

10) Dateisysteme

-Dateistrukturen:

- 1) Bytefolge: Datei als unstrukturierte Folge von Bytes. BS kennt den Inhalt nicht
- 2) Satz-Dateistruktur: Datei als Folge von Sätzen fester Länge. BS kennt Satzlänge
Lese/Schreib/Änderungsoperationen nur auf ganze Sätze anwendbar. Satznummern
- 3) Baum von Sätzen: Baum von Sätzen mit Schlüsselfeld. BS sieht Satzlänge
Operationen: - Satz bei gegebenem Schlüsselfeld suchen
- **Ordnungserhaltendes** Hinzufügen eines Satzes

- Dateizugriff: Sequenziell: - Verarbeitung von Anfang bis Ende
- Zurückspulen auf Dateianfang (Abstraktion Magnetband)
- Direkt : - Bytes/Sätze können beliebig gelesen/geschrieben werden
-alternativ kann die Position (z.B. seek()) gesetzt werden

- Dateiattribute: Typ, Zugriffsrechte, # Referenzen, Größe, Datum (Zugriff, Erstellung).
-möglich Attribute: Schutz, Passwort, Eigentümer, Urheber, Hidden, Readonly

- memory-mapped files: -Memory Mapping von Dateien bezeichnet das Ein- & Ausblenden von Dateien in den Adressraum eines Prozesses
- Vorteil: Dateizugriff mit normalen Befehlen (kein read/write)

- Verzeichnisse (Strukturierung von Dateien. Verzeichniss selbst besteht aus Dateien)
- Verzeichnissysteme mit einer Ebene (1 Verzeichnis pro Laufwerk oder Benutzer)
- hierarchisch: bestehend aus Verzeichnis, Datei, Symbolic Link (azyklischer Graph)

- Montierbare Verzeichnisbäume: UNIX mount / umount, Windows Laufwerksbuchstaben

- Festplatte: -5 bis 10 k RPM, $n * 100$ Mbit/s, Mittlere Positionierungszeit: < ca. 6 ms
- Sektoren, Spuren, Lücken (auch gaps oder Servo tracks genannt)
- $LBA = (C * Nheads + H) * Nblocks_per_track + S$ (heute wegabstrahiert)

- Halbleiterspeicher (NAND-Flash, Speicherung mithilfe von Transistor-Gates)

-Platzverwaltung:

- Partitionierung: i.d.R wird der gesamte Platz auf mehrere Partitionen verteilt
- MBR (master boot record) enthält ausführbaren Code, wird beim Booten gestartet
- logische Laufwerke: dynamisch veränderbare Partitionierung, die sich auch über mehrere physische Datenträger hinweg erstrecken kann.
- Linux LVM: physical volumes werden zu volume groups zusammengefaßt
Auf einer Laufwerksgruppe können logische Laufwerke eingerichtet (entspricht Partitionierung) und mit einem Dateisystem versehen werden. Im laufenden Betrieb kann die Kapazität der Laufwerksgruppe durch Hinzufügen weiterer physischer Volumes vergrößert werden. können Daten von alten Laufwerken auf neue verlagert und die alten Laufwerke außer Betrieb genommen werden. kann logischen Laufwerken mehr Speicherplatz zugeordnet werden oder Speicherplatz entzogen werden. LVM unterstützt „Filesystem Snapshots“ Beim Anlegen eines Snapshots wird ein neues logisches Laufwerk angelegt, das den momentanen Zustand seines zugehörigen Ursprungs-Laufwerks enthält (eingefrorene Sicht, keine Kopie)
→ Ermöglicht konsistente Backups über Snapshot-Laufwerk trotz weiterlaufenden Betriebs auf dem ursprünglichen Laufwerk

RAID (Redundant Array of Independent [Inexpensive] Disks)

- Realisierungen: -Hardware-RAID (spezieller Festplatten Controller)
 -Software-RAID (BS verwaltet Platten als RAID)
- Ziele: -Erhöhung der Datensicherheit
 -Austausch defekter Platten im laufenden Betrieb ohne Unterbrechung („hot-standby“-Platte)
 -Verteilung der Daten durch RAID level (0-6) definiert

