

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

# Vorlesung Unix-Praktikum

## 13. Dateisysteme

Carsten Gnörlich

Rechnerbetriebsgruppe  
Technische Fakultät  
Universität Bielefeld

01. Februar 2016

# Willkommen zur dreizehnten Vorlesung

Was gab es beim letzten Mal?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

- ▶ Dateisystem-Querverweise
- ▶ Ausgabeumleitung von *stdout* und *stderr*
- ▶ Paketverwaltung
- ▶ screen

# Willkommen zur dreizehnten Vorlesung

Was machen wir heute?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

## Werkzeuge

tee  
/dev/shm  
watch

## Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf Abbildern

# Ausgabeumleitung

Ausgabe gleichzeitig beobachten und in eine Datei umleiten

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Ein Prozeß läuft mehrere Stunden. Wir wollen

- ▶ seine Ausgabe in eine Datei umleiten
- ▶ gleichzeitig seine Ausgabe “live” beobachten

Beispiel (zum Spielen): `zeit.bash`

```
#!/bin/bash
```

```
while /bin/true; do  
    date "+%A, %d. %B %Y %H:%M:%S"  
    sleep 5  
done
```

# Ausgabeumleitung

Ausgabe gleichzeitig beobachten und in eine Datei umleiten

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

tee: Ein “T-Stück” in eine Pipe einbauen

Beispiel:

```
> ./zeit.bash | tee zeit.log
```

- ▶ man sieht die Ausgabe von `zeit.bash` direkt
- ▶ sie wird zusätzlich in `zeit.log` gespeichert

Anmerkung:

- ▶ Nach `tee` kann die Pipe weitergehen
- ▶ Man “zapft” (kopiert) damit also wirklich Daten aus der Pipe heraus

# Pseudo-Blockgeräte

/dev/shm: Die Festplatte im RAM

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

## /dev/shm bzw. /run/shm

- “Festplatte” im RAM
- Kann bis zur Hälfte des RAMs belegen
- + extrem schnell
- + gut zum Verschieben von Daten  
z.B. ISO-Abbilder bauen, tar-Archive ein-/auspacken
- + gut zum Schonen von SSDs  
z.B. Browser-Cache hierhin auslagern
- aber: Daten im RAM → nach Reboot sind sie weg!

# Kopieren zwischen Blockgeräten

typische Anwendungen von dd

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Eine 16GB große Datei aus Nullen erzeugen:

```
> dd if=/dev/zero of=datei.img bs=1M count=$((16*1024))
```

ISO-Abbild einer CD erzeugen (count weglassen!):

```
> dd if=/dev/scd0 of=~/abbild.iso bs=1M
```

Gesamte Festplatte testweise in die Datensenke kopieren  
(hilft um festzustellen ob sie defekte Sektoren hat)

```
> sudo dd if=/dev/sda of=/dev/null bs=1M
```

Gesamte Festplatte mit Nullen überschreiben (**Datenverlust!**)  
(z.B. um alle Daten vor einem Verkauf zu löschen)

```
> sudo dd if=/dev/zero of=/dev/sda bs=1M
```

# Kopieren zwischen Blockgeräten

dd: Status via *USR1*-Signal abfragen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Einen dd-Prozeß im Hintergrund starten:

```
> dd if=/dev/zero of=/dev/null &  
[4] 3077
```

*USR1*-Signal via *kill* schicken:

(*kill* dient also nicht nur zum Beenden von Prozessen!)

```
> kill -USR1 3077  
17128294+0 Datensätze ein  
17128293+0 Datensätze aus  
8769686016 Bytes (8,8 GB) kopiert, 6,31664 s, 1,4 GB/s
```

dd-Prozeß beenden:

```
> kill 3077  
[4]+ Beendet dd if=/dev/zero of=/dev/null
```

# Virtualisierung unter Ubuntu

watch: Periodisch einen Kommandozeilenbefehl ausführen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

## watch: Programm periodisch ausführen

Aufrufschema: `watch -n 5 kommandozeilenaufruf`

► -n : Anzahl der Sekunden, nach der Aufruf wiederholt wird

Alle 5 Sekunden Größe der Datei `abbild.iso` anzeigen:

```
> watch -n 5 ls -lah abbild.iso
```

