



universität  
**uulm**



Betriebssysteme SS2024

# Dateisysteme

Tutorium 07

# Inhaltsverzeichnis

1 Eigenschaften von Dateien

2 Linux Dateibäume

3 Inodes

# Dateien

Typische Attribute:

- Speichern Daten entweder
  - **unstrukturiert** als Folge von Bytes
  - **strukturiert** in mehreren Datensätzen
- Der Zugriff auf die Daten kann entweder
  - **sequentiell** erfolgen wobei die Datei von Anfang bis Ende gelesen wird
  - **wahlfrei** erfolgen und an einer beliebigen Position in der Datei stattfinden

## Dateien (2)

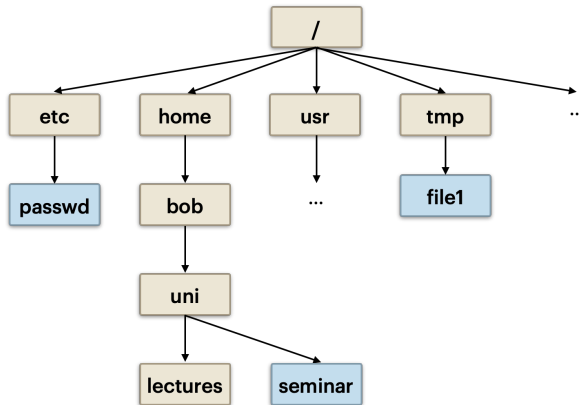
Typische Operationen auf Dateien:

- Öffnen (`open`), Schließen (`close`), Erzeugen (`create`), Löschen (`delete`)
- Lesen (`read`), Schreiben (`write`)
- Positionieren des Schreib-Lesezeigers (`seek`)

# Linux Dateibäume

- (Linux) Dateisysteme sind in einer Baumstruktur organisiert
- Die inneren Knoten des Baumes repräsentieren immer Verzeichnisse und die Blätter sind entweder leere Verzeichnisse oder Dateien
- Die Verbindungen zwischen den Knoten werden benannt wodurch Pfade angegeben werden können
- Spezielle Verweise:
  - “/” verweist auf das Wurzelverzeichnis
  - “.” verweist auf das aktuelle Verzeichnis
  - “..” verweist auf das Elternverzeichnis

# Linux Dateibäume - Beispiel



**Abbildung:** Vereinfachter Dateibaum wobei Verzeichnisse gelb und Dateien blau gefärbt sind

# Pfadnamen

- Pfade können entweder *absolut* sein, d.h. sie beginnen beim Wurzelverzeichnis, oder *relativ* zum aktuellen Verzeichnis angegeben werden
- Beispiele:
  - Absoluter Pfad zur Datei `file1`:

# Pfadnamen

- Pfade können entweder *absolut* sein, d.h. sie beginnen beim Wurzelverzeichnis, oder *relativ* zum aktuellen Verzeichnis angegeben werden
- Beispiele:
  - Absoluter Pfad zur Datei `file1`: `/tmp/file1`



# Pfadnamen

- Pfade können entweder *absolut* sein, d.h. sie beginnen beim Wurzelverzeichnis, oder *relativ* zum aktuellen Verzeichnis angegeben werden
- Beispiele:
  - Absoluter Pfad zur Datei `file1`: `/tmp/file1`
  - Relativer Pfad zur Datei `seminar` ausgehend von `/home`:

# Pfadnamen

- Pfade können entweder *absolut* sein, d.h. sie beginnen beim Wurzelverzeichnis, oder *relativ* zum aktuellen Verzeichnis angegeben werden
- Beispiele:
  - Absoluter Pfad zur Datei `file1`: `/tmp/file1`
  - Relativer Pfad zur Datei `seminar` ausgehend von `/home`: `bob/uni/seminar`
  - Relativer Pfad zur Datei `file1` ausgehend von `/usr`:

# Pfadnamen

- Pfade können entweder *absolut* sein, d.h. sie beginnen beim Wurzelverzeichnis, oder *relativ* zum aktuellen Verzeichnis angegeben werden
- Beispiele:
  - Absoluter Pfad zur Datei `file1`: `/tmp/file1`
  - Relativer Pfad zur Datei `seminar` ausgehend von `/home`: `bob/uni/seminar`
  - Relativer Pfad zur Datei `file1` ausgehend von `/usr`: `../tmp/file1`

# Verweise

Bei Verweisen unterscheidet man zwischen:

- Hard Links

# Verweise

Bei Verweisen unterscheidet man zwischen:

- Hard Links
  - Absoluter Verweis auf eine Datei
  - Mehrere Verweise pro Datei möglich
  - Eine Datei wird erst gelöscht wenn der letzte *Hard Link* entfernt wurde
  - Nicht für Verzeichnisse anlegbar

# Verweise

Bei Verweisen unterscheidet man zwischen:

- Hard Links
  - Absoluter Verweis auf eine Datei
  - Mehrere Verweise pro Datei möglich
  - Eine Datei wird erst gelöscht wenn der letzte *Hard Link* entfernt wurde
  - Nicht für Verzeichnisse anlegbar
- Symbolic Links

# Verweise

Bei Verweisen unterscheidet man zwischen:

## ■ Hard Links

- Absoluter Verweis auf eine Datei
- Mehrere Verweise pro Datei möglich
- Eine Datei wird erst gelöscht wenn der letzte *Hard Link* entfernt wurde
- Nicht für Verzeichnisse anlegbar

## ■ Symbolic Links

- Relative Referenz auf eine Datei oder ein Verzeichnis
- Können über Dateisysteme hinweg angelegt werden
- Löschen einer referenzierten Datei führt zu einem kaputten *Symbolic Link*

# Inodes

- Verwalten Metadaten für Dateien und Verzeichnisse
- Enthalten Infos über:
  - Typ des Inodes
  - Länge der Daten
  - Ortsinformationen für Datenblöcke
  - Eigentümer und die Berechtigungen
  - Verschiedene Zeitstempel
- Bei den Ortsinformationen sind mehrere Indirektionsstufen möglich
  - Beliebige große Dateien adressierbar
  - Schnelle Positionierung des Schreib-/Lesekopfes möglich