
255 heads, 63 sectors/track, 1300 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000ce798

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1        1241     9965568   83  Linux
/dev/sda2            1241        1300      475658   83  Linux
```

104.1: fdisk (4)

Neue erweiterte Partition erzeugen

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
e
Partition number (1-4): 2
First cylinder (1241-1300, default 1241):
Using default value 1241
Last cylinder, +cylinders or +size{K,M,G} (1241-1300, default 1300):
Using default value 1300

Command (m for help): p

Disk /dev/sda: 10.7 GB, 10694426624 bytes
255 heads, 63 sectors/track, 1300 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000ce798

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1        1241     9965568   83  Linux
/dev/sda2            1241        1300      475658    5  Extended
```

104.1: fdisk (5)

Neue logische Partition erzeugen

```
Command (m for help): n
Command action
   l   logical (5 or over)
   p   primary partition (1-4)
l
First cylinder (1241-1300, default 1241):
Using default value 1241
Last cylinder, +cylinders or +size{K,M,G} (1241-1300, default 1300): 1260

Command (m for help): p

Disk /dev/sda: 10.7 GB, 10694426624 bytes
255 heads, 63 sectors/track, 1300 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000ce798

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1        1241     9965568   83  Linux
/dev/sda2            1241        1300      475658    5  Extended
/dev/sda5            1241        1260      154326+  83  Linux
```

Auswahl geändert, weil es jetzt eine erweiterte Partition gibt!

!

104.1: fdisk (6)

Partitionstypen

```
Command (m for help): t
Partition number (1-5): 1
Hex code (type L to list codes): L

 0  Empty           24  NEC DOS         81  Minix / old Lin bf  Solaris
 1  FAT12           39  Plan 9          82  Linux swap / So c1  DRDOS/sec (FAT-
 2  XENIX root      3c  PartitionMagic  83  Linux           c4  DRDOS/sec (FAT-
 3  XENIX usr       40  Venix 80286     84  OS/2 hidden C:  c6  DRDOS/sec (FAT-
 4  FAT16 <32M      41  PPC PReP Boot   85  Linux extended  c7  Syrinx
 5  Extended        42  SFS             86  NTFS volume set da  Non-FS data
 6  FAT16           4d  QNX4.x          87  NTFS volume set db  CP/M / CTOS / .
 7  HPFS/NTFS       4e  QNX4.x 2nd part 88  Linux plaintext de  Dell Utility
 8  AIX             4f  QNX4.x 3rd part 8e  Linux LVM       df  BootIt
 9  AIX bootable    50  OnTrack DM      93  Amoeba          e1  DOS access
 a  OS/2 Boot Manag 51  OnTrack DM6 Aux 94  Amoeba BBT      e3  DOS R/O
 b  W95 FAT32       52  CP/M            9f  BSD/OS          e4  SpeedStor
 c  W95 FAT32 (LBA) 53  OnTrack DM6 Aux a0  IBM Thinkpad hi eb  BeOS fs
 e  W95 FAT16 (LBA) 54  OnTrackDM6      a5  FreeBSD         ee  GPT
 f  W95 Ext'd (LBA) 55  EZ-Drive        a6  OpenBSD         ef  EFI (FAT-12/16/
10  OPUS            56  Golden Bow      a7  NeXTSTEP        f0  Linux/PA-RISC b
11  Hidden FAT12    5c  Priam Edisk     a8  Darwin UFS      f1  SpeedStor
12  Compaq diagnost 61  SpeedStor       a9  NetBSD          f4  SpeedStor
14  Hidden FAT16 <3 63  GNU HURD or Sys ab  Darwin boot     f2  DOS secondary
16  Hidden FAT16    64  Novell Netware  af  HFS / HFS+      fb  VMware VMFS
17  Hidden HPFS/NTF 65  Novell Netware  b7  BSDI fs         fc  VMware VMKCORE
18  AST SmartSleep  70  DiskSecure Mult b8  BSDI swap       fd  Linux raid auto
1b  Hidden W95 FAT3 75  PC/IX           bb  Boot Wizard hid fe  LANstep
1c  Hidden W95 FAT3 80  Old Minix       be  Solaris boot    ff  BBT
1e  Hidden W95 FAT1
```

104.1: Formatieren (1)

- Einfaches Anlegen einer neuen Partition macht diese noch nicht benutzbar
- Partition muss man vor erster Nutzung **formatieren** (= mit einem **Dateisystem** versehen)
- Kommando allgemein: `mkfs` (make filesystem)
  - `mkfs -t` *`TYP`* `/dev/GERÄT`
  - ruft spezialisiertes Tool `mkfs.`*`TYP`* (z. B. `mkfs.ext3`) auf

```
root@dissdevel:/# ls /sbin/mkfs*
/sbin/mkfs          /sbin/mkfs.ext2   /sbin/mkfs.ext4dev   /sbin/mkfs.ntfs
/sbin/mkfs.bfs      /sbin/mkfs.ext3   /sbin/mkfs.minix     /sbin/mkfs.vfat
/sbin/mkfs.cramfs   /sbin/mkfs.ext4   /sbin/mkfs.msdos
```

104.1: Formatieren (2)

```
root@dissdevel:/# mkfs -t ext3 /dev/sda8
mke2fs 1.41.12 (17-May-2010)
Dateisystem-Label=
OS-Typ: Linux
Blockgröße=1024 (log=0)
Fragmentgröße=1024 (log=0)
Stride=0 Blöcke, Stripebreite=0 Blöcke
2560 Inodes, 10240 Blöcke
512 Blöcke (5.00%) reserviert für den Superuser
Erster Datenblock=1
Maximale Dateisystem-Blöcke=10485760
2 Blockgruppen
8192 Blöcke pro Gruppe, 8192 Fragmente pro Gruppe
1280 Inodes pro Gruppe
Superblock-Sicherungskopien gespeichert in den Blöcken:
    8193

Schreibe Inode-Tabellen: erledigt
Erstelle Journal (1024 Blöcke): erledigt
Schreibe Superblöcke und Dateisystem-Accountinginformationen: erledigt

Das Dateisystem wird automatisch nach jeweils 30 Einhäng-Vorgängen bzw.
alle 180 Tage überprüft, je nachdem, was zuerst eintritt. Dies kann durch
tune2fs -c oder -i geändert werden.
```

104.1: Formatieren (3)

- Auch **Swap-Partition** (Bereich, der für das Auslagern von Speicherseiten verwendet wird; → **Paging**) muss formatiert werden
- Tool heißt `mkswap`:

```
root@dissdevel:/# mkswap /dev/sda5
Setting up swapspace version 1, size = 475132 KiB
no label, UUID=5c43f2b7-8801-4fde-94a2-f154ffbabb42
```

- Swap-Bereich darf auch Datei sein
  → hilfreich, wenn keine Swap-Partition angelegt werden kann

104.3: Mounten (1)

- Linux bindet beim Systemstart nicht automatisch alle Dateisysteme (meist: Partitionen) ein, sondern tut dies nur für eine Auswahl, die durch Einträge in einer Konfigurationsdatei festgelegt wird. Ausnahme: Root-Dateisystem /, ohne das kein Systemstart möglich ist.
- Den Einbindevorgang nennt Linux (wie alle Unix-Systeme) **mounten**, die umgekehrte Operation, bei der das System nicht länger auf einen Datenträger zugreift, heißt **unmounten**.
- Die dafür zuständigen Kommandos heißen `mount` und `umount` (nicht u**n**mount!)
- Automatisches Mounten über Einträge in `/etc/fstab`

104.3: Mounten (2)

- Das Mounten stellt eine Verknüpfung zwischen einem Datenträger und einem Verzeichnis her, unter dem dann die Inhalte des Datenträgers erreichbar sind
- Diese Verzeichnisse (**Mount-Points**) sind das Gegenstück zu Windows-Laufwerksbuchstaben
- Linux- (Unix-) Ansatz ist flexibler

104.3: Mounten (3)

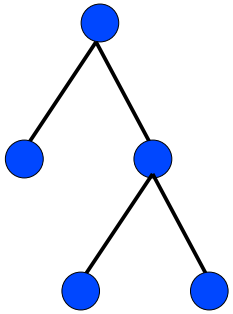
Datenträger unter Windows und Linux

Linux-Partition:	[Root-Dateisystem /dev/sda6 auf /]
nicht sichtbar	/home
	/usr
	/etc
	/var
	...
C: [Win]	[/dev/sda1 auf /mnt/win1]
C:\Windows	/mnt/win1/Windows
C:\Windows\System	/mnt/win1/Windows/System
C:\Users	/mnt/win1/Users
C:\Users\Esser	/mnt/win1/Users/Esser
C:\Users\Esser\Documents	/mnt/win1/Users/Esser/Documents
D: [Restore]	[/dev/sda2 auf /mnt/win2]
D:\Restore.Tmp	/mnt/win2/Restore.Tmp
E: [OfficeDVD]	[Office-DVD auf /media/OfficeDVD]
E:\Files	/media/OfficeDVD/Files

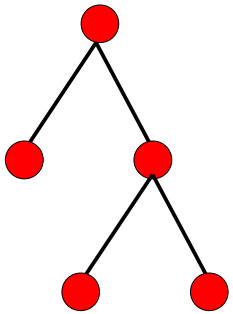
104.3: Mounten (4)

Windows

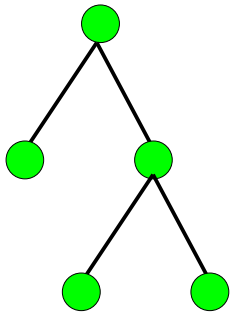
C:



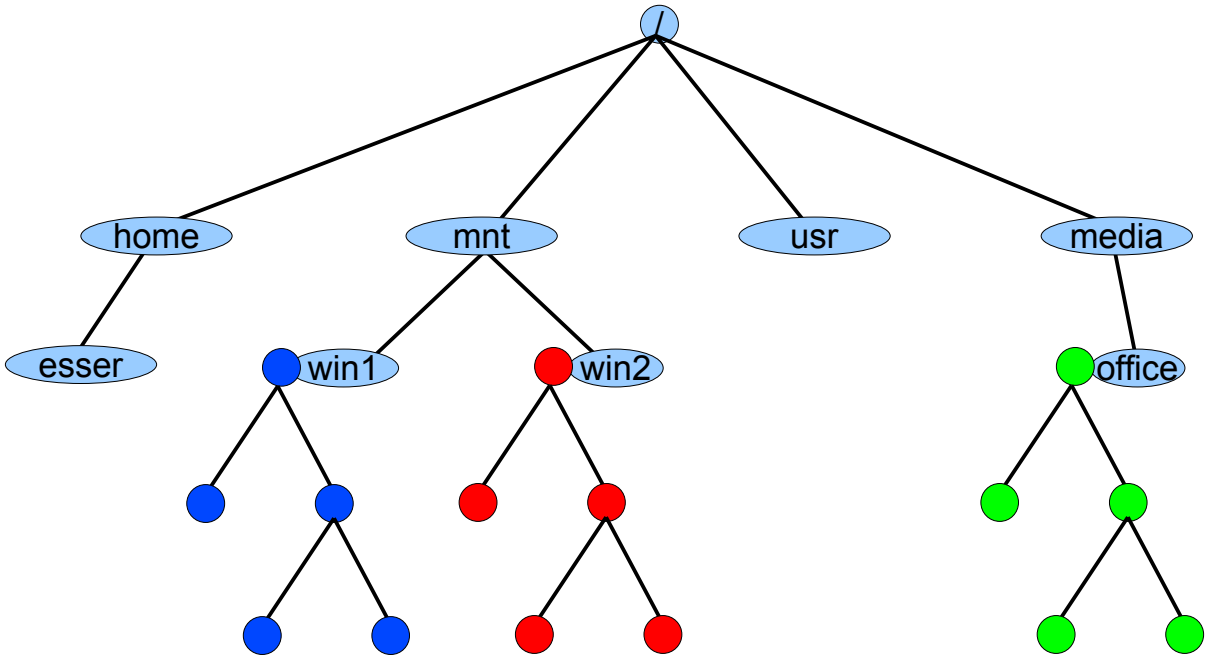
D:



E:



Linux



104.3: Mounten (5)