Alle 2 Sekunden (Standard) Massenspeicher in `/dev` anzeigen:  
(Ohne "" würde `sd*` expandiert und keine neuen Geräte gezeigt)

```
> watch "ls -l /dev/sd*"
```

Freien Speicherplatz in `/home` beobachten:

```
> watch -n 5 df -h /home
```

# Wiederholung: Partitionen

Logische Partitionen erlauben mehr als 4 Partitionen

Unix-Praktikum

Carsten Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

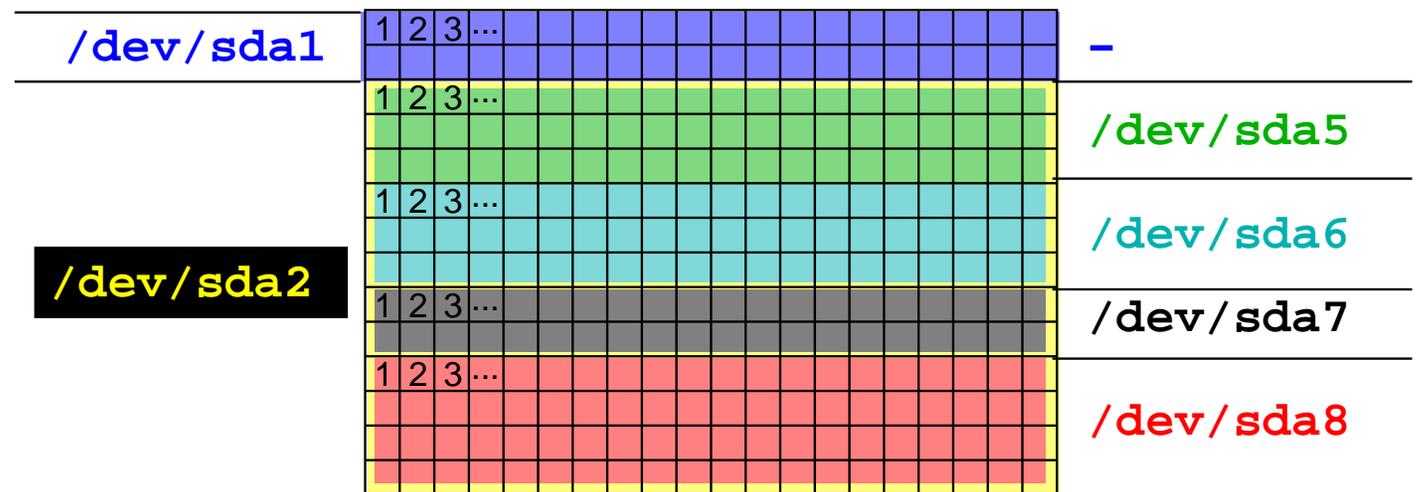
Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf Abbildern

Primäre Part.  
(max 4: sda1–sda4)

**/dev/sda**

Logische Part.  
(beliebig viele)



Eine primäre Partition kann in weitere logische Partitionen aufgeteilt werden:

- ▶ **/dev/sda5**, **/dev/sda6**, **/dev/sda7**, ...
- ▶ passiert während des Installationsdialogs
- ▶ was macht man mit nachträglich eingebauten Datenträgern?

# Partitionierung

## Werkzeuge

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

### Graphisches Werkzeug:

- ▶ gparted

### Kommandozeilenwerkzeug:

- ▶ fdisk

# Partitionierung mit fdisk

Neues Gerät partitionieren

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät

dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

- ▶ neue Festplatte
- ▶ Memorystick

Vorsicht: Richtiges Gerät bearbeiten!  
(sonst Datenverlust!)

