

# Kapitel 1

## Dateisysteme

### 1.1 Vom Speichermedium zur Datei

Physisches Speichermedium und Datei sind die beiden extreme bei der Betrachtung der Speicherung von Daten. Die Datei kann man als die kleinste zusammenhängende Menge an Daten betrachten. Das Speichermedium als extrem auf der anderen Seite ist der größte physikalisch zur Verfügung stehende Speicher für Daten. Um die Speicherung der Datei auf dem Speichermedium zu ermöglichen, müssen wir uns mit einigen grundlegende Fragen beschäftigen:

- Wie kann das Betriebssystem die Daten auf einem Speichermedium finden? Wie können somit mehrere Dateien konkurrenzfrei abgelegt werden? (-> Adressierung)
- Wie groß ist der Aufwand für die Adressierung im Verhältnis zu den zu speichernden Daten? Wie groß muss der Adressraum sein, um eine bestimmte Speichergröße bedienen zu können?
- Wenn es physikalisch technisch unterschiedliche Arten der Speicherung und Adressierung gibt, kann dann das Betriebssystem dies abstrahieren und vor dem Nutzer verstecken?
- Unterscheidet sich die Art der Speicherung und damit auch die Adressierung bei verschiedenen Speichermedien?

### 1.1.1 Serielle und sequenzielle Dateisysteme

Möchten wir Daten auf einem Streifen Papier speichern, so müssen wir uns bezüglich der Speicherung wenig Gedanken machen. In einem bestimmten Abstand bringen wir Löcher (0) oder keine Löcher (1) an und sind damit in der Lage Daten in einer Zeile zu speichern. Der Speicherplatz ist nur beschränkt durch die Länge des Papierstreifens. Nach diesem Prinzip funktionierten in früheren Zeiten auch Lochkarten.

Der Sprung zum Bandlaufwerk ist an dieser Stelle nicht mehr weit. Die Speicherung erfolgt dabei nicht mehr mechanisch, sondern elektromagnetisch. Auch auf Magnetbändern werden die Daten hintereinander geschrieben.

Was könnte nun ein Nachteil dieser gereihten Anordnung der Daten sein?

- Wie wird auf eine Information in der Mitte oder am Ende zugegriffen?
- Bei einer linearen Anordnung muss das Band/Lochkarte immer genau bis zu dem Punkt vorlaufen, auf dem die Information hinterlegt ist.
- Meist ist ein lineares Einlesen aller Informationen bis zur gewünschten Stelle notwendig.
- Zeitintensiv und mechanisch aufwändig.
- Der Anfang eines Magnetbandes wird automatisch öfters belastet, als das Ende, was zu erhöhtem Verschleiss führt.
- Die Anordnung der Dateien ist simpel/der Reihe nach.
- Daten können bitweise geschrieben und gelesen werden.
- Was passiert beim Löschen von Daten und dem Einfügen von Daten?  
Das Einfügen von Daten in Lücken, an denen etwas gelöscht wurde ist problematisch. Das Einfügen ist in der Regel immer nur am Ende sinnvoll.

Sogenannte **serielle Dateisysteme**

sind die Grundlage für Bandlaufwerke. Darin

werden sequentiell adressierbare Bereiche definiert, also Abschnitte die mit Daten beschrieben werden können. Die Information in welchem Abschnitt welche Daten gespeichert sind, wird in einer Sequenz ganz zu Beginn des Bandes hinterlegt. Um zur Sequenz zu springen, müssen erst alle Sequenzen davor (mechanisch) durchlaufen werden.

### 1.1.2 Direktadressierbare Speichermedien

Von dieser seriellen Anordnung der Daten weichen aktuelle Speichermedien wie Festplatten und CD/DVD ROMS und auch Disketten gravierend ab. Die Daten werden auf rotierenden Scheiben hinterlegt. Der Lesekopf muss nur entlang der radialen Achse verschoben werden und warten bis der richtige Speicherbereich vorbeidreht, um gelesen oder beschrieben zu werden. Ein sequenzielles Abfahren bis zum Ziel ist nicht mehr notwendig.

