



# SV3 Linux Server

## Dateisysteme Teil 2

# Agenda

## Datenträger

- Linux-Dateisysteme
- Partitionierung von Datenträgern
- Erstellen von Dateisystemen
- Mounten von Dateisystemen

## Dateisystem:

- Das auf dem Rechner vorhandene. Kommandos zum Kopieren von Dateien oder zum Erstellen von Backups, Hintergründe zu den Zugriffsrechten von Dateien etc.

## Netzwerk-Dateisysteme:

- Linux gibt Ihnen die Möglichkeit, Verzeichnisse anderer Rechner in Ihren Verzeichnisbaum zu integrieren.

## Disk-Quotas:

- Dabei handelt es sich um ein System, das steuert, wie viel Platz einzelne Benutzer auf der Festplatte beanspruchen dürfen. Wird die Grenze überschritten, können keine neuen Dateien mehr angelegt werden.

## Cluster-Dateisysteme:

- Cluster-Dateisysteme bzw. globale Dateisysteme verbinden Daten mehrerer Rechner zu einem virtuellen Dateisystem. Damit lassen sich riesige Datenspeicher bilden und von mehreren Rechnern parallel nutzen.

## Linux-Dateisysteme

sind zur Linux Installation und zum Betrieb von Linux geeignet. Die Dateisysteme unterscheiden sich durch Merkmale, die überwiegend für fortgeschrittene Anwender bzw. für den Server-Einsatz interessant sind:

- Geschwindigkeit beim Umgang mit sehr großen oder mit sehr vielen eher kleinen Dateien
- Effizienz bei Schreib- und Lese-Operationen
- CPU-Belastung
- usw. (Mehr Beispiele im Handout)

## Minix file system

ist das Dateisystem des ehemaligen Lehr-Betriebssystems Minix. Ein Minix-Dateisystem besteht aus sechs Bestandteilen:

- **Der Bootsektor** (der immer im ersten Block untergebracht ist. Er enthält den Bootloader)
- **Der Superblock** (der Daten über das Dateisystem enthält)
- **Die Inode-Bitmap** (ist eine einfache Darstellung des Status aller Inodes.)
- **Die Cluster-Bitmap** (genau wie die Inode-Bitmap, stellt jedoch den Status der Cluster dar.)
- **Der Inodes-Bereich** (Jede Datei und jedes Verzeichnis wird von mindestens einem Inode repräsentiert)
- **Der Daten-Bereich** (Hier wird der tatsächliche Inhalt der Dateien und Verzeichnisse gespeichert.)

**ext2:** Lange Jahre war das robuste ext2 das Standarddateisystem unter Linux. Es ist prinzipiell schneller als seine Nachfolger. Allerdings gibt es ein großes Manko: Der Dateisystemtest ist extrem zeitaufwendig

**ext3:** Durch die Einführung einer Journaling-Funktion entfallen die langwierigen Prüfungen des Dateisystems Auf älteren Systemen mit wenig Hauptspeicher immer noch sinnvoll.

**ext4:** Vorteile gegenüber den Vorgängern liegen insbesondere in der aggressiveren Nutzung des Arbeitsspeichers durch Puffer-Mechanismen (Cache) sowie sehr viel schnelleren Dateisystemchecks.

## btrfs

Dieses Dateisystem beinhaltet ein Copy-On-Write System sowie eine neuartige Daten- / Metaverwaltung.

- Wenn es nach dem Willen namhafter Kernelentwickler geht, ist btrfs das Linux-Dateisystem der Zukunft. Das Dateisystem beinhaltet Device-Mapper-, Snapshot- und RAID.
- Leider ist btrfs noch nicht vollständig ausgereift.
- Es wurde für Linux entwickelt und seit 2016, mit der Bezeichnung WinBtrfs, plattformübergreifend für Windows (ab Windows 7) sowie ReactOS weiterentwickelt.



## btrfs

Es soll in Zukunft die bislang im Linux-Umfeld vorherrschenden Dateisysteme ext3 bzw. ext4 mit seinen Beschränkungen (Dateigröße und Gesamtdatensystemgröße) ersetzen. Das Btrfs-Dateisystem ist u.a. auf die folgenden Eigenschaften ausgelegt:

- erweiterte Dateispeicherung (Dateigröße bis  $2^{64}$  Byte)
- effizientere Speicherung kleiner Dateien
- effizientere Verwaltung der Verzeichnisstrukturen
- uvm. (Handout)

## Journaling-Funktionen

Alle gängigen Linux-Dateisysteme unterstützen Journaling-Funktionen. In seiner Journaling einfachsten Form bedeutet Journaling, dass der Beginn und das Ende jeder Dateioperation in einer speziellen Datei mitprotokolliert werden.