# Partitionierung mit fdisk

## Neues Gerät finden

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät

dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Bei fest eingebauten Geräten:

- ▶ Bootmeldungen untersuchen
- ▶ Inhalt von `/dev` vorher und nachher vergleichen
- ▶ Partitionstabellen anschauen

Bei Wechseldatenträgern:

- ▶ Logfiles während des Einsteckens beobachten

# Partitionierung mit fdisk

Wie findet man das neue Gerät?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

## dmesg: Log-Meldungen des Kernels ausgeben

```
> dmesg | grep sd[a-z]
[  2.947433] sd 1:0:0:0: [sda] 781422768 512-byte logical bl
                                     (400 GB/37
[  2.947491] sd 1:0:0:0: [sda] Write Protect is off
[  2.947495] sd 1:0:0:0: [sda] Mode Sense: 00 3a 00 00
[  2.947520] sd 1:0:0:0: [sda] Write cache: enabled, read ca
[  2.977831]   sda: sda1 sda2 sda3 < sda5 >
[  2.978392] sd 1:0:0:0: [sda] Attached SCSI disk
[  4.204527] sd 0:0:0:0: [sdb] 156301488 512-byte logical bl
                                     (80.0 GB/74.
[  4.204583] sd 0:0:0:0: [sdb] Write Protect is off
[  4.204587] sd 0:0:0:0: [sdb] Mode Sense: 00 3a 00 00
[  4.204934] sd 0:0:0:0: [sdb] Write cache: enabled, read ca
[  4.243937]   sdb: sdb1
[  4.244265] sd 0:0:0:0: [sdb] Attached SCSI disk
```

► Größe und Partitionen betrachten

# Partitionierung mit fdisk

Wie findet man das neue Gerät?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Modellbezeichnung aus den Logs holen:

```
> dmesg | grep sd[a-z]
```

```
...
```

```
[    2.947433] sd 1:0:0:0: [sda] 781422768 512-byte logical bl  
                                     (400 GB/37
```

```
...
```

```
[    4.204527] sd 0:0:0:0: [sdb] 156301488 512-byte logical bl  
                                     (80.0 GB/74.
```

```
...
```

```
>  
dmesg | grep 0:0:0:0
```

```
...
```

```
[    4.204265] scsi 0:0:0:0: Direct-Access      ATA      ST3808
```

```
...
```

# Partitionierung mit fdisk

Wie findet man das neue Gerät?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Nachschauen im dev-Verzeichnis:

```
> ls -la /dev/sd*
```

```
brw-rw---- 1 root disk 8, 0 14. Jan 12:34 /dev/sda
brw-rw---- 1 root disk 8, 1 14. Jan 12:34 /dev/sda1
brw-rw---- 1 root disk 8, 2 14. Jan 12:34 /dev/sda2
brw-rw---- 1 root disk 8, 3 14. Jan 12:34 /dev/sda3
brw-rw---- 1 root disk 8, 5 14. Jan 12:34 /dev/sda5
brw-rw---- 1 root disk 8, 16 14. Jan 12:34 /dev/sdb
```

Aber Vorsicht:

- ▶ Reihenfolge kann sich nach Einbau ändern;  
die neue Platte kann auch sda werden!

# Partitionierung mit fdisk

## Partitionierung eines Gerätes anzeigen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
> sudo fdisk -l /dev/sda
```

```
Disk /dev/sda: 128.0 GB, 128035676160 bytes
255 heads, 63 sectors/track, 15566 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: 0x00082c67
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	66	530113+	83	Linux
/dev/sda2		67	15566	124503750	5	Extend
/dev/sda5		67	1111	8393931	83	Linux
/dev/sda6		1112	2156	8393931	83	Linux
/dev/sda7		2157	2679	4200966	82	Linux
/dev/sda8		2680	15566	103514796	83	Linux

► Neue Festplatten haben keine Partitionstabelle!