-RAID0 – „stripping“

RAID0 - „stripping“

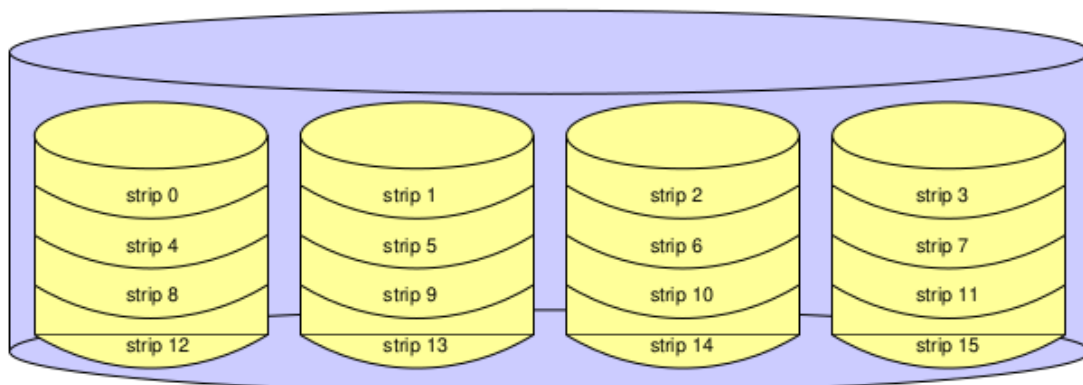


RAID-Platte wird in „Streifen“ mit k Blöcken eingeteilt

Streifen werden reihum auf den angeschlossenen Platten abgelegt.

- **keine Redundanz**, damit keine höhere Fehlertoleranz
- **schneller Zugriff** besonders bei großen Dateien, da Platten parallel arbeiten können

RAID-Kapazität: Summe der Plattenkapazitäten



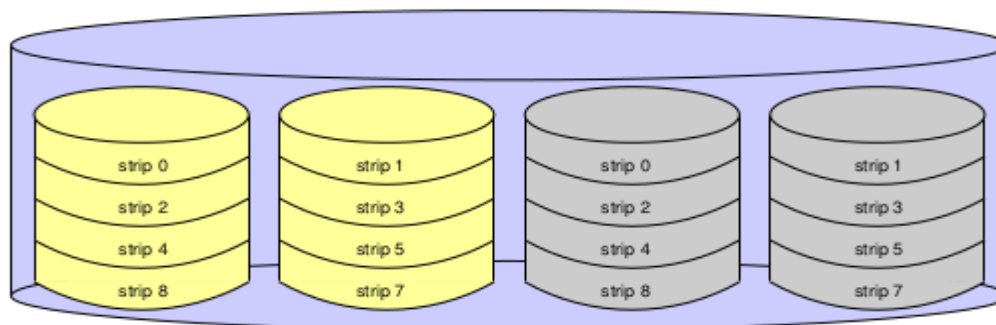
RAID1 - „mirroring“

Zu jeder Platte gibt es eine Spiegelplatte gleichen Inhalts

Fehlertoleranz: Wenn eine Platte ausfällt, kann andere sofort einspringen (übernimmt Controller automatisch)

Schreiben: etwas langsamer; **Lesen:** schneller durch Parallelzugriff auf beide zuständigen Platten

RAID-Kapazität: Hälfte der Summe der Plattenkapazitäten



RAID5 - parity

Paritätsinformation (XOR) auf alle Platten verteilt

Beispiel: P 0-2 enthält XOR-Verknüpfung über die Streifen 0, 1, 2

XOR Verknüpfung ist „selbstinvers“:

Wenn gilt: $P = A \oplus B \oplus C$

dann ist: $A = B \oplus C \oplus P$

und: $B = A \oplus C \oplus P$

und: $C = A \oplus B \oplus P$

+ Fehlertolerant bei guter Kapazitätsnutzung
+ Leseoperationen schnell
- Schreiben aufwändiger

→ Solange maximal eine (beliebige) Platte ausfällt, kann ihr Inhalt aus den Übrigen (im lfd. Betrieb) rekonstruiert werden (wieder per XOR)