- Dank des Protokolls kann später geprüft werden, ob eine bestimmte Dateioperation vollständig ausgeführt wurde.
- Wenn nun eine Dateioperation nicht vollständig abgeschlossen werden kann, geht dies aus dem Protokoll hervor.

## Größenlimits

In der Vergangenheit tauchte immer wieder die Frage auf, wie groß Dateien maximal sein dürfen. Die Antwort hängt davon ab, welchen Kernel, welche CPU-Architektur, welche glibc-Bibliothek und welches Dateisystem Sie verwenden..

Dateisystem	Maximale Dateigröße	Maximale Dateisystemgröße
btrfs	16.777.216 Tbyte	16.777.216 Tbyte
ext3	2 Tbyte	32 Tbyte ( bei 8 kbyte Blockgröße)
ext4	16 Tbyte	1.048.576 Tbyte = 1 Exabyte
xfs	9.437.184 Tbyte	9.437.184 Tbyte
zfs	16.777.216 Tbyte	16.777.216 Tbyte

## Logical Volume Manager (LVM)

Der Logical Volume Manager setzt eine logische Schicht zwischen das Dateisystem und die Partitionen der Festplatte. Was zuerst sehr abstrakt klingt, hat in der Praxis durchaus handfeste Vorteile:

- Im Rahmen des von LVM verwalteten Festplattenbereichs können Sie im laufenden Betrieb ohne Rechnerneustart Partitionen anlegen, vergrößern und verkleinern. Den vorhandenen LVM-Speicherpool können Sie jederzeit durch den Einbau einer weiteren Festplatte vergrößern.
- Sie können dank LVM Bereiche mehrerer Festplatten zu einer einzigen, riesigen virtuellen Partition zusammenfassen.

## Logical Volume Manager (LVM)

Der Logical Volume Manager setzt eine logische Schicht zwischen das Dateisystem und die Partitionen der Festplatte. Was zuerst sehr abstrakt klingt, hat in der Praxis durchaus handfeste Vorteile:

- Sie können sehr einfach einen sogenannten Snapshot eines Dateisystems erstellen. Das ist ideal für Backups im laufenden Betrieb.
- LVM ist sehr schnell. Sie bezahlen für die höhere Flexibilität also nicht mit einer spürbar verringerten Geschwindigkeit. Der Geschwindigkeitsunterschied gegenüber dem direkten Ansprechen einer Festplattenpartition ist kaum messbar. Die CPU-Belastung ist nur geringfügig höher.

## Logical Volume Manager (LVM)

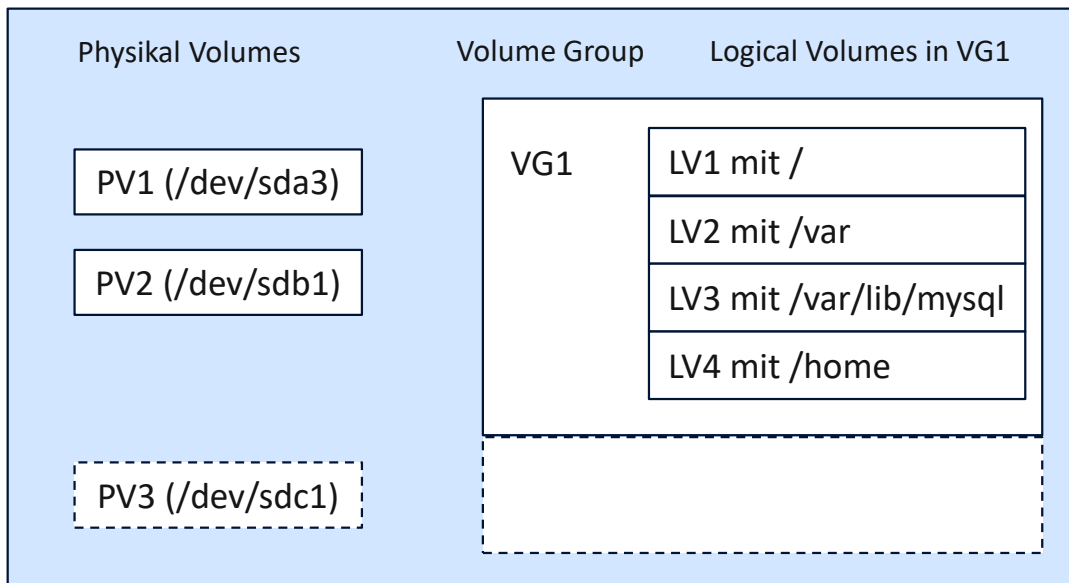
**Physical Volume (PV):** Ein PV ist im Regelfall eine von LVM verwaltete Partition der Festplatte. Es kann sich auch um eine ganze Festplatte oder um ein RAID-Device handeln.