# Partitionierung mit fdisk

Wie findet man eingesteckte Wechseldatenträger?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Systemmeldungen (u.a. über ein-/ausgesteckte Geräte):

- ▶ /var/log/syslog bzw. /var/log/messages

Ausgeben von neuen Meldungen:

**tail -f**: gebe neu hinzugekommene Zeilen einer Datei aus

Überwachen einer log-Datei:

```
> sudo tail -f /var/log/syslog
```

# Partitionierung mit fdisk

Wie findet man eingesteckte Wechseldatenträger?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Beispiel:

```
> sudo tail -f /var/log/syslog
```

*Memorystick einstecken, auf Ausgaben warten:*

```
Jan 14 16:50:18 wind kernel: [15341.080099] usb 2-4: new high speed USB device using ehci_hc
Jan 14 16:50:18 wind kernel: [15341.215791] usb 2-4: New USB device found, idVendor=0951, id
Jan 14 16:50:18 wind kernel: [15341.215800] usb 2-4: New USB device strings: Mfr=1, Product=
Jan 14 16:50:18 wind kernel: [15341.215806] usb 2-4: Product: DT HyperX
Jan 14 16:50:18 wind kernel: [15341.215811] usb 2-4: Manufacturer: Kingston
Jan 14 16:50:18 wind kernel: [15341.215815] usb 2-4: SerialNumber: 2008012500000000000008B2
... viele weitere Ausgaben ...
Jan 14 16:50:23 wind kernel: [15346.218847] sd 9:0:0:0: Attached scsi generic sg3 type 0
Jan 14 16:50:23 wind kernel: [15346.220433] sd 9:0:0:0: [sdc] 3928064 512-byte logical block
Jan 14 16:50:23 wind kernel: [15346.221161] sd 9:0:0:0: [sdc] Write Protect is off
... weitere Ausgaben ...
```

- ▶ Gerät ist /dev/sdc
- zur Sicherheit sollte man noch Hersteller und Größe prüfen

# Partitionierung mit fdisk

## fdisk - Übersicht

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

fdisk ist ein interaktives Kommandozeilenprogramm

- ▶ vgl. bc, gnuplot
- ▶ nach Eingabe eines Kommandos beginnt ein Dialog mit mehreren Fragen und Vorgaben
- ▶ Hilfefunktion "m" zeigt mögliche Funktionen

```
> sudo fdisk /dev/sdb
```

```
Command (m for help): m
```

```
Command action
```

- a toggle a bootable flag
- b edit bsd disklabel
- c toggle the dos compatibility flag
- d delete a partition
- l list known partition types
- m print this menu

```
...
```

# Partitionierung mit fdisk

fdisk - Partitionen löschen mit "d"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): d
Partition number (1-5): 5
```

```
Command (m for help): d
Partition number (1-5): 2
```

```
Command (m for help): d
Selected partition 1
```

```
Command (m for help): p
```

```
Disk /dev/sdc: 2011 MB, 2011168768 bytes
[... mehr Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

```
Command (m for help):
```

- ▶ alle Partitionen sind jetzt weg

# Partitionierung mit fdisk

fdisk - Neue Partitionstabelle anlegen mit "o"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Hinweis: Wenn man ohnehin *alle* Partitionen löschen will, ist das Anlegen einer neuen Partitionstabelle schneller:

```
Command (m for help): o
```

```
Building a new DOS disklabel with disk identifier 0x490a40e8.  
Changes will remain in memory only, until you decide to write  
After that, of course, the previous content won't be recoverable
```

```
Command (m for help): p
```

```
Disk /dev/sdb: 8004 MB, 8004304896 bytes  
[... mehr Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