Damit wird ein wahlfreier Zugriff möglich, da, mit einer kleinen Latenzzeit, ein direkter Zugriff auf beliebige Positionen des Speichermediums möglich ist. Dies erlaubt es nun auch effizienter mit dem Speicherplatz umzugehen und komplexere Konzepte der Datenspeicherung zu realisieren, da die Restriktion der Zugriffszeiten, die bei linearen Systemen noch das Hauptproblem darstellten, nun weggefallen sind. So war es z.B. vormals ein Problem Daten an einer Stelle zu löschen und diese Stelle für andere Daten zu benutzen: passen die neuen Daten überhaupt in die Lücke, die frei geworden ist? Es galt also effizienter Speicherplatz gegen Zugriffszeiten abzuwägen.

Bei heutigen Verzeichnissystemen kommen in der Regel folgende Konzepte zum Einsatz:

- Unterteilung des Mediums in Blöcke meist zw. 512 und 2048 Bytes zur Reduzierung des Adressierungsaufwands.
- Zusammenfassung von 8 oder mehr Blöcken zu Clustern (Verbesserung der Adressierung)
- Dateien können auf Cluster verteilt werden
- Adressdaten mit Verweisen auf die Cluster und/oder Blöcke werden in einen speziellen Bereich des Speichermediums geschrieben und dort vom OS verwaltet.
- Spezieller Bereich zur Information über Belegung oder Nicht-Belegung: Cluster Block Availability Map (BAM)
- Verwaltung der Dateien in hierarchischen Verzeichnisstrukturen

Wie diese Beispiele und Möglichkeiten moderner Dateisysteme zeigen, müssen wir bei der Dateiverwaltung zwischen unterschiedlichen Ebenen unterscheiden:

**Physikalisch:** Wie werden Daten auf dem Medium hinterlegt  
und wie erfolgt die Adressierung und Verteilung der  
Daten auf dem Medium

**Konzeptionell:** Wie werden die Verwaltungsinformationen auf dem  
Medium hinterlegt (Adressen, Dateinamen, Metadaten,  
Informationen zu freiem Speicherplatz

**Logisch:** Wie werden die Daten dem Nutzer präsentiert  
(Dateiorganisation, Verzeichnisstruktur und Hierarchie,...)

Moderne Betriebssysteme bilden auf dieser Basis verschiedene Abstraktionsebenen, die den Nutzer davon befreien, sich über Strukturen auf dem Speichermedium Gedanken machen zu müssen.

- Der Dateisystemstandard definiert, wie auf einem Medium die Daten abgelegt und verwaltet werden. Oftmals unterscheiden sich die Konzepte von Medium zu Medium.
- Das Betriebssystem bietet eine Schnittstelle, um unterschiedliche Dateisysteme nutzen zu können und kümmert sich um die korrekte physikalische Ablage der Daten und Metadaten
- Der Nutz sieht nur noch einen Verzeichnisbaum mit Dateien als logische Organisation

### 1.1.3 Unterteilung von Speichermedien

Die Festplatte als physisch reales Speichermedium ist die größte mögliche Einheit zur Speicherung von Daten. Betriebssysteme nutzen aber noch eine Zwischenschicht: die Partition.