**Volume Group (VG):** Ein oder mehrere Physical Volumes können zu einer Gruppe zusammengefasst werden.

**Logical Volume (LV):** Ein Logical Volume ist ein Teil der Volume Group. Für den Anwender wirkt ein Logical Volume wie eine virtuelle Partition.

## Logical Volume Manager (LVM)

Das Beispiel veranschaulicht die oben definierten Begriffe: Auf Beispiel einem System dienen die beiden Partitionen **/dev/sda3** und **/dev/sdb1** als Physical Volumes für eine Volume Group eines LVM-Systems. **/dev/sda3** umfasst 400 GByte, **/dev/sdb1** umfasst 900 GByte.



Der LVM-Speicherpool (also die Volume Group) ist somit 1,3 TByte groß. Darin befinden sich nun diverse Logical Volumes:

- LV1 mit der Systempartition (50 GByte)
- LV2 mit der Partition **/var** (200 GByte)
- LV3 mit der Partition **/var/lib/mysql** (200 GByte)
- LV4 mit der Partition **/home** (400 GByte)

## **Der Swapspeicher (oder Auslagerungsspeicher), kurz Swap**

ist eine Datei oder eine Partition, die die Kapazität des Arbeitsspeichers erweitert. Zudem ermöglicht dieser Speicherplatz es, bei ausreichend großer Kapazität, in den Suspend-to-Disk-Modus zu wechseln.

- Dieses Verfahren zur Vergrößerung des Arbeitsspeichers wird als Paging bezeichnet. Wenn der physikalische Speicher knapp wird, werden Memory-Pages (Speicherseiten) auf die Festplatte ausgelagert.



## Der Swapspeicher (oder Auslagerungsspeicher), kurz Swap

Beispielsweise empfiehlt Oracle für seinen Datenbankserver je nach verfügbarem RAM unterschiedliche Faktoren zur Berechnung der Swap-Größe:

Arbeitsspeicher	Swap-Speicher
Bis 2 GB	Faktor 1,5
2 bis 16 GB	Faktor 1
Mehr als 16 GB	16 GB oder deaktiviert

## Fstab (file systems table)

Die Datei `fstab` dient zum Einbinden der Datenträger unter Linux, sie befindet sich in der zentralen Datei `/etc/fstab`. Dort stehen die Datenträger, die man automatisch beim Starten einhängen (`mounten`) will, oder die man nachträglich mit einem verkürzten `mount` Befehl einbinden will.

- Die `fstab` wird jedes Mal beim Neustarten des PCs ausgelesen.
- Die eigenen Linux-Partitionen müssen in der `fstab` vorhanden sein, ansonsten bootet Linux nicht.
- Andere Datenträger wie ein DVD oder CD-Laufwerk müssen nicht unbedingt eingetragen sein.

## Fstab (file systems table)

**Automatisches mounten oder nicht:** In der fstab kann man festlegen, ob die Laufwerke automatisch beim Start des Rechners oder manuell eingebunden werden sollen. Für die letztere Variante setzt man den Parameter auf "noauto".

**Netzlaufwerke:** In fstab können auch Netzlaufwerke aufgelistet werden.. Hat man einmal die lan-Befehle in der fstab stehen, kann man das Netz automatisch einbinden oder es reicht ein kurzer Mount Befehl beispielsweise "rechnername:/verzeichnis".

**Aktualisierung von fstab:** Wenn man mit mount ein komplett neues Filesystem einhängt, ändert sich fstab nicht automatisch.