```
Command (m for help):
```

# Partitionierung mit fdisk

Ziel: Wir wollen zwei Partitionen auf dem USB-Stick anlegen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

- ▶ eine primäre zum Datenaustausch mit Windows (mit Windows-FAT32-Dateisystem)
- ▶ eine logische zur Nutzung innerhalb von Linux
  - ▶ mit Linux-ext4-Dateisystem
  - ▶ Windows kann dies nicht lesen
  - ▶ aber ext4 kann im Gegensatz zu FAT32 Unix-Berechtigungen speichern

# Partitionierung mit fdisk

fdisk - Primäre Partition anlegen mit "n", "p"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1):
Using default value 1
First sector (2048-3928063, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-3928063,
default 3928063): +512M
Command (m for help): p
```

```
Disk /dev/sdc: 2011 MB, 2011168768 bytes
[... mehr Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		2048	1050623	524288	83	Linux

► primäre Partition mit 512M Größe angelegt

# Partitionierung mit fdisk

fdisk - Erweiterte Partition anlegen mit "n", "e"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): n
```

```
Partition type:
```

```
  p   primary (1 primary, 0 extended, 3 free)
```

```
  e   extended
```

```
Select (default p): e
```

```
Partition number (1-4, default 2):
```

```
Using default value 2
```

```
First sector (1050624-3928063, default 1050624):
```

```
Using default value 1050624
```

```
Last sector, +sectors or +size{K,M,G} (1050624-3928063,  
default 3928063):
```

```
Using default value 3928063
```

```
Command (m for help): p
```

```
[... mehr Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		2048	1050623	524288	83	Linux
/dev/sdc2		1050624	3928063	1438720	5	Extend

- ▶ erweiterte Partition über Rest des Datenträgers angelegt
- ▶ Achtung: Dies ist nur die "Hülle" für log. Partitionen!

# Partitionierung mit fdisk

fdisk - Logische Partition anlegen mit "n", "l"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): n
```

```
Partition type:
```

```
    p   primary (1 primary, 1 extended, 2 free)
```

```
    l   logical (numbered from 5)
```

```
Select (default p): l
```

```
Adding logical partition 5
```

```
First sector (1052672-3928063, default 1052672):
```

```
Using default value 1052672
```

```
Last sector, +sectors or +size{K,M,G} (1052672-3928063,  
default 3928063): +1G
```

```
Command (m for help): p
```

```
[... weitere Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		2048	1050623	524288	83	Linux
/dev/sdc2		1050624	3928063	1438720	5	Extend
/dev/sdc5		1052672	3149823	1048576	83	Linux

- ▶ logische Partition mit 1G Größe angelegt
- ▶ ca. 512M noch frei für weitere log. Partitionen

# Partitionierung mit fdisk

fdisk - Partitions-Typ setzen mit "t"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): t
Partition number (1-5): 1
Hex code (type L to list codes): 0c
Changed system type of partition 1 to c (W95 FAT32 (LBA))
```

```
Command (m for help): p
[... mehr Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		2048	1050623	524288	c	W95 FAT32 (LBA)
/dev/sdc2		1050624	3928063	1438720	5	Extended
/dev/sdc5		1052672	3149823	1048576	83	Linux

- ▶ /dev/sdc1 soll mit FAT32 verwendet werden
- ▶ ist nur Hinweis für Betriebssystem, keine Formatierung!

# Partitionierung mit fdisk

fdisk - Partition bootfähig machen mit "a"

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): a
Partition number (1-5): 1
```

```
Command (m for help): p
```

```
Disk /dev/sdc: 2011 MB, 2011168768 bytes
[... weitere Ausgaben ...]
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1	*	2048	1050623	524288	c	W95 FAT32 (L
/dev/sdc2		1050624	3928063	1438720	5	Extended
/dev/sdc5		1052672	3149823	1048576	83	Linux

Einige (kaputte) Bios-Versionen können Partitionen nur booten, wenn das "Boot"-Flag gesetzt ist.

- ▶ auf der Systemplatte die Partition bootfähig machen, in der /boot enthalten ist!
- ▶ für den Memorystick u. andere Geräte ist dies sinnlos.

# Partitionierung mit fdisk

fdisk - Partitionstabelle mit "w" speichern

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

```
Command (m for help): w
```

```
The partition table has been altered!
```

```
Calling ioctl() to re-read partition table.
```

```
WARNING: If you have created or modified any DOS 6.x  
partitions, please see the fdisk manual page for additional  
information.
```

```
Syncing disks.
```

- vor dem "w" sind Änderungen nur im RAM
- ▶ man könnte es sich jetzt noch anders überlegen
- nach dem "w" wird fdisk beendet

# Partitionierung mit fdisk

Systemplatte partitionieren

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Partition ändern aus der das System gerade läuft:

- ▶ Live-DVD verwenden
- ▶ Daten vorher sichern!

# Partitionen

## Neue Partitionen verwenden

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Wir haben jetzt einen partitionierten Datenträger:

```
>sudo fdisk -l /dev/sdc  
[sudo] password for cg:
```

```
Disk /dev/sdc: 2011 MB, 2011168768 bytes  
62 heads, 62 sectors/track, 1021 cylinders, total 3928064 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x5257b8c3
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1	*	2048	1050623	524288	c	W95 FAT32 (L)
/dev/sdc2		1050624	3928063	1438720	5	Extended
/dev/sdc5		1052672	3149823	1048576	83	Linux

► aber noch keine Dateisysteme darauf!