- Was braucht man fürs Mounten?
  - Gerätedatei des Datenträgers (Partition o. ä.)
  - Mount-Point (Verzeichnis, muss schon existieren)
  - evtl. Typ des Dateisystems
  - evtl. Optionen fürs Mounten

```
mount
   -t TYP                -o OPTIONS
   /dev/PARTITION   /MOUNTPOINT
```

```
mount -t ext3 -o ro /dev/sda7 /mnt
```

104.3: Mounten (6)

- Dateisystemtyp (`-t` *TYP*)
  - `ext4`: 4th extended filesystem (Linux, aktuell)
  - `ext3`: 3rd extended filesystem (Linux, älter)
  - `ext2`: 2nd extended filesystem (Linux, veraltet)
  - `reiserfs`: Reiser-Dateisystem (Linux, älter)
  - `ntfs`: New Technology Filesystem (Windows)
  - `vfat`: Virtual File Allocation Table (DOS, Windows)
  - `iso9660`: CD-/DVD-Dateisystem
  - `udf`: DVD-Dateisystem (z. B. Video-DVD)

104.3: Mounten (7)

- Dateisystemtyp (`-t` *TYP*)
  - Liste tatsächlich noch länger; Auszug aus Manpage:

```
-t, --types vfstype
  The argument following the -t is used to indicate the filesystem type.  The
  filesystem types which are currently supported include: adfs, affs, autofs,
  cifs,  coda, coherent, cramfs, debugfs, devpts, efs, ext, ext2, ext3, ext4,
  hfs, hfsplus, hpfs, iso9660, jfs, minix, msdos,  ncpfs,  nfs,  nfs4,  ntfs,
  proc,  qnx4,  ramfs,  reiserfs, romfs, squashfs, smbfs, sysv, tmpfs, ubifs,
  udf, ufs, umsdos, usbfs, vfat, xenix, xfs, xiafs.
```

  - Welche Dateisysteme unterstützt der Kernel (im Moment)?

```
root@dissdevel:~# grep -v nodev /proc/filesystems
    ext3
    fuseblk
    udf
    iso9660
    ntfs
    vfat
```

104.3: Mounten (8)

- Mount-Optionen (`-o` *`OPTIONEN`*); Auswahl:
  - `ro`: read-only (nur lesen)
  - `rw`: read-write (lesen und schreiben; Standard)
  - `async`, `sync`: alle Zugriffe asynchron bzw. synchron (sofort schreiben, kein Puffer) ausführen
  - `noatime`: Zugriffe auf Dateien nicht in Metadaten speichern (u. a. für Flash-Datenträger sinnvoll)
  - `nodiratime`: wie `noatime`, für Verzeichnisse
  - `noexec`: Programme sind nicht ausführbar
  - `remount`: bereits gemountetes FS nochmal mounten
  - `loop`: Dateisystem-Image mounten

104.3: Mounten (9)

- Swap-Partitionen werden nicht gemountet, sondern aktiviert (`swapon`) oder deaktiviert (`swapoff`)

```
root@dissdevel:/# swapon -v /dev/sda5
swapon on /dev/sda5
swapon: /dev/sda5: found swap signature: version 1, page-size 4,
        same byte order
swapon: /dev/sda5: pagesize=4096, swapsize=486539264,
        devsize=486539264

root@dissdevel:/# swapoff -v /dev/sda5
swapoff on /dev/sda5
```

  - (ohne Option `-v` keine Ausgabe)
  - Swap darf auch eine Datei sein

104.3: Mounten (10)

- Übersicht über aktive Swap-Bereiche

```
root@dissdevel:/# cat /proc/swaps
Filename        Type         Size     Used  Priority
/dev/sda5       partition    475128   0     -1
/tmp/swapfile   file         10232    0     -2
```

104.3: Unmounten (1)

- Dateisystem wieder aushängen (unmounten)
  - Kommando `umount`
  - Argument: Wahlweise Name der Gerätedatei (`/dev/...`) oder Mount-Point
  - Beispiele:
    `umount /dev/sda6` (Gerätedatei)
    `umount /mnt/win1` (Mount-Point)

104.3: Unmounten (2)

- Kommando `umount` schlägt manchmal fehl:

```
root@dissdevel:/mnt/tmp# pwd
/mnt/tmp
root@dissdevel:/mnt/tmp# umount /mnt
umount: /mnt: device is busy.
        (In some cases useful info about processes that use
         the device is found by lsof(8) or fuser(1))
root@dissdevel:/mnt/tmp# cd /
root@dissdevel:/# umount /mnt/
root@dissdevel:/# _
```

  - Es darf keine Datei im FS geöffnet sein
  - Es darf keine Shell (oder ein anderes Programm) das aktuelle Arbeitsverzeichnis in diesem FS haben

104.3: Mounten mit /etc/fstab (1)

- Konfigurationsdatei `/etc/fstab` (**file**s**ystem** **tab**le) legt fest, welche FS beim Systemstart eingebunden werden
  - Aufbau einer Zeile der Datei:

```
# <fs>      <mount point>   <type>      <options>          <dump>  <pass>
```

  - Beispiel:

```
proc         /proc           proc        defaults           0       0
/dev/sda1    /               ext3        errors=remount-ro  0       1
/dev/sda5    none            swap        sw                 0       0
/dev/scd0    /media/cdrom0   udf,iso9660 user,noauto         0       0
```

104.3: Mounten mit /etc/fstab (2)

- Einige Einträge haben im Optionenfeld die Option `noauto`
- Solche Einträge werden nicht automatisch gemountet, können aber einfacher von Hand gemountet werden

```
root@server:~# grep scd0 /etc/fstab
/dev/scd0   /media/cdrom   udf,iso9660  user,noauto    0    0

root@server:~# mount /media/cdrom
```

- Zusatzoption `user` bedeutet: Mounten auch ohne Root-Rechte möglich

104.3: Mounten mit /etc/fstab (3)

- Neben `/etc/fstab` gibt es noch eine Datei `/etc/mtab` (**mount table**)
- Diese enthält Informationen über gemountete Dateisysteme und wird automatisch (vom System) erstellt und aktualisiert

```
root@dissdevel:/# cat /etc/mtab
/dev/sda1 / ext3 rw,errors=remount-ro 0 0
tmpfs /lib/init/rw tmpfs rw,nosuid,mode=0755 0 0
proc /proc proc rw,noexec,nosuid,nodev 0 0
sysfs /sys sysfs rw,noexec,nosuid,nodev 0 0
udev /dev tmpfs rw,mode=0755 0 0
tmpfs /dev/shm tmpfs rw,nosuid,nodev 0 0
devpts /dev/pts devpts rw,noexec,nosuid,gid=5,mode=620 0 0
fusectl /sys/fs/fuse/connections fusectl rw 0 0
Daten /media/sf_Daten vboxsf gid=1001,rw 0 0
```

104.2: Filesystem Check (1)

- Dateisysteme werden i. d. R. beim Systemstart auf Konsistenz überprüft (filesystem check)
- Auf Wunsch auch manuelle Überprüfung möglich
- Dateisystem darf dabei nicht gemountet sein
- Generisches Tool: `fsck` (**f**ile**s**ystem **c**hec**k**)

```
root@dissdevel:/# fsck /dev/sda1
fsck from util-linux-ng 2.17.2
e2fsck 1.41.12 (17-May-2010)
/dev/sda1 ist eingehängt.

WARNUNG!!!  Die Benutzung von e2fsck auf einem eingehängten
Dateisystem führt zu SCHWERWIEGENDEN SCHÄDEN im Dateisystem.

Wirklich fortfahren (j/n)?
```

104.2: Filesystem Check (2)

- Automatische Überprüfung beim Systemstart:

```
Activating swap...done.
Checking root file system...fsck from util-linux-ng 2.17.2
/dev/sda1 contains a file system with errors, check forced.
/dev/sda1: 187822/623392 files (0.8% non-contiguous), 1128272/2491392 blocks
done.
Loading kernel modules...done.
Cleaning up ifupdown....
Setting up networking....
Activating lvm and md swap...done.
Checking file systems...fsck from util-linux-ng 2.17.2
done.
Mounting local filesystems...done.
```

  - Welche Dateisysteme überprüft werden, legt letzte Spalte in `/etc/fstab` fest: `1` = prüfen

```
proc          /proc          proc          defaults           0       0
/dev/sda1     /              ext3          errors=remount-ro  0       1
/dev/sda5     none           swap          sw                 0       0
/dev/scd0     /media/cdrom0  udf,iso9660   user,noauto        0       0
```

104.2: Filesystem Check (3)

- Statt `fsck` besser direkt das für das Dateisystem passende Tool (`fsck.`*TYP*) aufrufen
  → dann sind auch individuelle Optionen möglich
- Beispiel `fsck.ext3`, Optionen:
  - `-f` : force, auch als „clean“ erkanntes FS prüfen
  - `-p` : versuche, Fehler automatisch zu beheben
  - `-y` : alle Fragen, die `fsck.ext3` stellt, automatisch mit „y“ (yes) beantworten
  - `-c` : Programm `badblocks` aufrufen (findet defekte Blöcke und trägt diese in Bad Blocks List ein)

104.2: Filesystem Check (4)

- Beispiel `fsck` auf Ext3-Dateisystem

```
root@dissdevel:/# fsck /dev/sda8
fsck from util-linux-ng 2.17.2
e2fsck 1.41.12 (17-May-2010)
/dev/sda8: sauber, 11/65536 Dateien, 12644/262144 Blöcke
```

(jetzt mit `-f` erzwingen)

```
root@dissdevel:/# fsck -f /dev/sda8
fsck from util-linux-ng 2.17.2
e2fsck 1.41.12 (17-May-2010)
Durchgang 1: Prüfe Inodes, Blocks, und Größen
Durchgang 2: Prüfe Verzeichnis Struktur
Durchgang 3: Prüfe Verzeichnis Verknüpfungen
Durchgang 4: Überprüfe die Referenzzähler
Durchgang 5: Überprüfe Gruppe Zusammenfassung
/dev/sda8: 11/65536 Dateien (0.0% nicht zusammenhängend), 12644/262144
Blöcke
```

104.2: Filesystem Check (5)

- Beispiel `fsck.ext3` – mit Fehlern

```
root@dissdevel:/home/esser# fsck.ext3 -f /dev/sda8
e2fsck 1.41.12 (17-May-2010)
Durchgang 1: Prüfe Inodes, Blocks, und Größen
Durchgang 2: Prüfe Verzeichnis Struktur
Eintrag »..« in ??? (41972) hat gelöscht/unbenutzt Inode 19152.  Bereinige<j>? ja
Eintrag »..« in ??? (42004) hat gelöscht/unbenutzt Inode 19167.  Bereinige<j>? ja
Eintrag »..« in ??? (42006) hat gelöscht/unbenutzt Inode 19167.  Bereinige<j>? ja
Durchgang 3: Prüfe Verzeichnis Verknüpfungen
Durchgang 4: Überprüfe die Referenzzähler
Durchgang 5: Überprüfe Gruppe Zusammenfassung

Die Anzahl freier Inodes ist falsch (59759, gezählt=58271).
Repariere<j>? ja


/dev/sda8: ***** DATEISYSTEM WURDE VERÄNDERT *****
/dev/sda8: 7265/65536 Dateien (0.0% nicht zusammenhängend), 44392/262144 Blöcke
```

Alternativnamen mkfs, fsck

- Die FS-spezifischen `mkfs`- und `fsck`-Tools sind meist noch unter anderen (kürzeren) Namen erreichbar:
  - `mkfs.ext3  = mke2fs` `fsck.ext3  = e2fsck`
  - `mkfs.ext4  = mke2fs` `fsck.ext4  = e2fsck`
  - `mkfs.vfat  = mkdosfs` `fsck.vfat  = dosfsck`
  - `mkfs.msdos = mkdosfs` `fsck.msdos = dosfsck`
- Aber: dann bei `mk*fs` aufpassen, welches das Standard-FS ist (`mke2fs`: Ext2, also nicht sinnvoll...)
- `mkfs` ohne `-t`: auch Ext2
- `vfat` und `msdos` sind identische FS

104.2: FS-Informationen, du/df (1)

