

# Aufbau Datenbanksysteme

## Datenbanktechnologien

Prof. Dr. Ingo Claßen   Prof. Dr. Martin Kempa

Hochschule für Technik und Wirtschaft Berlin

Überblick

Speicher

Datenbankpuffer

Speicherstrukturen für relationale Daten

Zugriffsstrukturen

Systemtabellen

# Systemarten

- ▶ In-Memory versus Disk-based Systeme
- ▶ Zeilenorientierte versus spaltenorientierte Speicherung
- ▶ Zeilenorientierte Speicherung

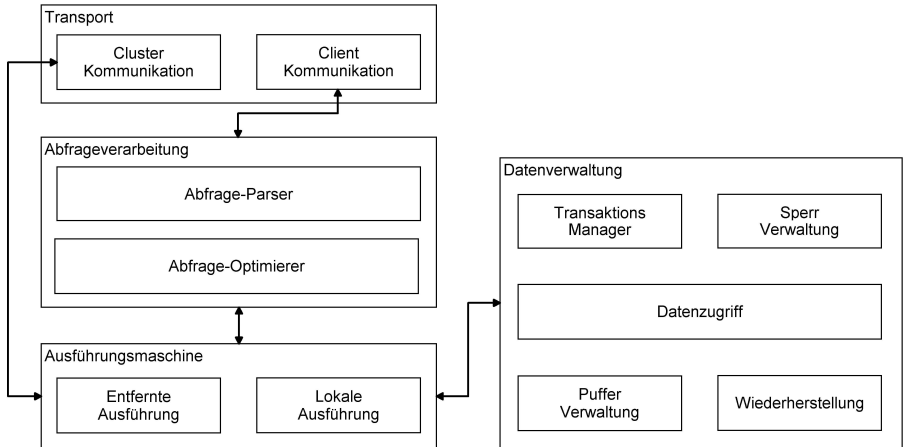
ProduktID	Bezeichnung	Kategorie	Preis
100000	Gartentisch Athen	Tisch	120,00
100001	Blumentopf Kreta	Topf	4,50
100002	Pflanztisch Kos	Tisch	75,00

- ▶ Spaltenorientierte Speicherung

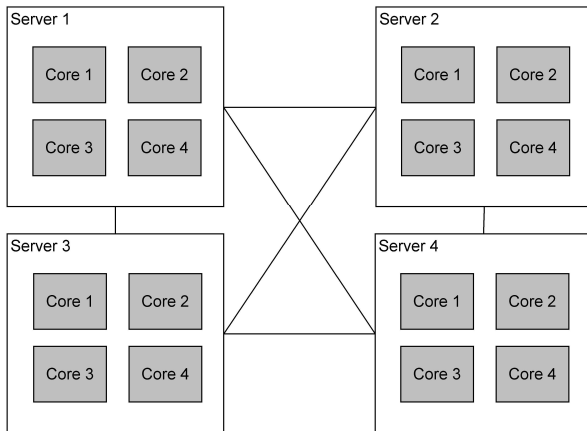
ProduktID	Bezeichnung	Kategorie	Preis
100000	Gartentisch Athen	Tisch	120,00
100001	Blumentopf Kreta	Topf	4,50
100002	Pflanztisch Kos	Tisch	75,00

- ▶ In Memory / in Processor Cache
  - ▶ Komplette DB im Hauptspeicher
  - ▶ Cache-Optimierung (z.B. Join)
    - L2-Cache-Referenz: 10ns
    - RAM-Referenz: 100ns