Die Unterteilung eines physischen Speichermediums in mehrere logische Speichermedien kann aus unterschiedlichen Gründen sinnvoll sein:

- Das Betriebssystem bzw. verfügbare Dateisystem unterstützt nicht die Adressierung eines so großen Speichermediums (beschränkter Adressraum) (früher ein Problem)
- Es sollen zur Speicherung unterschiedlicher Daten auch unterschiedliche Dateisysteme zum Einsatz kommen
- Es sollen zwei unterschiedliche Betriebssysteme installiert werden können, die aber verschiedene Dateisysteme erfordern bzw. nicht auf dem gleichen Speichermedium installiert werden können/dürfen
- Betriebssysteme erfordern mehr als ein Speichermedium, um die unterschiedlichen Speichermedien für verschiedene Zwecke einzusetzen (z. B. Systempartition und Swap-Partition unter Linux)

Ein reales Speichermedium lässt sich damit in mehrere virtuelle Speichermedien, sogenannte Partitionen, unterteilen. Hierzu wird in einem definierten Bereich des realen Speichermediums ein Datum hinterlegt, das über die Art der Aufteilung in Partitionen informiert. Dies hat an zwei Stellen eine wichtige Bedeutung:

- Beim Anschluss eines Speichermediums muss das Betriebssystem die Partitionsaufteilung und die jeweiligen Dateisysteme der Partition erkennen können.
- die Firmware muss im Bootstrapping einen Anknüpfungspunkt für das Betriebssystem (Speicherort) finden

## Aufgabe: Dateisysteme in der Anwendung

Recherchieren Sie im Internet nach einem der folgenden Dateisysteme und tragen Sie die wesentlichen Informationen zu diesem System zusammen. Erstellen Sie eine kurze Präsentation und stellen Sie das Dateisystem, dessen Stärken und Schwächen sowie die Entwicklung den anderen Kursteilnehmern vor.

Folgende Dateisysteme stehen zur Wahl:

- ISO9660, CDFS, Joliet
- HFS, HFS+, HFSX
- ext2, ext3 und ext4
- FAT16, FAT32, VFAT
- NTFS und ReFS
- ~~exFAT~~
- ~~NFS~~
- ~~SMB~~

## 1.2 Spezielle Konzepte

### 1.2.1 Hierarchische Dateisysteme

- Je mehr Dateien verwendet werden, um mso mehr Ordnungssystematik benötigt man
- Verwaltung von Dateien in Verzeichnissen als Ordnungsstruktur in hierarchischen Bäumen
- Einrichtung eines Namensraums, in denen Daten strukturiert hinterlegt werden (Ansprechen über Pfadnamen)

### 1.2.2 Netzwerkdateisysteme

- Standardisierter Zugriff auf Dateisysteme, die auf einem anderen/zentralen Rechner liegen.
- Unter Unix lokal in Verzeichnisbaum eingebunden (für Nutzer fast nicht erkennbar) (Befehl: \$ mount)
- SMB unter Windows für Netzlaufwerke oder NFS für Unix
- WebDAV unter http / https

### 1.2.3 Virtuelle Dateisysteme

- Konzept zur Nutzung des Dateisystemkonzepts aus der OS Logik, um auf andere Informationen, Daten oder Systemkomponenten zuzugreifen (\$ ls /proc)
- z. B. Dateisystem im Arbeitsspeicher für schnelleren Zugriff auf Daten (Bootprozess)
- Zugriff auf Daten von Devices, in dem ein Dateisystem vorgegaukelt wird (RAM Disks, Linux: tmpfs)

### 1.2.4 Journaling Dateisysteme

- Änderungen im Dateisystem bedeuten immer Änderungen an mehreren Stellen auf dem Datenträger
- Annahme, dass Vorgang abgeschlossen, wenn Datenträger von einem konsistenten Zustand in den anderen überführt wurde.
- Was passiert bei Abbruch während des Schreibens?
- > Vor dem Schreibprozess werden Änderungen in einen Journal-Bereich geschrieben; erst dann erfolgt die Änderung am Datenträger
- Metadatenjournaling: Konsistenz des Dateisystems
- Fulljournaling: Konsistenz Dateisystem + Dateinhalt
- Schnellere Konsistenzprüfung der Daten möglich (z. B. beim Systemstart)
- Verhindert Datenverlust bei plötzlicher Unterbrechung im Schreibvorgang