## Fstab (file systems table)

Sie können sich die `fstab` mit dem folgenden Befehl auf Ihrem Rechner ansehen:

```
cat /etc/fstab
```

```
sb@ub:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>    <type>  <options>          <dump>  <pass>
UUID=c20aef0f-47bf-4b1e-82e1-db6b039905f9 /        ext4      errors=remount-ro  0        1
UUID=4E89-7C53 /boot/efi vfat      umask=0077         0        1
/swapfile none      swap      sw                 0        0
sb@ub:~$
```

## Filesystem

Hier wird die Gerätedatei angegeben.

- `/dev/hda1` - Erste Festplatte, mit der ersten Partition.
- `/dev/sda1` – Erste Festplatte, erste Partition (SATA, SCSI)
- `/dev/fd0` – Floppy
- UUID der Festplatte

## Mountpoint (Mountpunkt, Einhängepunkt)

Hier wird das Verzeichnis angegeben, in das der Datenträger eingebunden werden soll.

- /mnt/hdd-fat - Standard Mountpunkt für Festplatten
- /mnt/dvd-rom - Standard Mountpunkt für ROM-Laufwerke
- /mnt/floppy0 - Standard Mountpunkt für Floppies

## Type (fs\_vfstype)

Hier wird das Filesystem angegeben

- ext2 - Für das Dateisystem ext2
- ext3 - Für das Dateisystem ext3
- ...

## Optionen (fs\_mntops)

- auto - Automatisches Einhängen
- acl - aktiviert die Access Control List um Benutzerrechte an Dateien genauer zu kontrollieren.
- sw – Swap
- ...



## **dump (fs\_freq)**

Hier sind Informationen für das Programm dump enthalten, die inzwischen aber ignoriert werden, da es das Programm dump nicht mehr gibt.

- 0 - Für alle Partitionen, die keine Systempartitionen sind.
- 1 - Steht üblicherweise für die Systempartitionen.

## **pass (fs\_passno)**

Hier wird die Reihenfolge für die Überprüfung der Dateisysteme bestimmt, die beim Systemstart durchgeführt wird.

- 0 - Das Dateisystem wird nicht überprüft!
- 1 - wird als erstes überprüft, empfohlen für das root-Dateisystem '/'.
- 2 - werden zuletzt überprüft.

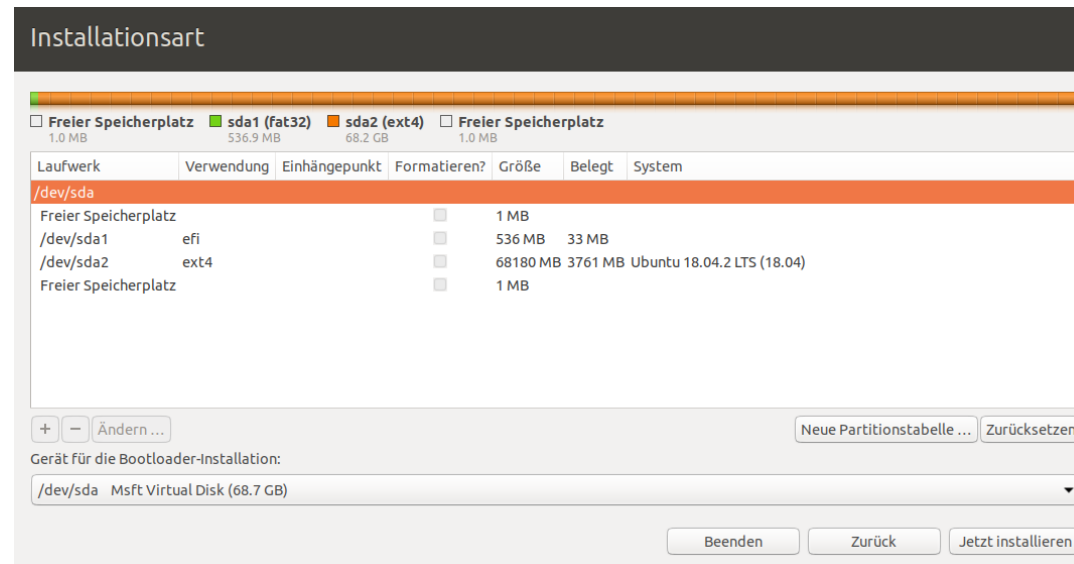
## Änderungen von fstab

fstab kann mit jedem Editor bearbeitet werden (Es gibt aber auch GUI-Anwendungen zur fstab-Änderung. Dazu brauchen Sie allerdings root-Rechte.