# Komponenten eines Datenbanksystems



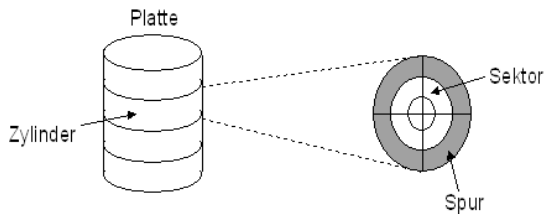
# Cluster



# Speicherhierarchie

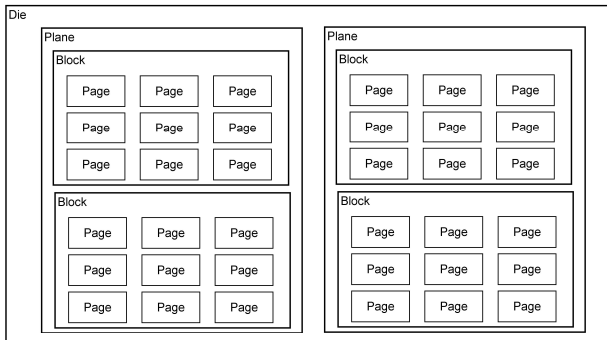
- ▶ Prozessocache (Level 1, 2, 3)
  - ▶ sehr schnell
  - ▶ sehr klein
  - ▶ flüchtig
- ▶ Hauptspeicher (RAM)
  - ▶ schnell
  - ▶ klein
  - ▶ flüchtig
- ▶ Sekundärspeicher (Festplatte, SSD)
  - ▶ langsam
  - ▶ groß
  - ▶ nicht flüchtig
- ▶ Archivspeicher (Bänder, CD)
  - ▶ sehr langsam
  - ▶ sehr groß
  - ▶ nicht flüchtig

# Magnetische Festplatte



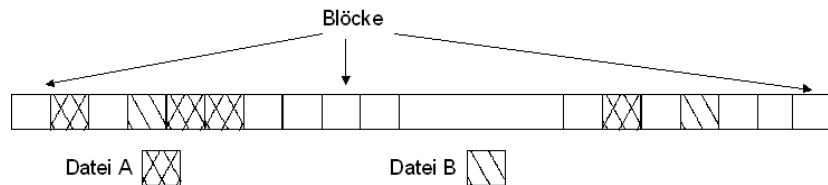
- ▶ Lese-/Schreibkopf: Positionierung zeitaufwändig
- ▶ Sequentielles Lesen von Sektoren viel schneller als wahlfreies Lesen
- ▶ Sektoren werden auch als Böcke bezeichnet