- Speicherplatz-Verbrauch
  - `df` (**d**isk **f**ree) zeigt freien Platz auf einem Datenträger (oder auf allen) an
  - `du` (**d**isk **u**sage) zeigt verwendeten Platz in einem Verzeichnis an
  - für beide Tools: mit Optionen die Ausgabe anpassen

104.2: FS-Informationen, du/df (2)

- df

```
root@dissdevel:/tmp# df
Dateisystem           1K  Blöcke   Benutzt Verfügbar Ben% Eingehängt auf
/dev/sda5               9809032   6446020   2864736  70% /
/dev/sda1             118022124  87966344  30055780  75% /mnt/win
tmpfs                    517240         0    517240   0% /lib/init/rw
udev                     512884       176    512708   1% /dev
tmpfs                    517240         0    517240   0% /dev/shm
```

-h = „human-readable“

```
root@dissdevel:/tmp# df -h
Dateisystem            Size  Used Avail Use% Eingehängt auf
/dev/sda5              9,4G  6,2G  2,8G  70% /
/dev/sda1              113G   84G   29G  75% /mnt/win
tmpfs                  506M     0  506M   0% /lib/init/rw
udev                   501M  176K  501M   1% /dev
tmpfs                  506M     0  506M   0% /dev/shm

root@dissdevel:/tmp# df -h /
Dateisystem            Size  Used Avail Use% Eingehängt auf
/dev/sda1              9,4G  6,2G  2,8G  70% /
```

104.2: FS-Informationen, du/df (3)

- du

```
esser@dissdevel:~/Daten/FOM$ du
80  ./Briefe
7300    ./BS-Alt
60324   ./BS-Praxis
20184   ./BS-Theorie/Klausur
20  ./BS-Theorie/Uebung02-Loesungen/aufgabe-b
20  ./BS-Theorie/Uebung02-Loesungen/aufgabe-c
20  ./BS-Theorie/Uebung02-Loesungen/aufgabe-d
88  ./BS-Theorie/Uebung02-Loesungen
45492   ./BS-Theorie
440 ./IT-Infrastruktur
4780    ./Material und Downloads/FOM_IT-Infrastruktur_(REP)_510-r.15_sw
31920   ./Material und Downloads
2648    ./Seminar/bearbeitet
4196    ./Seminar
149812  .

esser@dissdevel:~/Daten/FOM$ du -s
149812  .

esser@dissdevel:~/Daten/FOM$ du -sm
147 .
```

-s = summary,
-m = megabytes

104.2: FS-Informationen, du/df (4)

- du -s * | sort -n

```
esser@dissdevel:~/Daten$ ls -d *
Anstel  Buecher  Erlangen  FOM  FU-Hagen  Heise  HM  LNM  privat
Promotion

esser@dissdevel:~/Daten$ du -sm * | sort -n
1   privat
3   Heise
6   Anstel
6   Buecher
9   Erlangen
15  HM
60  FU-Hagen
61  LNM
147 FOM
1715    Promotion
```

sort -n = numerisch sortieren

104.2: debugfs, dumpe2fs, tune2fs (1)

- Arbeiten am Dateisystem (für Fortgeschrittene)
- Tools für die Familie der Ext-Dateisysteme (Ext2, Ext3, Ext4)
  - `debugfs`: Eingriffe in die „Interna“ des Dateisystems
  - `dumpe2fs`: Ausgabe aller wichtigen Metadaten des Dateisystems
  - `tune2fs`: „Tuning“ für Ext-Dateisysteme, Einstellen von Optionen

104.2: debugfs, dumpe2fs, tune2fs (2)

```
root@dissdevel:/tmp# dumpe2fs /dev/sda1
dumpe2fs 1.41.12 (17-May-2010)
Filesystem volume name:   <none>
Last mounted on:          <not available>
Filesystem UUID:          27a7303b-9479-47ea-8ae8-67f9b6206920
Filesystem magic number:  0xEF53
Filesystem revision #:    1 (dynamic)
Filesystem features:      has_journal ext_attr resize_inode dir_index filetype needs_recovery
                          sparse_super large_file
Filesystem flags:         signed_directory_hash
Default mount options:    (none)
Filesystem state:         clean
Errors behavior:          Continue
Filesystem OS type:       Linux
Inode count:              623392
Block count:              2491392
Reserved block count:     124569
Free blocks:              1363120
Free inodes:              435570
First block:              0
Block size:               4096
Fragment size:            4096
Reserved GDT blocks:      608
Blocks per group:         32768
Fragments per group:      32768
Inodes per group:         8096
Inode blocks per group:   506
Filesystem created:       Tue May  3 11:55:19 2011
Last mount time:          Thu Jun  2 17:15:51 2011
Last write time:          Thu Jun  2 17:15:41 2011
Mount count:              1
Maximum mount count:      20
Last checked:             Thu Jun  2 17:15:41 2011
Check interval:           15552000 (6 months)
Next check after:         Tue Nov 29 16:15:41 2011
...
```

104.2: debugfs, dumpe2fs, tune2fs (3)

- `tune2fs`
  - Einstellen, was bei FS-Fehler passiert (continue, panic, remount-ro)
  - Intervall zwischen FS-Checks ändern
  - **Journal** ergänzen oder entfernen (→ Journaling, nächste Folie)
  - **Volume-Label** ändern
  - Größe des **reservierten Bereichs** ändern
    - dieser Teil des FS kann nur von root verwendet werden
    - für normale Nutzer erscheint das FS ggf. als voll

Journaling

- Moderne Dateisysteme (z. B. ext3, ext4, ReiserFS) verwenden **Journaling**
  - Vor jeder Änderung an den **Metadaten** einer Datei wird in einen Protokollbereich (das **Journal**) die geplante Änderung geschrieben
  - Ist Änderung erfolgreich abgeschlossen, wird Eintrag aus Journal wieder gelöscht
- Beschleunigt (nach Absturz) den FS-Check:
  - nur prüfen, welche Einträge im Journal stehen – diese wurden evtl. nicht erfolgreich durchgeführt
- Variante: nicht nur Metadaten, sondern auch Daten

Etwas Theorie (1)

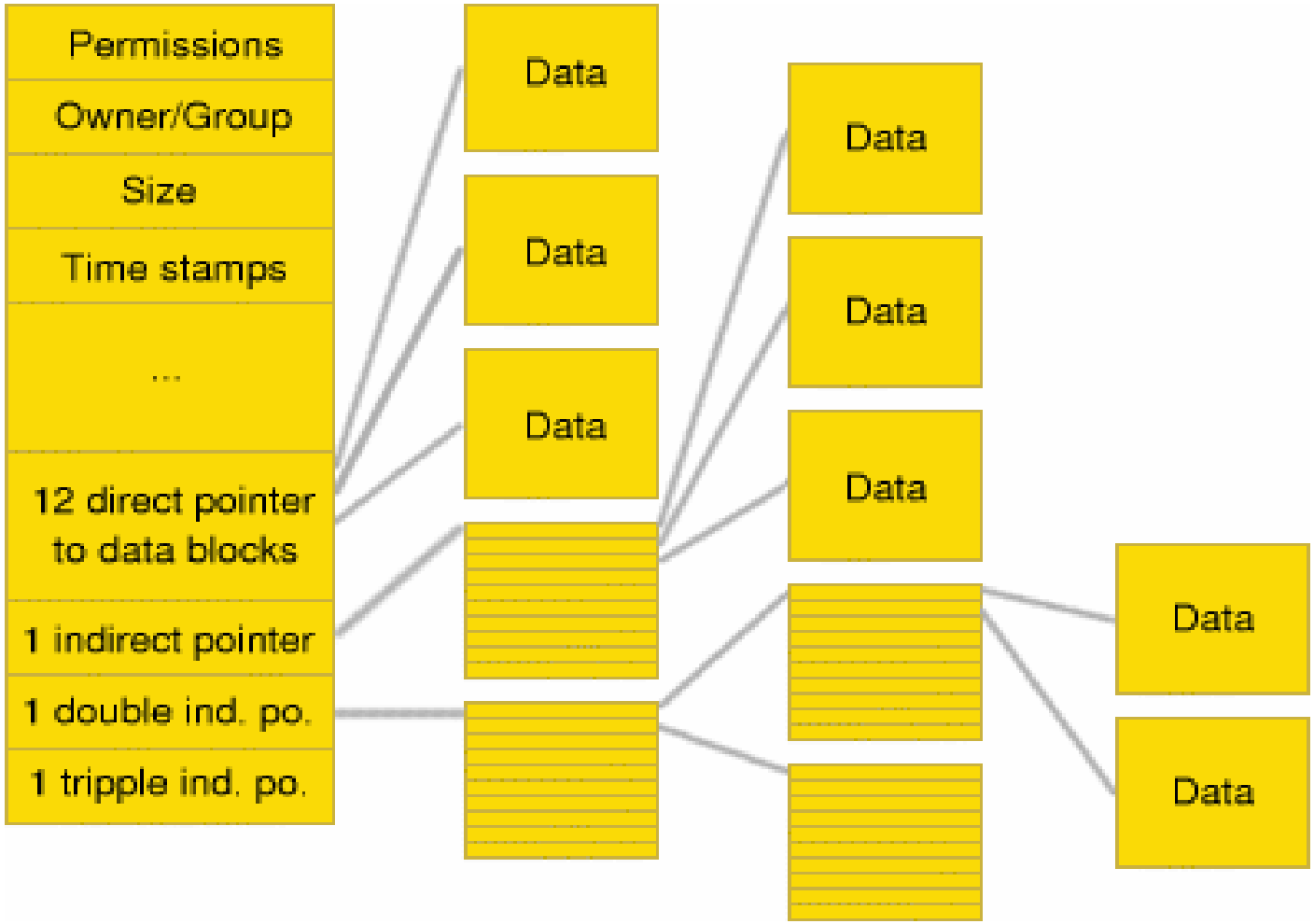
- Wichtige Konzepte in Linux-Dateisystemen:
  - I-Nodes
  - Dateien und Verzeichnisse
  - Datenblöcke

Etwas Theorie (2)

- I-Nodes:
  - Wenn eine neue Datei angelegt wird, sucht Linux zunächst einen freien **I-Node** (Index Node) – das ist ein Verwaltungseintrag auf der Partition
  - I-Node enthält Metadaten:
    - Dateigröße, Liste der verwendeten Blöcke
    - Besitzer und Standard-Gruppe
    - Zugriffsrechte, Timestamps ()
    - **nicht im I-Node: Dateiname und/oder Pfad (!)**
  - Danach zu I-Node Eintrag in Verzeichnis anlegen

Etwas Theorie (3)

- I-Node – grafisch:



Quelle: http://de.linwiki.org/wiki/Linuxfibel_-_System-Administration_-_Dateisysteme

Etwas Theorie (4)

- Dateien
  - Eine Datei besteht „klassisch“ aus
    - den eigentlichen Nutzdaten, die in Datenblöcken gespeichert sind,
    - einem Dateinamen (mit Pfadangabe)
    - Metadaten (Besitzer, Zugriffsrechte, Größe etc.)
  - Aus Linux-Sicht ist eine Datei zunächst die Sammlung der Datenblöcke + der I-Node (mit Metadaten und Blockliste)
  - Durch Eintragen in ein Verzeichnis (also Zuordnung: Dateiname → I-Node) wird die Datei im Dateisystem sichtbar

Etwas Theorie (5)

- Verzeichnisse
  - … sind in Linux-Dateisystemen spezielle Dateien, welche nur Zuordnungen Name → I-Node enthalten
  - entspricht der Unix-Philosophie „alles ist eine Datei“
  - Da Verzeichnis nur eine Datei ist, ist auch ein schnelles Verschieben eines kompletten Ordners mit Unterordnern schnell erledigt:

    ```
    mv /home/esser/Videos /tmp/Videos
    ```

    benötigt keine messbare Zeit (falls Verschieben innerhalb einer Partition!)