# Partitionen

Neue Partition mit Dateisystem versehen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

## mkfs.ext4 - Partition mit Ext4-Dateisystem versehen

▶ nennt man auch “initialisieren” / “formatieren”

> `sudo mkfs.ext4 -m 0 /dev/sdc5`

- `-m 0` unterdrückt Reservieren von Blöcken;  
Reservieren ist nur sinnvoll bei der Systempartition

# Partitionen

## Neues Dateisystem einhängen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

### mount - Dateisystem einhängen

```
> sudo mount /dev/sdc5 /mnt
```

```
> df -h
```

Dateisystem	Size	Used	Avail	Use%	Eingehängt a
/dev/sda1	111G	87G	19G	83%	/
[... weitere Ausgaben ...]					
/dev/sdc5	1008M	34M	975M	4%	/mnt

- mount hängt Blockdevice (mit Dateisystem) im Zielverzeichnis ein
- ▶ Verzeichnis kann danach genutzt werden
- Zielverzeichnis muß existieren
- Einhängen bleibt nur bis zum Reboot bestehen

# Partitionen

Neues Dateisystem wieder aushängen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
**mount, umount**  
/etc/fstab  
Label  
mount auf  
Abbildern

## umount - Dateisystem aushängen

```
> sudo umount /dev/sdc5
```

oder

```
> sudo umount /mnt
```

# Partitionen

## Neues Dateisystem aushängen - Problembehebung

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

### Typische Fehlermeldung und Behebung:

```
> sudo mount /dev/sdc5 /mnt
> cd /mnt # Kommandozeile ist im Verzeichnis
> sudo 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))

```
> cd # Kommandozeile aus Verzeichnis nehmen
> sudo umount /mnt
```

- kein Prozeß darf Dateien oder Verzeichnisse im auszuhängenden Verzeichnis mehr geöffnet haben
- ▶ betreffende Programme beenden
- ▶ in allen Kommandozeilen aus (Unter-)Verzeichnissen herausgehen

# Partitionen

Neues Dateisystem permanent einhängen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

## /etc/fstab: Konfigurationsdatei des Dateisystems

```
> cat /etc/fstab
# <file system> <mount point> <type> <options> <dump> <pa
proc                /proc              proc              defaults          0
/dev/sda5           /                  ext4              errors=remount-ro 0
/dev/sda1           /boot              ext4              defaults          0
/dev/sda8           /export            ext4              defaults          0
/dev/sda7           none               swap              sw                0
/dev/scd0           /media/cdrom0     udf,iso9660      user,noauto       0
```

- ▶ Vorsicht: fstab defekt → System bootet ggf. nicht!
- ▶ siehe man fstab, mount

Hinweis: Für Wechseldatenträger ist das Eintragen in der  
fstab unsinnig!

# Partitionen

Neues Dateisystem permanent einhängen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Mountpoint (Zielverzeichnis) anlegen:

```
> sudo mkdir /daten
```

/etc/fstab erweitern:

```
> cat /etc/fstab
```

```
# <file system> <mount point> <type> <options> <dump> <pa  
proc /proc proc defaults 0  
/dev/sda5 / ext4 errors=remount-ro 0  
/dev/sda1 /boot ext4 defaults 0  
/dev/sda8 /export ext4 defaults 0  
/dev/sdc5 /daten ext4 defaults 0  
[... weitere Einträge...]
```

mount einmalig aktivieren: `sudo mount /daten`

- System erst herunterfahren wenn dies fehlerfrei geht
- ▶ ab jetzt wird die Partition bei jedem Start eingehängt

# Dateisystem-Label

Potentielle Probleme bei Geräten mit 2 oder Platten

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

`/dev/sda` und `/dev/sdb` können sich vertauschen:

- Die erste erkannte Platte wird `sda`, die zweite `sdb`
- ▶ Problem wenn Platten nicht immer in der selben Reihenfolge online gehen

`/dev/sda` defekt → `/dev/sdb` wird zu `/dev/sda`

- ▶ in beiden Fällen wird das System nicht mehr starten (z.B. auch wenn Linux nur auf `/dev/sdb` installiert ist)

# Dateisystem-Label

Abhilfe: Dateisystem über "Label" identifizieren

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Beim Neuanlegen des Dateisystems:

```
> sudo mkfs.ext4 -m 0 -L "Daten" /dev/sdc5
```

Nachträglich für ein bestehendes Dateisystem:

```
> sudo e2label /dev/sdc5 "Daten"
```

Danach kann Dateisystem über das "Label" verwendet werden:

```
> sudo mount LABEL=Daten /mnt
```

oder in der `fstab`:

```
LABEL=Daten /daten ext3 defaults 0 2
```

- ▶ Gerätename (`/dev/sdc5`) wird irrelevant
- ▶ alternativer Weg mit zufälligen "Labels": `UUID=...`  
(nutze `blkid` zum Anzeigen der UUID)

# Virtualisierte Festplatten

Wie mountet man Partitionen aus virtuellen Festplatten?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

mount arbeitet auf Dateisystemen (= typischerw. Partitionen),  
*nicht* auf gesamten Festplatten / Partitionstabellen

Virtuelle Festplatten kann man mit fdisk ansehen:

```
> /sbin/fdisk -l ubuntu.img
```

```
Disk ubuntu.img: 17.2 GB, 17179869184 bytes
255 heads, 63 sectors/track, 2088 cylinders, total 33554432 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x9e93ecc4
```

Device	Boot	Start	End	Blocks	Id	System
ubuntu.img1	*	2048	25391103	12694528	83	Linux
ubuntu.img2		25393150	33552383	4079617	5	Extended
ubuntu.img5		25393152	33552383	4079616	82	Linux swap

► wie greift man auf die Partitionen zu?

# Virtualisierte Festplatten

Wie mountet man Partitionen aus virtuellen Festplatten?

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

Wir haben folgende Informationen (aus der `fdisk`-Ausgabe):

```
...
Sector size (logical/physical): 512 bytes / 512 bytes
...
          Device Boot Start          End      Blocks Id System
ubuntu.img1      *    2048 25391103 12694528 83  Linux
```

- die erste primäre Partition beginnt bei Startsektor *2048*.
- ein Sektor entspricht 512 Bytes
- ▶ die erste Partition beginnt in Byte **2048 · 512** in der Abbild-Datei!
- ▶ diesen “Offset” können wir `mount` mitteilen!

# Virtualisierung unter Ubuntu

## Daten aus der VM herausholen

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

1. VM *herunterfahren!*
2. Dateisystem-Abbild der VM mounten:  
> sudo mount ubuntu.img /mnt -r -t ext4 -o  
loop,offset=\$((2048\*512))
3. Daten aus /mnt herauskopieren
4. > sudo umount /mnt

### Erläuterung der mount-Optionen:

- ▶ -r: read-only, wir können lesen, aber nichts verändern
- ▶ -o loop: Das Loop-Device wandelt die Abbild-Datei in ein Blockgerät um, damit mount sie bearbeiten kann.
- ▶ -o ..., offset=...: Das Dateisystem beginnt nicht am Anfang des Blockgeräts, sondern erst die angegebene Anzahl Bytes später!
- ▶ -t ext4: Mit diesem Dateisystem ist die Partition formatiert

# Ende der heutigen Vorlesung

Unix-  
Praktikum

Carsten  
Gnörlich

Werkzeuge

tee  
/dev/shm  
watch

Dateisysteme

Neues Gerät  
dmesg  
/var/log, tail -f  
mkfs.ext4  
mount, umount  
/etc/fstab  
Label  
mount auf  
Abbildern

**Vielen Dank fürs Zuhören!**

**Bis nächste Woche!**