# Solid State Disk (SSD)



- ▶ Erst Löschen, dann Schreiben
- ▶ Löschung nur auf Blockebene

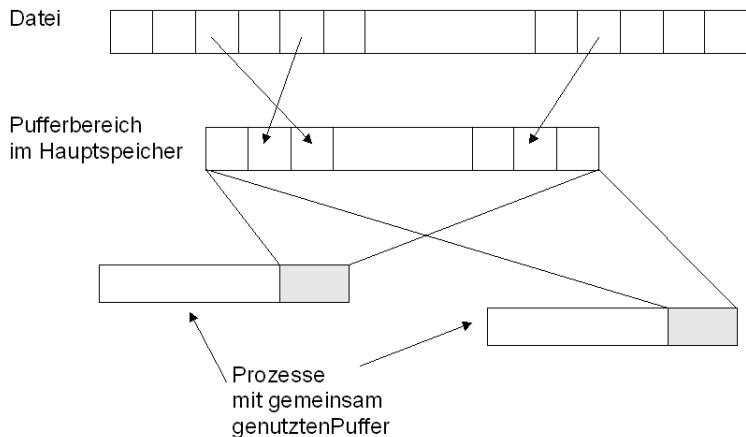
## Dateien



Datei = Menge von Blöcken/Seiten



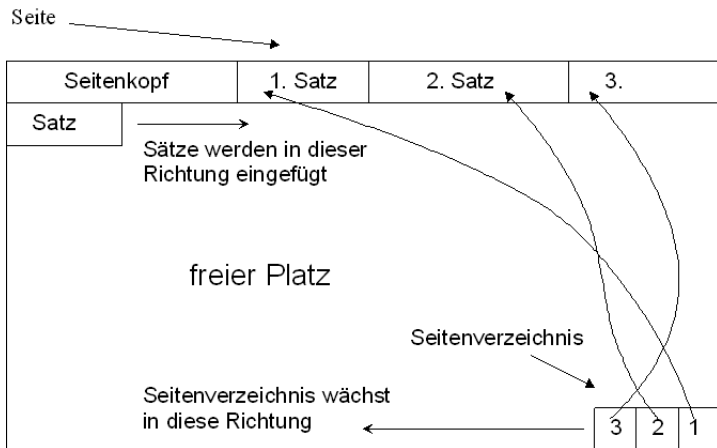
## Seitenpuffer



## Pufferverwaltung: Bereitstellen einer Seite

```
/* Seite s = (d,b), d. h. Block b aus Datei d */  
  
if s ist im Puffer then  
    return Pufferspeicheradresse von s  
else  
    suche freien Pufferplatz  
    if kein freier Pufferplatz vorhanden then  
        suche geeignete Seite zu entfernen aus dem Puffer  
        schreibe Seite auf Platte, falls notwendig  
        entferne Seite aus dem Puffer  
    endif  
    /* an dieser Stelle ist p der freie Pufferplatz */  
    lies Block b aus Datei d in Pufferplatz p  
    return Pufferspeicheradresse von p  
endif
```

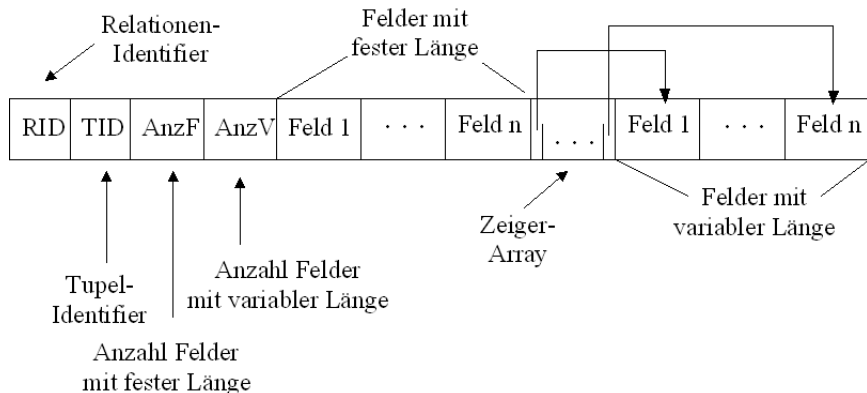
## Verwaltung von Sätzen in Seiten



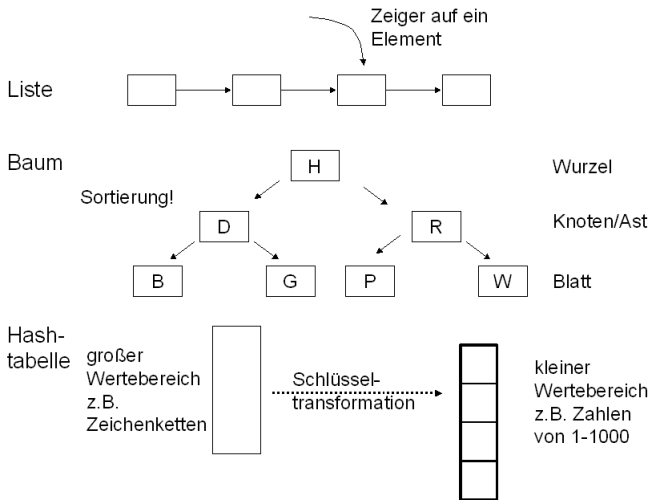
Tuple-Identifizier (TID)

TID = (Seitennummer, Index im Seitenverzeichnis)