Etwas Theorie (6)

- Datenblöcke
  - Dateisystem verwaltet eine Liste freier / belegter Datenblöcke
  - Beim Löschen einer Datei werden alle verwendeten Datenblöcke als „frei“ gekennzeichnet (und bald wiederverwendet)

104.6: Soft Links / Hard Links (1)

- Grundidee hinter Links: Datei unter mehreren Namen (und ggf. an verschiedenen Orten) ansprechen
  - **symbolische Links** (**soft links**): spezielle Dateien, die den Pfad (absolut oder relativ) zu einer anderen Datei speichern

    → können „broken“ sein, also auf etwas zeigen, das es nicht gibt (wie im Web: broken link)
  - **Hard Links**: Eintrag in einem Verzeichnis, der auf denselben I-Node zeigt

104.6: Soft Links / Hard Links (2)

- Symbolische Links / Soft Links
  - erstellen mit `ln -s` (s = soft)
  - funktionieren auch Dateisystem-übergreifen (wenn anderes FS auch eingebunden ist)

```
esser@dissdevel:~$ ls -l /mnt/windows/config.sys
-rwxr-xr-x 1 root root 36  2. Jun 20:08 /mnt/windows/config.sys
esser@dissdevel:~$ ln -s /mnt/windows/config.sys config.sys
esser@dissdevel:~$ ls -l config.sys
lrwxrwxrwx 1 esser esser 31  2. Jun 20:08 config.sys -> /mnt/windows/config.sys
esser@dissdevel:~$ ln -s /mnt/windows/BROKEN broken.txt
esser@dissdevel:~$ ls -l broken.txt
lrwxrwxrwx 1 esser esser 27  2. Jun 20:09 broken.txt -> /mnt/windows/BROKEN
esser@dissdevel:~$ cat broken.txt
cat: broken.txt: Datei oder Verzeichnis nicht gefunden
esser@dissdevel:~$ file broken.txt
broken.txt: broken symbolic link to `/mnt/windows/Windows/BROKEN'
```

104.6: Soft Links / Hard Links (3)

- Hard Links
  - erstellen mit `ln` (ohne Option)
  - Quelle und Ziel zeigen auf gleichen I-Node
    → darum nur innerhalb eines Dateisystems möglich

```
esser@dissdevel:~$ touch datei.txt
esser@dissdevel:~$ cp datei.txt kopie.txt
esser@dissdevel:~$ ln datei.txt link.txt

esser@dissdevel:~$ ls -il *.txt
12589 -rw-r--r-- 2 esser esser  0  2. Jun 20:16 datei.txt
12590 -rw-r--r-- 1 esser esser  0  2. Jun 20:16 kopie.txt
12589 -rw-r--r-- 2 esser esser  0  2. Jun 20:16 link.txt

esser@dissdevel:~$ ln /mnt/windows/config.sys config.sys
ln: Erzeuge harte Verknüpfung „config.sys“     „/mnt/windows/config.sys“:
Ungültiger Link über Gerätegrenzen hinweg
```

`-i` : I-Nodes anzeigen

rot: link count

104.6: Soft Links / Hard Links (4)

- Hard Links / Links / Löschen
  - Jeder Eintrag in einem Verzeichnis ist ein Link
  - Das Anlegen eines Hard Links bedeutet also nur: Für die Datei (für den I-Node!) existieren jetzt zwei Einträge in einem (oder mehreren) Verzeichnissen
  - Linux kennt intern keine „Lösch“-Operation, sondern nur eine „Unlink“-Operation
    - sie entfernt den ausgewählten Link, also die Zuordnung Dateiname → I-Node
    - und zählt den Link Count um 1 runter
    - Wenn Link Count 0 erreicht wird, wird I-Node freigegeben

Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz F: „Auskunft“
Topic 101.1: Informationen über das System
Topic 104.7: Suche nach Dateien

101.1 Determine and configure hardware settings

Description: Candidates should be able to determine and configure fundamental system hardware.

Key Knowledge Areas:

- Enable and disable integrated peripherals.
- Configure systems with or without external peripherals such as keyboards.
- Differentiate between the various types of mass storage devices.
- Set the correct hardware ID for different devices, especially the boot device.
- Know the differences between coldplug and hotplug devices.
- Determine hardware resources for devices.
- Tools and utilities to list various hardware information (e.g. lsusb, lspci, etc.)
- Tools and utilities to manipulate USB devices
- Conceptual understanding of sysfs, udev, hald, dbus

The following is a partial list of the used files, terms and utilities:
/sys, /proc, /dev, modprobe, lsmod, lspci, lsusb

Quelle:
http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

Auskunft

104.7 Find system files and place files in the correct location

Description: Candidates should be thouroughly familiar with the Filesystem Hierarchy Standard (FHS), including typical file locations and directory classifications.

Key Knowledge Areas:

- Understand the correct locations of files under the FHS.
- Find files and commands on a Linux system.
- Know the location and purpose of important file and directories as defined in the FHS.

The following is a partial list of the used files, terms and utilities:
find, locate, updatedb, whereis, which, type, /etc/updatedb.conf

Quelle: http://www.lpi.org/eng/certification/the_lpic_program/lpic_1/exam_101_detailed_objectives

101.1 Hardware-Überblick (1)

- Hardware: Was gibt's im / am Rechner?
  - Prozessor(en) → `/proc/cpuinfo`
  - Festplatten → `fdisk -l`; `/proc/partitions`
  - Hauptspeicher → `free`; `/proc/meminfo`
  - PCI-Karten → `lspci`
  - USB-Geräte → `lsusb`
  - SCSI-Geräte → `/proc/scsi/scsi`
  - Drucker (konfiguriert) → `lpc stat`
  - Gesamtüberblick → `lshw`

101.1 Hardware-Überblick (2)

- CPU: `/proc/cpuinfo`

```
root@quadamd:/home/esser# cat /proc/cpuinfo
processor       : 0
vendor_id       : AuthenticAMD
cpu family      : 16
model           : 5
model name      : AMD Athlon(tm) II X4 640
Processor
stepping        : 3
cpu MHz         : 800.000
cache size      : 512 KB
physical id     : 0
siblings        : 4
core id         : 0
cpu cores       : 4
apicid          : 0
initial apicid  : 0
fdiv_bug        : no
hlt_bug         : no
f00f_bug        : no
coma_bug        : no
fpu             : yes
fpu_exception   : yes
cpuid level     : 5
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8
apic sep mtrr pge mca cmov pat pse36 clflush mmx
fxsr sse sse2 ht syscall nx mmxext fxr_opt pdpe1gb
rdtscp lm 3dnowext 3dnow constant_tsc nonstop_tsc
extd_apicid pni monitor cx16 popcnt lahf_lm
cmp_legacy svm extapic cr8_legay abm sse4a
misalignsse 3dnowprefetch osvw ibs skinit wdt npt
lbrv svm_lock nrip_save
bogomips        : 6000.17
clflush size    : 64
cache_alignment : 64
address sizes   : 48 bits physical, 48 bits virtual
power management: ts ttp tm stc 100mhzsteps hwpstate

processor       : 1
vendor_id       : AuthenticAMD
cpu family      : 16
model           : 5
model name      : AMD Athlon(tm) II X4 640 Processor
stepping        : 3
cpu MHz         : 800.000
cache size      : 512 KB
physical id     : 0
siblings        : 4
core id         : 1
[...]
```

101.1 Hardware-Überblick (3)

- Festplatten: `fdisk -l`

```
root@quadamd:/home/esser# fdisk -l

Platte /dev/sda: 1000.2 GByte, 1000204886016 Byte
255 Köpfe, 63 Sektoren/Spur, 121601 Zylinder
Einheiten = Zylinder von 16065 * 512 = 8225280 Bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x7c353451

   Gerät  boot.     Anfang        Ende     Blöcke   Id  System
/dev/sda1              1       91381   734009850    7  HPFS/NTFS
/dev/sda2          118989      121601    20980891    f  W95 Erw. (LBA)
/dev/sda3           91381       91903     4191941+  82  Linux Swap / Solaris
/dev/sda4   *       91903      118989   217576295+  83  Linux
/dev/sda5          118990      121601    20980890    b  W95 FAT32

Platte /dev/sdb: 1500.3 GByte, 1500301910016 Byte
255 Köpfe, 63 Sektoren/Spur, 182401 Zylinder
Einheiten = Zylinder von 16065 * 512 = 8225280 Bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x2bd2c32a

   Gerät  boot.     Anfang        Ende     Blöcke   Id  System
/dev/sdb1   *           1          13      102400    7  HPFS/NTFS
/dev/sdb2              13      178355  1432527872    7  HPFS/NTFS
/dev/sdb3          178355      182271    31457280    7  HPFS/NTFS
/dev/sdb4          182271      182402     1048984   12  Compaq Diagnostics
```

101.1 Hardware-Überblick (4)

- Festplatten: `/proc/partitions`

```
root@quadamd:/home/esser# cat /proc/partitions
major minor  #blocks  name

 250        0    1030460 ramzswap0
   8        0  976762584 sda
   8        1  734009850 sda1
   8        2          1 sda2
   8        3    4191941 sda3
   8        4  217576295 sda4
   8        5   20980890 sda5
   8       16 1465138584 sdb
   8       17     102400 sdb1
   8       18 1432527872 sdb2
   8       19   31457280 sdb3
   8       20    1048984 sdb4
```

101.1 Hardware-Überblick (5)

- RAM: `free`; `/proc/meminfo`

```
root@quadamd:/home/esser# free
             total       used       free     shared    buffers     cached
Mem:       4121844    2195528    1926316          0     615280     782168
-/+ buffers/cache:     798080    3323764
Swap:      4191936          0    4191936

root@quadamd:/home/esser# cat /proc/meminfo
MemTotal:        4121844 kB
MemFree:         1926316 kB
Buffers:          615280 kB
Cached:           782172 kB
SwapCached:            0 kB
Active:          1124828 kB
Inactive:         838884 kB
Active(anon):     468932 kB
Inactive(anon):   101772 kB
[...]
```

101.1 Hardware-Überblick (6)

- PCI-Karten: `lspci`

```
root@quadamd:/home/esser# lspci
00:00.0 Host bridge: ATI Technologies Inc RX780/RX790 Chipset Host Bridge
00:02.0 PCI bridge: ATI Technologies Inc RD790 PCI to PCI bridge (external gfx0 port A)
00:05.0 PCI bridge: ATI Technologies Inc RD790 PCI to PCI bridge (PCI express gpp port B)
00:06.0 PCI bridge: ATI Technologies Inc RD790 PCI to PCI bridge (PCI express gpp port C)
00:11.0 SATA controller: ATI Technologies Inc SB7x0/SB8x0/SB9x0 SATA Controller [AHCI mode]
00:12.0 USB Controller: ATI Technologies Inc SB7x0/SB8x0/SB9x0 USB OHCI0 Controller
00:12.1 USB Controller: ATI Technologies Inc SB7x0 USB OHCI1 Controller
00:12.2 USB Controller: ATI Technologies Inc SB7x0/SB8x0/SB9x0 USB EHCI Controller
00:13.0 USB Controller: ATI Technologies Inc SB7x0/SB8x0/SB9x0 USB OHCI0 Controller
00:13.1 USB Controller: ATI Technologies Inc SB7x0 USB OHCI1 Controller
00:13.2 USB Controller: ATI Technologies Inc SB7x0/SB8x0/SB9x0 USB EHCI Controller
00:14.0 SMBus: ATI Technologies Inc SBx00 SMBus Controller (rev 3c)
00:14.1 IDE interface: ATI Technologies Inc SB7x0/SB8x0/SB9x0 IDE Controller
00:14.2 Audio device: ATI Technologies Inc SBx00 Azalia (Intel HDA)
00:14.3 ISA bridge: ATI Technologies Inc SB7x0/SB8x0/SB9x0 LPC host controller
00:14.4 PCI bridge: ATI Technologies Inc SBx00 PCI to PCI Bridge
00:14.5 USB Controller: ATI Technologies Inc SB7x0/SB8x0/SB9x0 USB OHCI2 Controller
00:18.0 Host bridge: Advanced Micro Devices [AMD] Family 10h Processor HyperTransport
Configuration
00:18.1 Host bridge: Advanced Micro Devices [AMD] Family 10h Processor Address Map
00:18.2 Host bridge: Advanced Micro Devices [AMD] Family 10h Processor DRAM Controller
00:18.3 Host bridge: Advanced Micro Devices [AMD] Family 10h Processor Miscellaneous Control
00:18.4 Host bridge: Advanced Micro Devices [AMD] Family 10h Processor Link Control
01:00.0 VGA compatible controller: nVidia Corporation NV43 [GeForce 6600] (rev a2)
02:00.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL8111/8168B PCI Express Gigabit
Ethernet controller (rev 06)
03:00.0 USB Controller: NEC Corporation uPD720200 USB 3.0 Host Controller (rev 03)
```