- Erstellen Sie aber am besten eine Sicherungskopie bevor Sie die Originaldatei ändern.
- Die Änderungen von fstab werden erst beim Neustart des PCs wirksam oder durch Eingabe von `mount -a`

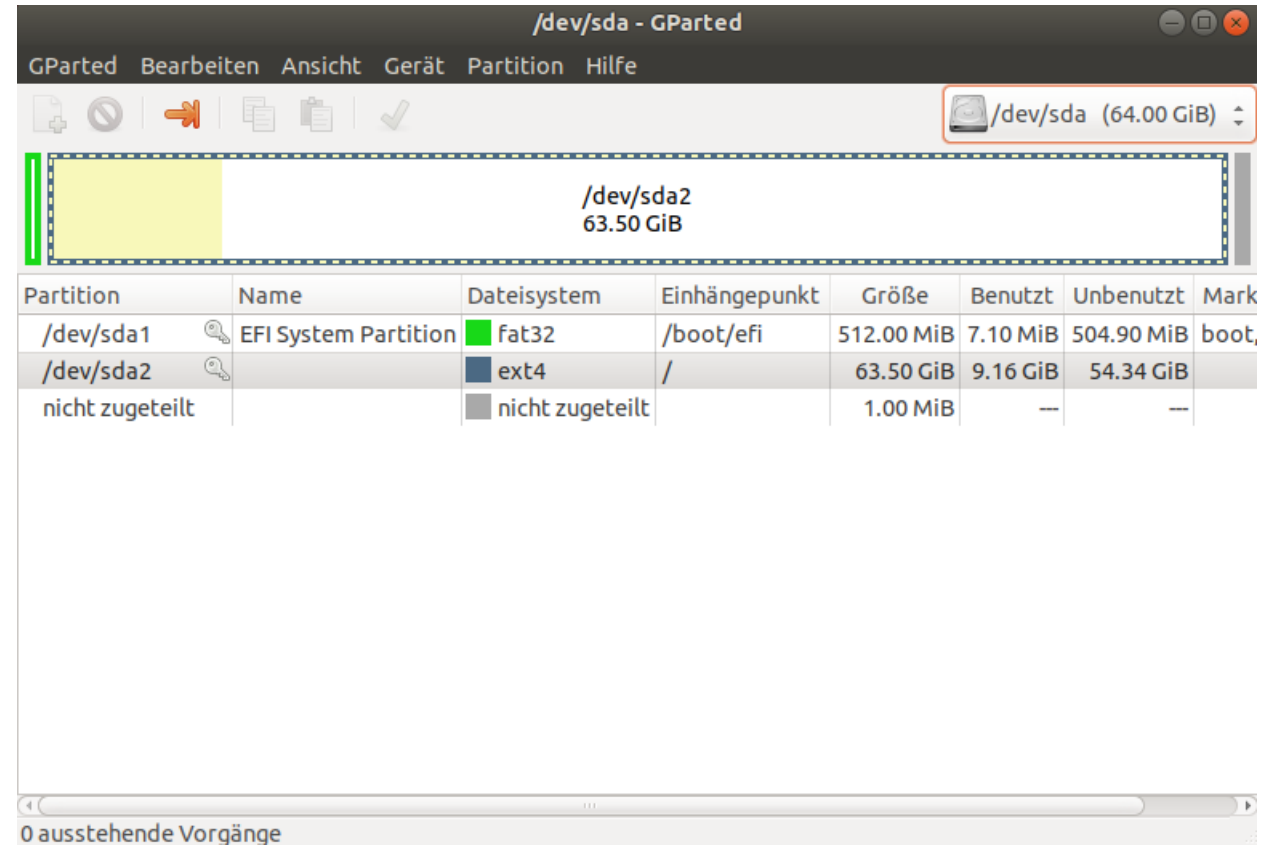
## Partitionierung der Festplatte

Einer der wichtigsten Schritte während der Linux-Installation ist das Anlegen neuer Linux-Partitionen. Alle gängigen Installationsprogramme enthalten zu diesem Zweck einfach zu bedienende Partitionierungshilfen



## Partitionierung der Festplatte

Auch unter der Grafischen Oberfläche findet man Programme, die einem hier die Arbeit erleichtern, z.B. GParted



## Partitionierung der Festplatte

Bei jeder Linux-Installation benötigen Sie eine Systempartition. Darüber hinaus ist eine Swap-Partition sehr zu empfehlen.