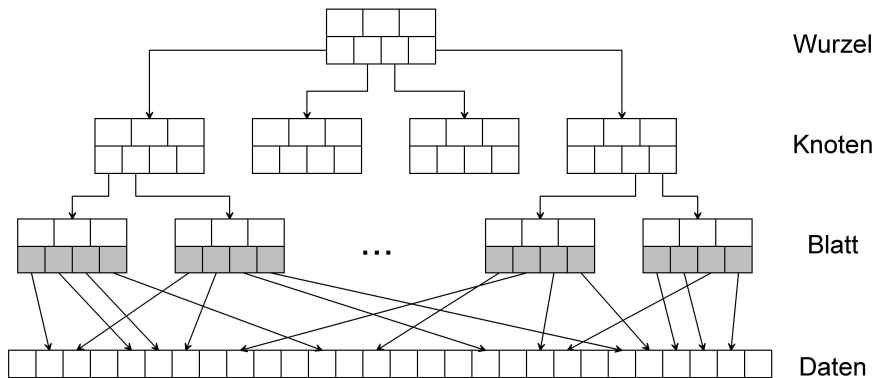
# Speicherstruktur für Sätze



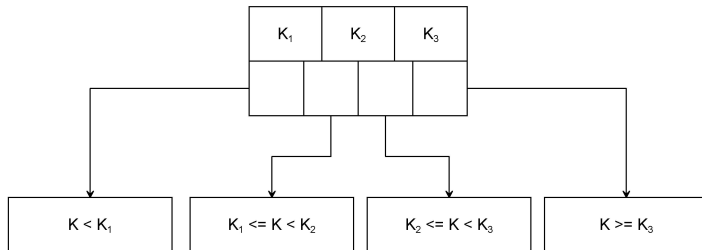
## Exkurs: Datenstrukturen



## B-Baum – Überblick



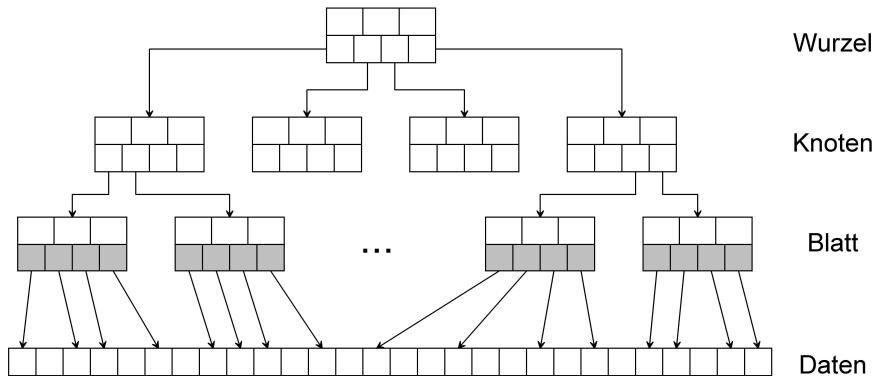
## B-Baum – Seitendetails



Annahme: 50 Schlüssel pro Seite

- ▶ 2 Ebenen  $\approx$  2500 Sätze
- ▶ 3 Ebenen  $\approx$  125.000 Sätze
- ▶ 4 Ebenen  $\approx$  6.250.000 Sätze

## B-Baum – clustered



Daten sind nach Schlüsselwerten sortiert

- ▶ Nur einer pro Tabelle
- ▶ Schnell für Bereichszugriffe



## Systemtabellen (Systemkatalog, Data Dictionary)

- ▶ Tabellen, die die Objekte innerhalb der Datenbank beschreiben
  - ▶ Tabellen
  - ▶ Spalten
  - ▶ Schlüssel
  - ▶ Indizes
  - ▶ Gespeicherte Prozeduren
  - ▶ Trigger
  - ▶ ...
- ▶ Werden für den laufenden Betrieb benötigt
  - ▶ Kontextanalyse bei Anfragen: Existieren die in der SQL-Anweisung angegebenen Tabellen und Spalten
  - ▶ Anfrageoptimierung: Welche Indizes existieren
  - ▶ Datenzugriff: Wo beginnt eine Spalte im Datensatz
  - ▶ Berechtigungsüberprüfung

## Beispiel: Oracle-Server

### ► Tabellen

```
select T.table_name
from user_tables T
```

TableName
STUDENT VERANSTALTUNG STUDIENGANG BEWERTUNG

### ► Spalten

```
select TC.column_name, TC.column_id
from user_tab_columns TC
where tc.table_name = 'STUDENT'
```

ColumnName	ColumnId
MATRNR	1
VORNAME	2
NAME	3
STUDIENGANG	4