101.1 Hardware-Überblick (7)

- PCI-Karten: evtl. numerische Angaben besser

```
root@quadamd:/home/esser# lspci
00:00.0 Host bridge: ATI Technologi ...
00:02.0 PCI bridge: ATI Technologie ...
00:05.0 PCI bridge: ATI Technologie ...
00:06.0 PCI bridge: ATI Technologie ...
00:11.0 SATA controller: ATI Techno ...
00:12.0 USB Controller: ATI Technol ...
00:12.1 USB Controller: ATI Technol ...
00:12.2 USB Controller: ATI Technol ...
00:13.0 USB Controller: ATI Technol ...
00:13.1 USB Controller: ATI Technol ...
00:13.2 USB Controller: ATI Technol ...
00:14.0 SMBus: ATI Technologies Inc ...
00:14.1 IDE interface: ATI Technolo ...
00:14.2 Audio device: ATI Technolog ...
00:14.3 ISA bridge: ATI Technologie ...
00:14.4 PCI bridge: ATI Technologie ...
00:14.5 USB Controller: ATI Technol ...
00:18.0 Host bridge: Advanced Micro ...
00:18.1 Host bridge: Advanced Micro ...
00:18.2 Host bridge: Advanced Micro ...
00:18.3 Host bridge: Advanced Micro ...
00:18.4 Host bridge: Advanced Micro ...
01:00.0 VGA compatible controller:  ...
02:00.0 Ethernet controller: Realte ...
03:00.0 USB Controller: NEC Corpora ...
```

```
root@quadamd:/home/esser# lspci -n
00:00.0 0600: 1002:5957
00:02.0 0604: 1002:5978
00:05.0 0604: 1002:597b
00:06.0 0604: 1002:597c
00:11.0 0106: 1002:4391
00:12.0 0c03: 1002:4397
00:12.1 0c03: 1002:4398
00:12.2 0c03: 1002:4396
00:13.0 0c03: 1002:4397
00:13.1 0c03: 1002:4398
00:13.2 0c03: 1002:4396
00:14.0 0c05: 1002:4385 (rev 3c)
00:14.1 0101: 1002:439c
00:14.2 0403: 1002:4383
00:14.3 0601: 1002:439d
00:14.4 0604: 1002:4384
00:14.5 0c03: 1002:4399
00:18.0 0600: 1022:1200
00:18.1 0600: 1022:1201
00:18.2 0600: 1022:1202
00:18.3 0600: 1022:1203
00:18.4 0600: 1022:1204
01:00.0 0300: 10de:0141 (rev a2)
02:00.0 0200: 10ec:8168 (rev 06)
03:00.0 0c03: 1033:0194 (rev 03)
```

101.1 Hardware-Überblick (8)

- `lspci` beherrscht auch
  - kombinierte Ausgabe von Namen und Nummern (über Option `-nn`)

```
# lspci -nn
01:00.0 VGA compatible controller [0300]: nVidia Corporation
NV43 [GeForce 6600] [10de:0141] (rev a2)
```

  - Ausgabe der benutzten Kernel-Module (`-k`)

```
# lspci -k
01:00.0 VGA compatible controller: nVidia Corporation NV43
          [GeForce 6600] (rev a2)
        Subsystem: Giga-byte Technology GV-NX66128DP Turbo
          Force Edition
        Kernel driver in use: nvidia
        Kernel modules: nvidia-current, nouveau, nvidiafb
```

101.1 Hardware-Überblick (9)

- USB-Geräte: `lsusb`

```
root@quadamd:/home/esser# lsusb
Bus 008 Device 002: ID 174c:55aa ASMedia Technology Inc.
Bus 008 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 007 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 006 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 005 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 004 Device 004: ID 03f0:0e17 Hewlett-Packard LaserJet 1015
Bus 004 Device 003: ID 05e3:0604 Genesys Logic, Inc. USB 1.1 Hub
Bus 004 Device 002: ID 046d:c05b Logitech, Inc.
Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 002 Device 003: ID 13d3:3306 IMC Networks
Bus 002 Device 002: ID 058f:6360 Alcor Micro Corp. Multimedia Card Reader
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 04a9:2213 Canon, Inc. CanoScan LiDE 50/LiDE 35/LiDE 40
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

- mehr Informationen mit Option `-v`

101.1 Hardware-Überblick (10)

- SCSI-Geräte: `/proc/scsi/scsi`
  - Achtung: hier tauchen auch SATA-Platten auf

```
root@quadamd:/home/esser# cat /proc/scsi/scsi
Attached devices:
Host: scsi2 Channel: 00 Id: 00 Lun: 00
  Vendor: ATA      Model: WDC WD10EACS-22D Rev: 01.0
  Type:   Direct-Access                    ANSI  SCSI revision: 05
Host: scsi3 Channel: 00 Id: 00 Lun: 00
  Vendor: TSSTcorp Model: CDDVDW SH-S223C  Rev: me01
  Type:   CD-ROM                           ANSI  SCSI revision: 05
Host: scsi5 Channel: 00 Id: 00 Lun: 00
  Vendor: ATA      Model: ST31500541AS     Rev: CC34
  Type:   Direct-Access                    ANSI  SCSI revision: 05
Host: scsi8 Channel: 00 Id: 00 Lun: 00
  Vendor: Generic  Model: CF Card Reader   Rev: 1.00
  Type:   Direct-Access                    ANSI  SCSI revision: 00
Host: scsi8 Channel: 00 Id: 00 Lun: 01
  Vendor: Generic  Model: SD MS Reader     Rev: 1.01
  Type:   Direct-Access                    ANSI  SCSI revision: 00
Host: scsi9 Channel: 00 Id: 00 Lun: 00
  Vendor: ST950032 Model: 5AS              Rev: 0002
  Type:   Direct-Access                    ANSI  SCSI revision: 00
```

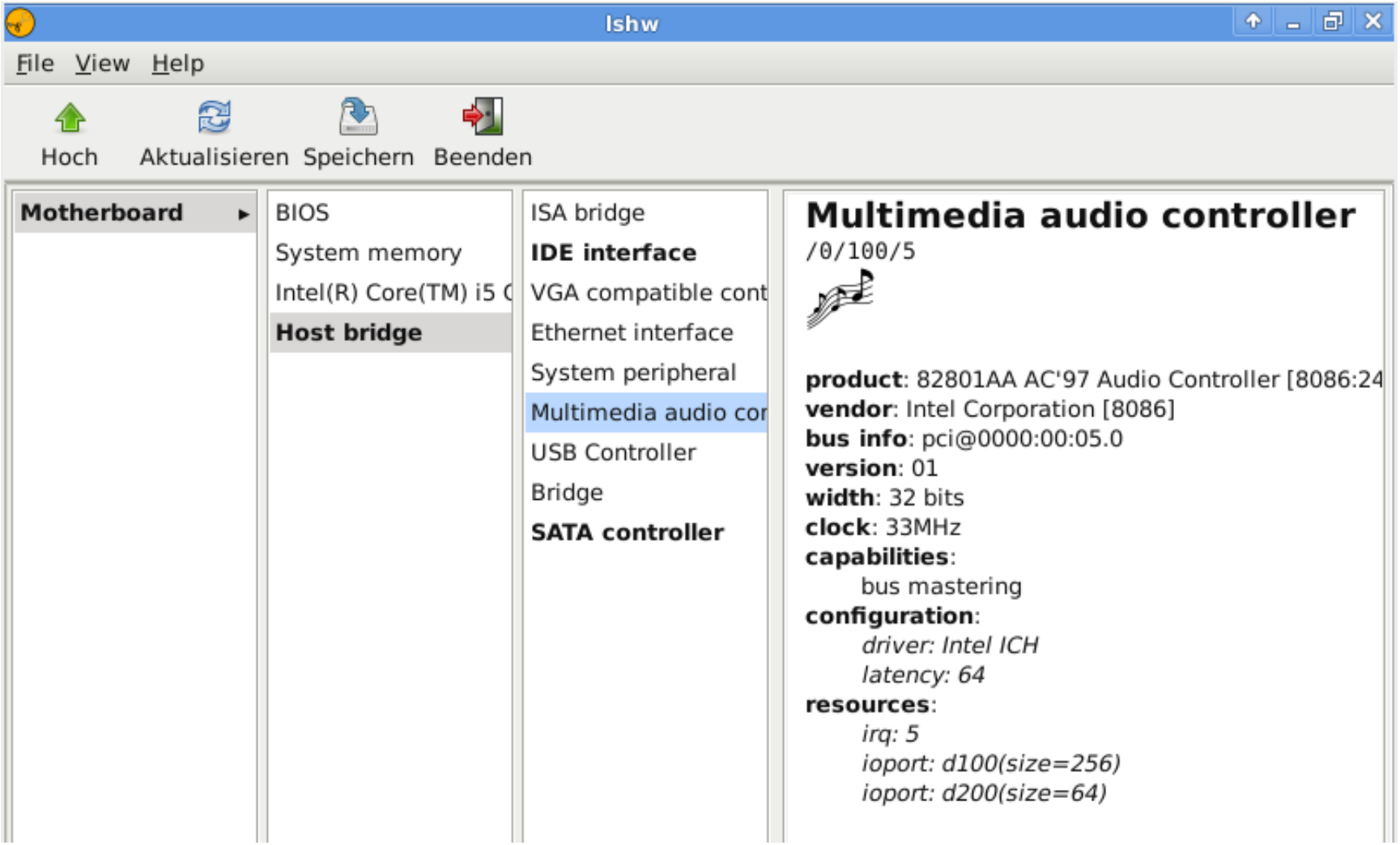
101.1 Hardware-Überblick (11)

- Drucker
  - tatsächlich angeschlossene Geräte: `lsusb` oder Blick ins Systemlog (für Geräte am Parallelport)
  - konfigurierte Geräte: `lpc stat`

```
root@quadamd:/# lpc stat
canon:
        printer is on device 'tpu' speed -1
        queuing is enabled
        printing is enabled
        no entries
        daemon present
hp-LaserJet-1015:
        printer is on device 'hp' speed -1
        queuing is enabled
        printing is enabled
        no entries
        daemon present
mc2430DL:
        printer is on device 'dnssd' speed -1
        queuing is enabled
[...]
```

101.1 Hardware-Überblick (12)

- Gesamtüberblick: `lshw`, auch als graf. Tool (lshw-gtk)



101.1 Hardware-Überblick (13)

- Weitere Quellen für Informationen zur Hardware:
  - Boot-Meldungen → Kommando `dmesg`
  - Aktuelles Syslog (z. B. beim Einstecken eines USB-Geräts): in `/var/log/syslog`, `/var/log/messages` oder `/var/log/kern.log`

```
Jul  1 20:46:40 quadamd kernel: [50743.187928] scsi10 : usb-storage 8-4:1.0
Jul  1 20:46:41 quadamd kernel: [50744.188998] scsi 10:0:0:0: Direct-Access     OLYMPUS  DVR
                                               1.00 PQ: 0 ANSI: 2
Jul  1 20:46:41 quadamd kernel: [50744.190680] sd 10:0:0:0: Attached scsi generic sg7 type 0
Jul  1 20:46:41 quadamd kernel: [50744.191046] sd 10:0:0:0: [sdg] 512000 2048-byte logical blocks:
                                               (1.04 GB/1000 MiB)
Jul  1 20:46:41 quadamd kernel: [50744.191353] xhci_hcd 0000:03:00.0: WARN: Stalled endpoint
Jul  1 20:46:41 quadamd kernel: [50744.191878] sd 10:0:0:0: [sdg] Write Protect is off
Jul  1 20:46:41 quadamd kernel: [50744.191890] sd 10:0:0:0: [sdg] Mode Sense: 0b 00 00 08
Jul  1 20:46:41 quadamd kernel: [50744.192563] sd 10:0:0:0: [sdg] No Caching mode page present
Jul  1 20:46:41 quadamd kernel: [50744.192573] sd 10:0:0:0: [sdg] Assuming drive cache: write through
Jul  1 20:46:41 quadamd kernel: [50744.196479] sd 10:0:0:0: [sdg] 512000 2048-byte logical blocks:
                (1.04 GB/1000 MiB)
Jul  1 20:46:41 quadamd kernel: [50744.196598] xhci_hcd 0000:03:00.0: WARN: Stalled endpoint
Jul  1 20:46:41 quadamd kernel: [50744.197658] sd 10:0:0:0: [sdg] No Caching mode page present
Jul  1 20:46:41 quadamd kernel: [50744.197669] sd 10:0:0:0: [sdg] Assuming drive cache: write through
Jul  1 20:46:41 quadamd kernel: [50744.202389]  sdg: sdg1
```

101.1 Kernel & Module (1)

- Kernel-Version: `uname -a`, `/proc/version`
- Module: `lsmod`
- Info zu einzelnen Modulen: `modinfo`
- Module liegen in `/lib/modules/`

101.1 Kernel & Module (2)

- Versionsabfrage mit `uname -a`:

```
root@quadamd:/# uname -a
Linux quadamd 3.0.0-12-generic-pae #20-Ubuntu SMP
Fri Oct 7 16:37:17 UTC 2011 i686 athlon i386
GNU/Linux
```

- `uname -r`: nur Versionsnummer

```
root@quadamd:/# uname -r
3.0.0-12-generic-pae
```

- Versionsabfrage über `/proc/version`:

```
root@quadamd:# cat /proc/version
Linux version 3.0.0-12-generic-pae
(buildd@vernadsky) (gcc version 4.6.1
(Ubuntu/Linaro 4.6.1-9ubuntu3) ) #20-Ubuntu SMP Fri
Oct 7 16:37:17 UTC 2011
```

101.1 Kernel & Module (3)

- Übersicht der geladenen Module: `lsmod`
  - Achtung: ohne fest einkompilierte Treiber

```
root@quadamd:/# lsmod
Module                  Size  Used by
nls_iso8859_1          12617  1
nls_cp437              12751  1
vfat                   17335  1
fat                    55505  1 vfat
parport_pc             32111  0
ppdev                  12849  0
dm_crypt               22463  0
vesafb                 13449  1
iptable_filter         12706  0
ip_tables              18125  1 iptable_filter
x_tables               21907  2 iptable_filter,ip_tables
[...]
```

101.1 Kernel & Module (4)

- Detail-Informationen zu einzelnen Modulen: `modinfo`

```
root@quadamd:/# modinfo vfat
filename:       /lib/modules/3.0.0-12-generic-pae/kernel/fs/fat/vfat.ko
author:         Gordon Chaffee
description:    VFAT filesystem support
license:        GPL
srcversion:     899FE608BF04B9A6A235C93
depends:        fat
vermagic:       3.0.0-12-generic-pae SMP mod_unload modversions 686

root@quadamd:/# modinfo dm_crypt
filename:       /lib/modules/3.0.0-12-generic-pae/kernel/drivers/md/
                dm-crypt.ko
license:        GPL
description:    device-mapper target for transparent encryption / decryption
author:         Christophe Saout <christophe@saout.de>
srcversion:     5EA8E8471ED48BF61B54ECA
depends:
vermagic:       3.0.0-12-generic-pae SMP mod_unload modversions 686
```

101.1 Kernel & Module (5)

- Module in `/lib/modules`
  - Jede Kernel-Version (auch mit unterschiedlichen Optionen kompilierte Varianten derselben Version) hat dort eigenen Ordner
  - Module nach Themen sortiert (Dateisysteme, Netzwerk, Treiber etc.)

```
root@quadamd:/# ls -l /lib/modules/
insgesamt 20
drwxr-xr-x 3 root root 4096 2011-05-24 22:53 2.6.32-25-generic-pae
drwxr-xr-x 5 root root 4096 2011-10-29 22:53 2.6.38-11-generic-pae
drwxr-xr-x 4 root root 4096 2011-10-29 22:22 3.0.0-12-generic
drwxr-xr-x 5 root root 4096 2011-11-19 11:04 3.0.0-12-generic-pae
drwxr-xr-x 3 root root 4096 2011-12-13 22:18 3.0.0-14-generic
```

104.7: Standardverzeichnisse (1)

- Der Linux Filesystem Hierarchy Standard (FHS, siehe http://www.pathname.com/fhs/) gibt Empfehlungen für die korrekten Positionen diverser Dateiarten:
  - `/bin`: Programme (**bin**aries),
  - `/sbin`: Systemprogramme, (**s**ystem **bin**aries)
  - `/lib`: Bibliotheken (**lib**raries),

  } die schon beim Systemstart verfügbar sein (= auf der Root-Partition liegen) sollen

  - `/usr/bin`, `/usr/sbin`, `/usr/lib`: wie `/bin`, `/sbin` und `/lib`, aber nicht zwingend schon beim Systemstart

```
$ du -sm /bin /sbin /lib /usr/bin /usr/sbin /usr/lib | column -c40
4       /bin            92      /usr/bin
3       /sbin           21      /usr/sbin
7       /lib            236     /usr/lib
```

104.7: Standardverzeichnisse (2)

- `/usr/local/*` (* = bin, sbin, lib): wie oben, aber für vom Anwender installierte Programme und Bibliotheken (Software, die nicht über die Paketverwaltung kommt)
- `/usr/share`: architekturunabhängige Dateien, Doku
  - `/usr/share/man`: Manpages
- `/boot`: Startdateien (Kernel, Initial Ramdisk, Konfiguration des Bootmanagers)
- `/etc`: systemweite Konfigurationsdateien
- `/etc/init.d`: Start- / Stop-Skripte (u. a. für Server)
- `/etc/rc*.d`: Konfigurations verschiedener Runlevels (symbolische Links auf die Skripte in `/etc/init.d`)

104.7: Standardverzeichnisse (3)

- `/var`: veränderliche (**var**iable) Systemdateien, z. B. Spool-Verzeichnisse, temporäre Dateien
  - `/var/spool`: Spooler (z. B. cups, cron, Mail)
  - `/var/lock`: Lock-Dateien
- `/home`: private Verzeichnisse der Anwender (außer root)
- `/root`: Home-Verzeichnis des Administrators (sollte auf der Root-Partition liegen)
- `/dev`: alle Gerätedateien (z. B. `/dev/sda1`)
- `/proc`: Prozess- und Systeminformationen
- `/sys`: mehr Systeminformationen (ähnlich `/proc`)

104.7: Standardverzeichnisse (4)

- `/opt`: Programmpakete von anderen Anbietern, liegen meist in eigenen Unterordnern, die den Programm- oder Herstellernamen tragen (`/opt/oracle`, `/opt/kde`)
  - darunter dann oft Unterordner `bin`, `lib` etc.
- `/media`: Entfernbare Datenträger (CD, DVD, USB-Stick, Speicherkarte, externe Platte)
  - jeder Datenträger erhält ein eigenes Verzeichnis in `/media`, oft: `/media/`*`VolumeName`* (falls verfügbar), sonst generisch, z. B. `/media/cdrom`

104.7 Dateien finden (1)

- Mehrere Tools helfen bei der Suche nach Dateien:
  - `find` – durchsucht ein Verzeichnis
  - `locate` – sucht mit Hilfe einer Datenbank
  - `which` – findet Ort, an dem ein bekanntes Programm liegt (Verzeichnis muss im Pfad, `$PATH`, enthalten sein)
  - `whatis` – findet Kommandos, die eine Manpage haben
  - `whereis` – findet Programm und Manpage (Pfade)
  - `type` – Auskunft über Befehle (Bash-Kommando)

104.7 Dateien finden (2): find

- `find` sucht in einem angegebenen Verzeichnis nach Dateien, die eines oder mehrere dieser Kriterien erfüllen
  - Dateiname (mit Wildcards)
  - Größe (<, >, =)
  - Dateiart (normale Datei, Symlink, Verzeichnis)
  - Zeit seit letzter Änderung an Datei
  - Zeit seit letztem (Lese- oder Schreib-) Zugriff
  - Besitzer der Datei
  - Zugriffsrechte

104.7 Dateien finden (3): find

- Allgemeine Syntax:

  `find` *`Ordner`* [*`Ord2`* …] *`Option`* [*`Opt2`* …] [*`Aktionen`*]

- Beispiele:
  - `find ~ -name '*.pdf'`
    (Dateien im Home-Verz. des Benutzers, die auf `.pdf` enden)
  - `find /usr/bin -type l`
    (alle symbolischen Links unterhalb `/usr/bin`)
  - `find /tmp -user esser`
    (Dateien in `/tmp`, die dem Benutzer `esser` gehören)
  - `find /home/a /home/b -iname '*jpg*'`
    (Dateien unterhalb `/home/a` *oder* `/home/b`, deren Name `jpg` – in beliebiger Groß-/Kleinschreibung – enthält)

104.7 Dateien finden (4): find

- Weitere `find`-Optionen:
  - `-mindepth` *n*, `-maxdepth` *n*
    (Verzeichnistiefe mindestens / höchstens *n*)
  - `-newer` *Referenzdatei*
    (Dateien, die nach der Referenzdatei zuletzt geändert wurden)
  - `-nouser`
    (Dateien, die keinem im System bekannten User gehören)
  - `-readable`, `-writable`
    (lesbare bzw. beschreibbare Dateien)
  - `-regex`
    (Namenssuche mit regulärem Ausdruck statt Wildcards)

104.7 Dateien finden (5): find

- `find`-Aktionen:
  - Standard: Ausgabe des Dateinamens (`-print`)
  - `-ls`: Treffer im `ls`-Format (`ls -dils`) ausgeben
  - `-print0`: Zeichen `\0` als Trenner (statt Zeilenumbruch), siehe Folie zu `find` + `xargs`
  - `-delete`: Treffer löschen
  - `-exec` *`command`* `{} \;`
    Befehl für jeden Treffer (also mehrfach) ausführen, z. B.
    `find . -name '*.pdf' -exec cp {} {}.bak \;`
  - `-exec` *`command`* `{} \+`
    Befehl *einmal* ausführen, alle Treffer als Argumente, z. B.
    `find . -name '*.pdf' -exec chmod o-rw {} \+`

104.7 Dateien finden (6): find

- Beispiele für `-exec`:

```
[esser@d107:tmp]$ ls -l
-rw-r--r--   1 esser  wheel    0 Jul  1 19:55 Eine Datei.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Ganz viele Blanks.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Zwei.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 test.doc
[esser@d107:tmp]$ find . -name '*.txt' -exec cp {} {}.bak \;
[esser@d107:tmp]$ ls -l
-rw-r--r--   1 esser  wheel    0 Jul  1 19:55 Eine Datei.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Eine Datei.txt.bak
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Ganz viele Blanks.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Ganz viele Blanks.txt.bak
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Zwei.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Zwei.txt.bak
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 test.doc
[esser@d107:tmp]$ find . -name '*.txt' -exec chmod go-r {} \+
[esser@d107:tmp]$ ls -l | head -3
-rw-------   1 esser  wheel    0 Jul  1 19:55 Eine Datei.txt
-rw-r--r--   1 esser  wheel    0 Jul  1 19:56 Eine Datei.txt.bak
-rw-------   1 esser  wheel    0 Jul  1 19:56 Ganz viele Blanks.txt
```

104.7 Dateien finden (7): find

- `find` und `xargs`:
  - Relikt aus der Zeit, bevor es `find -exec {} \+` gab; leistet im Prinzip dasselbe
    - `find` erzeugt Liste der Treffer, diese gehen per Pipeline (`|`) weiter an `xargs`
    - mit `xargs` Kommando zusammenbauen, das die `find`-Treffer (am Schluss angehängt) enthält
  - `find ... -print0 | xargs -0 command`

    erzeugt denselben Befehl wie

    `find ... -exec command {} \+`

104.7 Dateien finden (8): find

- `xargs` mit freier Positionierung der Argumente:
  - standardmäßig hängt xargs alle via Pipeline übergebenen Argumente hinten an den Befehl, also:

```
$ echo 1 2 3 | xargs echo Hallo
Hallo 1 2 3
```

  - über Option -i Argument als {} frei platzieren:

```
$ echo 1 2 3 | xargs -i echo {} Hallo
1 2 3 Hallo
```

  - Achtung: Anderes Unix, andere Option; z. B. Mac OS:

```
$ echo 1 2 3 | xargs -J{} echo {} Hallo
```

104.7 Dateien finden (9): locate

- `locate` findet mit Hilfe einer Datenbank Dateien viel schneller als `find`
- diese Datenbank muss mit `updatedb` erst erzeugt (und regelmäßig aktualisiert) werden
- `locate` ist i. d. R. nicht vorinstalliert (Pakete: Debian: mlocate, Fedora: findutils, Suse: findutils-locate)
- Beispiel:

```
[esser@quadamd:~]$ locate  '*LNM-Rechnung*pdf*'
/home/esser/Daten/Briefe/LNM-Rechnung-2009-10-15.odt
/home/esser/Daten/Briefe/LNM-Rechnung-2009-10-15.pdf
/home/esser/Daten/Briefe/LNM-Rechnung-2009-10-21.odt
/home/esser/Daten/Briefe/LNM-Rechnung-2009-10-21.pdf
[...]
```

104.7 Dateien finden (10): locate

- auch `locate` mit `xargs` kombinierbar:
  `locate` *`muster`* `| xargs` *`command`*
- `\0`-Terminierung für Dateinamen mit Leerzeichen:
  `locate -0` *`muster`* `| xargs -0` *`command`*
- Beispiel:

```
[esser@quadamd:~]$ locate -0 Evaluation-2011 | xargs -0 file
/home/esser/FOM-SS2011-BS-Praxis/Evaluation-2011-05-14.odt:  OpenDocument Text
/home/esser/FOM-SS2011-BS-Praxis/Evaluation-2011-05-14.pdf:  PDF document
/home/esser/FOM-SS2011-BS-Theorie/Evaluation-2011-05-05.odt: OpenDocument Text
/home/esser/FOM-SS2011-BS-Theorie/Evaluation-2011-05-05.pdf: PDF document
/home/esser/HS-Muenchen-2011/Evaluation-2011-05-13.odt:      OpenDocument Text
[esser@quadamd:~]$ locate -0 Evaluation-2011 | xargs -0 -i cp {} /tmp/eval/
[esser@quadamd:~]$
```

104.7 Dateien finden (11): locate

- Aufbau und Update der Datenbank mit `updatedb`
- Programm meist so konfiguriert, dass es automatisch (per Cronjob) läuft
- je nach `locate`-Variante (es gibt verschiedene) läuft `updatedb` mit den Rechten des Benutzers `nobody`
  → findet nur Dateien, die für jeden sichtbar sind
  (z. B. bei Suse / findutils-locate)
- Einige `locate`-Varianten blenden beim Aufruf durch Benutzer Dateien aus, die für den Anwender nicht auffindbar sind (z. B. bei Debian / `mlocate`)

104.7 Dateien finden (12): which

- `which` zeigt an, welches Programm startet, wenn Sie einen Befehl eingeben
- es wertet dazu die Pfadvariable `$PATH` aus
- gibt es z. B. das Programm `example` in `/bin` und in `/usr/bin`, entscheidet die Reihenfolge in $PATH

```
[esser@quadamd:~]$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:
/sbin:/bin:/usr/games:/home/esser/bin
[esser@quadamd:~]$ which example
/usr/bin/example
```

104.7 Dateien finden (13): whatis

- `whatis` sucht in Liste der Manpages nach passendem Eintrag
- über Option `-w` auch Suche mit Wildcards
- ähnliches Feature (mehr Treffer): `man -k` *`keyword`*

```
[esser@quadamd:~]$ whatis apt-get
apt-get (8)          - APT-Werkzeug für den Umgang mit Paketen --
                       Befehlszeilenschnittstelle
[esser@quadamd:~]$ whatis -w 'apt*' | head -4
apt (8)              - Fortschrittliches Paketwerkzeug (Advanced Package Tool)
apt-cache (8)        - query the APT cache
apt-cdrom (8)        - APT-CDROM-Verwaltungswerkzeug
apt-config (8)       - APT-Konfigurationsabfrageprogramm
[esser@quadamd:~]$ man -k apt
apt (8)              - Fortschrittliches Paketwerkzeug (Advanced Package Tool)
apt-cache (8)        - query the APT cache
[...]
synaptic (8)         - graphical management of software packages
[...]
```

104.7 Dateien finden (14): whereis

- `whereis` kann Programme, die zugehörigen Manpages und ggf. Sourcen (Quellen) finden

```
[esser@quadamd:~]$ whereis bash
bash: /bin/bash /usr/share/man/man1/bash.1.gz
[esser@quadamd:~]$ whereis rpm
rpm: /usr/bin/rpm /usr/lib/rpm /usr/share/man/man8/rpm.8.gz
[esser@quadamd:~]$ whereis pwd
pwd: /bin/pwd /usr/include/pwd.h /usr/share/man/man1/pwd.1.gz
[esser@quadamd:~]$ whereis ld.so.conf
ld.so: /etc/ld.so.cache /etc/ld.so.conf /usr/share/man/man8/ld.so.8.gz
[esser@quadamd:~]$ whereis cd
cd:
```

- Suche einschränken mit Optionen `-b` (binaries), `-m` (manual pages), `-s` (sources)

104.7 Dateien finden (15): type

- `type` ist ein in die Bash eingebautes Kommando
- es gibt zu einem Befehl aus, welcher Art er ist (Shell-Built-in, Alias, Funktion, Binary)
- `type -a`: auch Alternativen anzeigen

```
[esser@quadamd:~]$ type apt-get
apt-get ist /usr/bin/apt-get
[esser@quadamd:~]$ type ls
ls is hashed (/bin/ls)
[esser@quadamd:~]$ type pwd
pwd is a shell builtin
[esser@quadamd:~]$ type -a pwd
pwd is a shell builtin
pwd ist /bin/pwd
[esser@quadamd:~]$ type ll
ll ist ein Alias von
   `ls -l --color'.
```

```
[esser@quadamd:~]$ type append_path
append_path is a function
append_path ()
{
    if ! eval test -z "\"\${$1##*:
$2:*}\"" -o -z "\"\${$1%%*:$2}\"" -o
-z "\"\${$1##$2:*}\"" -o -z "\"\$
{$1##$2}\""; then
        eval "$1=\$$1:$2";
    fi
}
```

Studium neben dem Beruf

FOM Hochschule für Oekonomie & Management
University of Applied Sciences

Betriebssysteme Praxis

WS 2012/13

Hans-Georg Eßer
Dipl.-Math., Dipl.-Inform.

Foliensatz G: „Zugriffsrechte“
„Manage file permissions and ownership“

Zugriffsrechte

Manage file permissions and ownership

Description: Candidates should be able to control file access through the proper use of permissions and ownerships.

Key Knowledge Areas:

- Manage access permissions on regular and special files as well as directories.
- Use access modes such as suid, sgid and the sticky bit to maintain security.
- Know how to change the file creation mask.
- Use the group field to grant file access to group members.

The following is a partial list of the used files, terms and utilities:
chmod, umask, chown, chgrp

Quelle: http://www.lpi.org/linux-certifications/programs/lpic-1/exam-101/

Dateien, Benutzer, Gruppen

- Jede Datei
  - … gehört einem Benutzer (Besitzer, **u**ser)
  - … und zu einer Gruppe (**g**roup)
- Benutzer können Mitglieder in verschiedenen Gruppen sein
- Zugriffsrechte entscheiden, ob eine Datei gelesen (**r**ead), geschrieben (**w**rite) oder ausgeführt (e**x**ecute) werden darf

Gruppen (1)

- In welchen Gruppen bin ich Mitglied?

```
$ groups
fom cdrom floppy audio dip video plugdev netdev
powerdev scanner
```

- Mitgliedschaft durch Einträge in `/etc/group` geregelt:

```
$ grep fom /etc/group
cdrom:x:24:fom
floppy:x:25:fom
audio:x:29:fom
...
fom:x:1002:fom
```

- Gruppenmitgliedschaft bearbeiten: manuell oder (besser!) mit `gpasswd`

Gruppen (2)

- Gruppe mit `gpasswd -a` *`user group`* (**a**dd) ergänzen:

```
# gpasswd -a fom neugr
Benutzer fom wird zur Gruppe neugr hinzugefügt.
# groups fom
fom cdrom floppy audio dip video plugdev netdev
powerdev scanner neugr
```

- Entfernen einer Gruppenmitgliedschaft: `gpasswd -d` *`user group`* (**d**elete)

```
# gpasswd -d fom neugr
Benutzer fom wird aus der Gruppe neugr
entfernt.
```

Gruppen (3)

- Jeder Benutzer ist in einer Standardgruppe Mitglied. Welche ist das?

```
$ id
uid=1002(fom) gid=1002(fom) Gruppen=1002(fom),
24(cdrom),25(floppy),29(audio),30(dip),...
```

- Zwei Standards für Standardgruppe
  - Debian-System: Jeder Benutzer hat seine eigene Standardgruppe (User: `fom`, Group: `fom`)
  - andere Systeme: Standardgruppe `users` für alle „normalen“ Benutzer
- Im Namen der Standardgruppe handeln Sie, bis Sie mit `newgrp` die Gruppe ändern.

Gruppen (4)

- Neue Gruppen kann der Administrator mit `groupadd` erzeugen, um Kooperation von Teams zu erleichtern
  - z. B. mit Dateien, die für alle Gruppenmitglieder (und nur diese) les- und schreibbar sind
- Beispielszenario folgt ...

Beispielszenario (1)

- Gruppe `profs`: Mitglieder `prof1`, `prof2`
- Gruppe `studis`: Mitglieder `anna`, `tom`, `fritz` und (!) `prof1`, `prof2`
- Ziele:
  - `profs`-Mitglieder können Daten untereinander austauschen und teilweise auch Studenten zur Verfügung stellen
  - `studis`-Mitglieder können Daten untereinander austauschen und auf die von Profs zur Verfügung gestellten Skripte, Aufgaben etc. zugreifen

Beispielszenario (2)

- Verzeichnisstruktur

```
/srv/profs/
/srv/profs/intern/                  ; Austausch der Profs untereinander
/src/profs/intern/klausuren/
/srv/profs/public/                  ; Lesezugriff für Studenten möglich
/srv/profs/public/skripte/
/srv/studis/
/srv/studis/mitschriften/
/srv/studis/pruefungsprot/
```

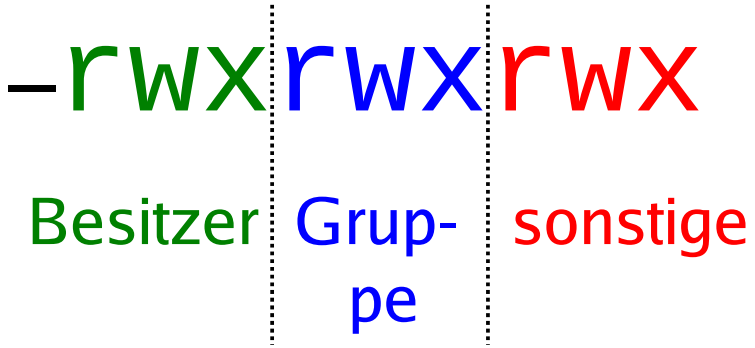
- Gruppenzugehörigkeiten und Zugriffsrechte
  - `/srv/profs/intern`: gehört Gruppe `profs`; lesen und schreiben für `profs` erlaubt, kein Zugriff für `studis`
  - `/srv/profs/public`: gehört Gruppe `profs`; schreiben für `profs` erlaubt, lesen für alle (auch Nicht-Studis)
  - `/srv/studis`: gehört Gruppe `studis`; lesen und schreiben für Gruppenmitglieder erlaubt

Beispielszenario (3)

- Umsetzung: später
- Nachteil: keine vernünftige Zugriffsbeschränkung für `/srv/profs/public` möglich
  → ACLs

Unix-Dateiattribute (1)

- Neben Besitzer und Gruppe gibt es noch die sonstigen Systembenutzer (**o**, others)
- ergibt 9 Zugriffsrechte; Notation bei `ls`:



Unix-Dateiattribute (2)

- `chown` (change owner) und `chgrp` (change group) ändern Besitzer und Gruppe einer Datei
- `chmod` (change mode) ändert Zugriffsrechte
- Beispiele:

```
chown fom /tmp/log.txt
chgrp www-data /var/www/srv1
chmod o+r /tmp/log.txt
chmod o-rwx,ug+rw /tmp/log.txt
chmod u=rw,g=r,o= /tmp/log.txt
```

- Abkürzung a (all) für ogu (`chmod a=rw` ...)

Unix-Dateiattribute (3)

- numerische Rechte:
  - Leserecht: 4 (2^2)
  - Schreibrecht: 2 (2^1)
  - Ausführrecht: 1 (2^0)
  - aufaddieren, z. B.: rw = Lesen/Schreiben: 4+2=6
- für Benutzer, Gruppe und Sonstige: **nnn**
  - z. B. **640**:
    - Benutzer: 6 = lesen + schreiben (nicht ausführen)
    - Gruppe: 4 = lesen (nicht schreiben, nicht ausführen)
    - Sonstige: 0 = nichts

Unix-Dateiattribute (4)

- `chmod` mit numerischen Rechten nutzen
  - `rw- r-- ---` = 640 (4+2+0, 4+0+0, 0+0+0)
  - `chmod u=rw,g=r,o= /tmp/log.txt`
    `chmod 640 /tmp/log.txt`
- bei der numerischen Angabe kein „Geben“ und „Nehmen“ von Rechten möglich
  (wie mit `chmod u+x ...`, `chmod o-rwx ...`)

Unix-Dateiattribute (5)

- Beim Erzeugen einer Datei werden Standard-rechte gesetzt – welche das sind, bestimmt die UMASK (**u**ser file creation **mask**)

```
$ umask
0022
$ umask a=rw
$ umask
0111
$ touch Datei; ls -l Datei
-rw-rw-rw-  1 esser users 0 2008-12-04 20:48 Datei
$ umask u=rw,g=r,o=
$ umask
0137
$ touch Test; ls -l Test
-rw-r-----  1 esser users 0 2008-12-04 20:50 Test
```

Standard:
Gruppe: nicht schreiben,
Sonstige: nicht schreiben

Unix-Dateiattribute (6)

- umask wird von 666 (`rw-rw-rw-`: Standard-wert für Dateien) bitweise abgezogen, um konkrete Dateirechte zu berechnen;
- Ausführrecht wird beim Erzeugen einer Datei nie vergeben

- Linux unterstützt diese klassischen Unix-Dateiattribute und einige zusätzliche...

Unix-Dateiattribute (7)

- Dateiattribute nur auf echten Unix-Dateisystemen nutzbar – auf Windows-Datenträgern nur stark eingeschränkt:

```
# mount | grep windows
/dev/sda3 on /windows/D type vfat (rw,gid=100,umask=0002)
# touch /windows/D/Testdatei
# ls -l /windows/D/Testdatei
-rwxrwxr-x  1 root users 0 2006-12-04 21:07 /windows/D/Testdatei
# chmod a-rwx /windows/D/Testdatei
# ls -l /windows/D/Testdatei
----------  1 root users 0 2006-12-04 21:07 /windows/D/Testdatei
# umount /windows/D; mount /windows/D; ls -l /windows/D/Testdatei
-r-xr-xr-x  1 root users 0 2006-12-04 21:07 /windows/D/Testdatei
```

- Windows kennt kein Ausführattribut – wohl aber ein Read-Only-Attribut

Unix-Dateiattribute (8)

- Bedeutung der Attribute für Verzeichnisse:
  - **r**ead: Verzeichnisinhalt lesen (`ls` in einem Verzeichnis ausführen)
  - **w**rite: Verzeichnisinhalt ändern (z. B. neue Datei erzeugen, Datei umbenennen)
  - e**x**ecute: Verzeichnis betreten, also zum aktuellen Arbeitsverzeichnis machen (`cd`)
  - Standardrechte, von denen die umask abgezogen wird, sind bei Verzeichnissen 777 (denn x = execute steht ja für „Verzeichnis betreten“)

Beispielszenario (4)

- Zurück zum Beispielszenario
  - Ersteinrichtung:

```
root# chown -R root /srv/profs /srv/studis
root# chgrp -R profs /srv/profs/intern
root# chmod ug=rwx,o= /srv/profs/intern
root# chgrp -R profs /srv/profs/public
root# chmod ug=rwx,o=rx /srv/profs/public
root# chgrp -R studis /srv/studis
root# chmod ug=rwx,o= /srv/studis
```

  - neue Dateien erzeugen:

```
prof1$ newgrp profs    # als „profs“-Mitglied arbeiten
prof1$ umask 0007      # Neue Dateien nicht für andere
prof1$ cd /srv/profs/intern
prof1$ touch pruefung.doc
prof1$ ls -l pruefung.doc
-rw-rw---- 1 prof1 profs ...     pruefung.doc
```

SUID, SGID (1)

- Problem: Es gibt Dateien, die Benutzer nur „unter kontrollierten Bedingungen“ ändern dürfen, z. B. die Passwortdatei `/etc/shadow`
  - Änderung an der Datei mit dem Tool `passwd`
  - Dafür sind Root-Rechte nötig
  - Normale Anwender haben keine Root-Rechte
- Zwei Lösungen
  - klassisch: SUID (siehe nächste Folie)
  - neuer: `sudo` (behandeln wir hier nicht)

SUID, SGID (2)

- Ausführbare Dateien (nur Binaries) können ein SUID- (Set User ID) und/oder ein SGID-Bit (Set Group ID) haben
  - SUID: Programm läuft immer mit den Rechten des Dateibesitzers, meist `root`
  - SGID: Programm läuft immer mit den Gruppenrechten der Dateigruppe (seltener verwendet)
  - Beispiel: `passwd` muss Systemdateien ändern

```
$ ls -l /usr/bin/passwd /etc/shadow
-rwsr-xr-x 1 root root    ...  /usr/bin/passwd
-rw-r----- 1 root shadow  ...  /etc/shadow
```

SUID, SGID (3)

- SUID- und SGID-Bits mit chmod setzen

```
# cp /usr/bin/passwd /tmp/mypasswd
# chmod u-s,g+s /tmp/mypasswd
# ls -l /usr/bin/passwd /tmp/mypasswd
-rwsr-xr-x 1 root root    ...  /usr/bin/passwd
-rwxr-sr-x 1 root shadow  ...  /etc/shadow
```

- s-Bits erscheinen in der ls-Ausgabe immer an der Stelle, wo sonst das x steht
- Diese Bits sind bei Shell-Skripten wirkungslos (in einigen älteren Unix-Versionen funktionierte das auch mit Skripten)

Extra-Flags (1)

- Ext2-/Ext3-Extra-Flags (immutable, append-only etc.) mit `chattr` bearbeiten

```
NAME
       chattr - change file attributes on a Linux second extended file system

SYNOPSIS
       chattr [ -RV ] [ -v version ] [ mode ] files...

DESCRIPTION
       chattr changes the file attributes on a Linux second extended file
       system.

       The format of a symbolic mode is +-=[ASacDdIijsTtu].

       The operator `+' causes the selected attributes to be added to the
       existing attributes of the files; `-' causes them to be removed; and
       `=' causes them to be the only attributes that the files have.
```

Extra-Flags (2)

- Beispiel für `chattr`:

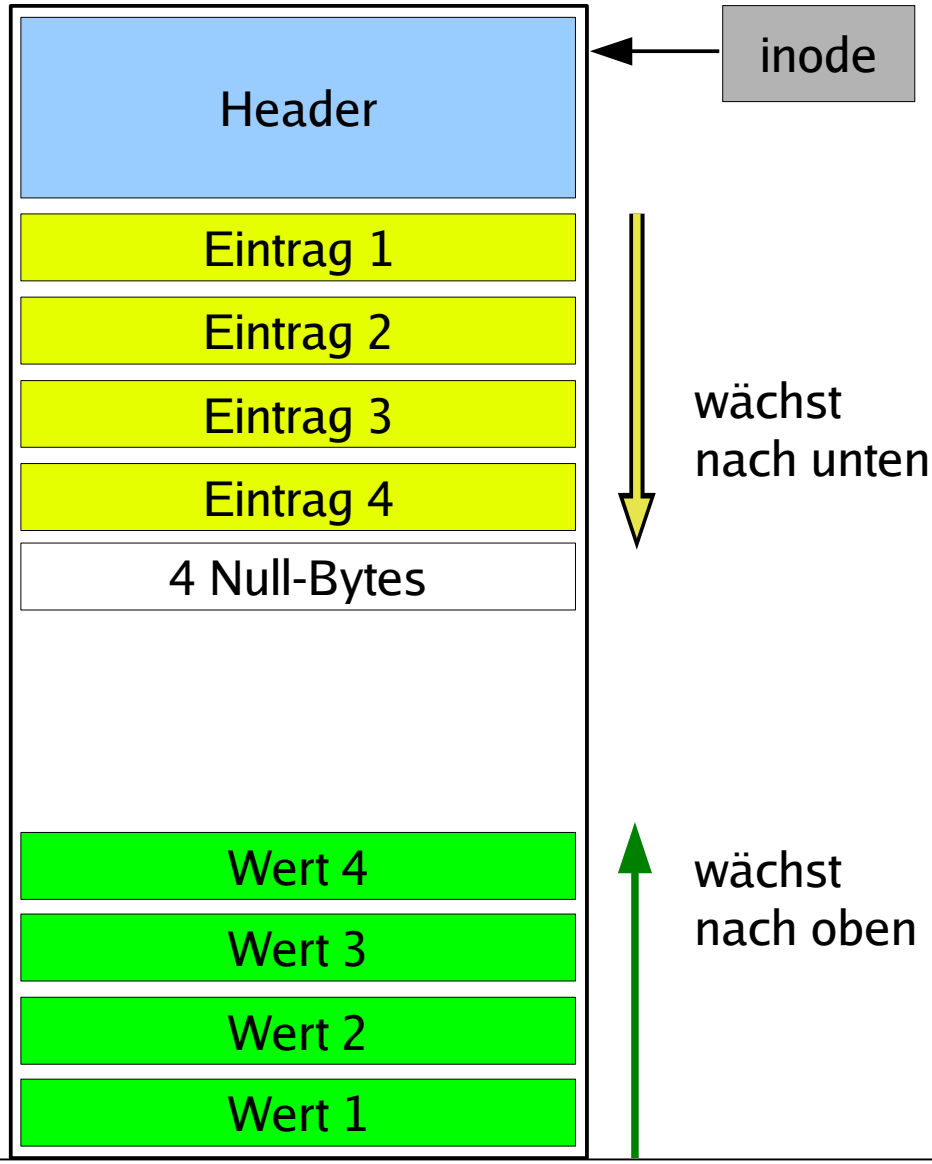
```
# cd /tmp;  touch logdatei
# chattr +a logdatei
# echo Hallo >> logdatei          #  >> = anhängen
# echo Welt >> logdatei
# cat logdatei
Hallo
Welt
# echo Ueberschreiben > logdatei
bash: logdatei: Die Operation ist nicht erlaubt
```

- Attribute anzeigen mit `lsattr`:

```
# lsattr -l logdatei
logdatei                        Append_Only
```

Erweiterte Attribute (1)

- Erweiterte Attribute speichern beliebige Name-/Wert-Paare, u. a. ACLs
- Inode-Größe: 128 Byte
  - kein Platz für erweiterte Attribute
  - Vergrößerung auf 256 Byte nicht effizient
- Lösung: Separater Block für extended attributes



Erweiterte Attribute (2)

- bearbeiten mit `setfattr`, `getfattr`, `attr`:

```
amd64:/home/esser # setfattr -n user.foo -v betriebssysteme test.txt
amd64:/home/esser # getfattr -d test.txt
# file: test.txt
user.foo="betriebssysteme"

amd64:/home/esser # attr -g user.foo test.txt
Attribute "user.foo" had a 15 byte value for test.txt:
betriebssysteme
```

- Software ist auf dem Debian-System nicht installiert → `apt-get install attr`
- Verwaltung von ACLs über das Paket `acl` → `apt-get install acl`
  - Tools: `getfacl`, `setfacl`

Dreierlei Attribute

Nicht verwechseln:

- Standard-Unix-Dateiattribute
  - UID, GID
  - Standardzugriffsrechte rwx für user/group/others
  - Zugriffszeiten, ...
- Extra-Flags
  - immutable, compressed, secure deletion, ...
- Extended Attributes
  - beliebige, frei definierbare Attribute (inkl. ACLs)