- Das Einrichten weiterer Partitionen ist optional, sehr stark von der geplanten Anwendung von Linux abhängig und auch eine Geschmacksfrage.

Verzeichnis	Verwendung
<b>/boot/efi</b>	EFI-Partitionen (Bei EFI-Systemen, ca. 200 MB)
<b>/boot</b>	BIOS-GRUB-Partition (1MB, nur für die Kombination BIOS + GPT)
<b>Swap</b>	Swap-Partition (etwas größer als Ram am besten)
<b>/</b>	Systempartition (mindestens 20 GB)
<b>/home</b>	Datenpartition (Größe je nach voraussichtlicher Nutzung)

## **fdisk (fixed disk)**

Ein Relikt aus der Vergangenheit. Als Alternative zu parted ist bis heute auch fdisk beliebt.

- Da sich fdisk aber nur für Datenträger mit MBR-Partitionen eignet, arbeiten wir mit parted oder alternativ gdisk.

```
fdisk /dev/GERÄT  
fdisk /dev/sda
```

## cdisk (ncurses fixed disk)

Eine komfortable Möglichkeit eine Festplatte in der Shell zu formatieren ist cfdisk.

Dieses Programm kann nur mit Root-Rechten ausgeführt werden.

- Cfdisk kann Partitionen erstellen, verwalten und mit einem Dateisystem versehen (formatieren).

```
sb@ub: ~  
Datei Bearbeiten Ansicht Suchen Terminal Hilfe  
  
Festplatte: /dev/sda  
Größe: 64 GiB, 68719476736 Bytes, 134217728 Sektoren  
Bezeichner: gpt, Identifikator: A41D1493-25D0-4AEB-B96C-4EF09338596B  


| Gerät        | Anfang  | Ende      | Sektoren  | Größe | Typ               |
|--------------|---------|-----------|-----------|-------|-------------------|
| >> /dev/sda1 | 2048    | 1050623   | 1048576   | 512M  | EFI-System        |
| /dev/sda2    | 1050624 | 134215679 | 133165056 | 63,5G | Linux-Dateisystem |


Partitionsname: EFI System Partition  
Partitions-UUID: D402F234-0D3A-4BAB-A4C3-E0ECE05274C6  
Partitionstyp: EFI-System (C12A7328-F81F-11D2-BA4B-00A0C93EC93B)  
Dateisystem-UUID: 4E89-7C53  
Dateisystem: vfat  
Einhängpunkt: /boot/efi (eingehängt)

  
[ Löschen ] [ Größe verändern ] [ Ende ] [ Typ ]  
[ Hilfe ] [ Schreib. ] [ Speichern ]  
  
Beende das Programm ohne die Partitionstabelle zu schreiben
```



**Beim "Formatieren" eines Datenträgers wird dieser mit einem Dateisystem versehen.**

- Es wird also festgelegt, mit welcher Struktur die Daten abgelegt werden.
- Vor dem Formatieren muss man in der Regel eine Partition anlegen, die anschließend formatiert wird.

## Mkfs (make filesystem)

Es gibt für jedes Dateisystem ein eigenes Kommando für die Formatierung. Die Befehle sind aber alle nach dem gleichen Schema benannt:

```
mkfs.ext3 OPTIONEN DEVICE  
mkfs.ext3 /dev/sda2
```

ext3 kann dabei natürlich auch durch die anderen Dateisysteme ersetzt werden. Die Angabe des DEVICE ist zwingend.

## mount

Nach dem Formatieren müssen Sie die Partition in die fstab eintragen damit sie automatisch beim Systemstart gemounted wird.

- Diese können Sie aber auch manuell mounten, das Dienstprogramm zum Einhängen auf der Kommandozeile heißt Mount.

```
mount /dev/sdc1 /media/usb  
mount /dev/sdd1 /mnt/speicher
```



**VIELEN DANK  
FÜR IHRE  
AUFMERKSAMKEIT!**