# Einführung in die Informatik: Systeme und Anwendungen

## Teil: Datenbanksysteme


Prof. Dr. Stefan Conrad

Ludwig-Maximilians–Universität München
Institut für Informatik
Lehr- und Forschungseinheit für Datenbanksysteme
Oettingenstraße 67, D-80538 München

conrad@dbs.informatik.uni-muenchen.de


Universität München
Sommersemester 2001

— *Folienskriptum* —

Stefan Conrad


**Danksagung**

Diese Folien stellen einen überarbeiteten Auszug aus einem Foliensatz zu dem folgenden Buch dar:

Heuer, A.; Saake, G.; Sattler, K.:
*Datenbanken kompakt.*
MITP, Bonn, April 2001.

Den Buchautoren sei an dieser Stelle herzlichst gedankt für die Bereitstellung Ihres Folienmaterials!

# 1. Datenbanksysteme — Einleitung

- Motivation
  - Datenredundanz $\leftrightarrow$ Datenintegration
  - Historie
- Prinzipien und Architektur
  - 3-Ebenen-Architektur
  - Codd'sche Regeln (Anforderungen)
  - Systemarchitektur
  - Einsatzgebiete und Grenzen
- das Relationenmodell
  - Konzept der Tabelle / Relation
  - Begriffe des Relationenmodells
  - Relationenalgebra





## 1.1. Motivation

- Basis- oder Anwendungssoftware verwaltet ihre eigenen Daten in ihren eigenen (Datei-)Formaten; *Bsp.:*
  - Textverarbeitung: Texte, Artikel und Adressen
  - Buchhaltung: Artikel, Adressen
  - Lagerverwaltung: Artikel, Aufträge
  - Auftragsverwaltung:
    Aufträge, Artikel, Kundenadressen
  - CAD-System:
    Artikel, Technische Daten, Technische Bausteine
  - Produktion:
    . . . , Bestelleingang:. . . , Kalkulation:. . .
- Daten sind redundant: mehrfach gespeichert; Probleme:
  - Verschwendung von Speicherplatz
  - „Vergessen“ von Änderungen;
  - keine zentrale, „genormte“ Datenhaltung

# Ohne Datenbanken: Datenredundanz

- Andere Software-Systeme (auch Programmiersprachen, Tabellenkalkulation, Dateiverwaltungssysteme, . . . ) können große Mengen von Daten nicht effizient verarbeiten.
- Mehrere Benutzer oder Anwendungen können nicht parallel auf den gleichen Daten arbeiten, ohne sich zu stören.
- Anwendungsprogrammierer bzw. Benutzer können Anwendungen nicht programmieren oder benutzen, ohne
  - interne Darstellung der Daten
  - Speichermedien oder Rechner (bei verteilten Systemen)

  zu kennen.

  (**Datenunabhängigkeit** nicht gewährleistet)
- Datenschutz und Datensicherheit nicht gewährleistet



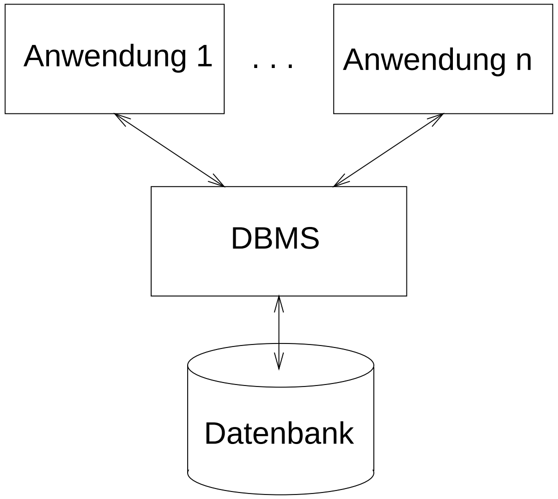


# Mit Datenbanken: Datenintegration

- Die gesamte Basis- und Anwendungssoftware arbeitet auf denselben Daten
  ($\leadsto$ Datenbankentwurf, Datendefinition)
- *Bsp.:*
  Adressen und Artikel werden nur einmal gespeichert
- Auch andere Probleme (Effizienz, Parallelität, Datenschutz, Datensicherheit) werden gelöst
  - Datenbanksysteme können große Datenmengen effizient verwalten (Anfragesprachen, Optimierung, Interne Ebene)
  - Benutzer können parallel auf Datenbanken arbeiten (Transaktionskonzept)
  - Datenunabhängigkeit durch 3-Ebenen-Konzept
  - Datenschutz (kein unbefugter Zugriff) und Datensicherheit (kein ungewollter Datenverlust) werden vom System gewährleistet

# Historie

- Anfang 60er Jahre: elementare Dateien, anwendungsspezifische Datenorganisation
  $\rightarrow$ geräteabhängig, redundant, inkonsistent
- Ende 60er Jahre: Dateiverwaltungssysteme (SAM, ISAM) mit Dienstprogrammen (Sortieren)
  $\rightarrow$ geräteunabhängig, aber redundant & inkonsistent
- 70er Jahre: Datenbanksysteme
  $\rightarrow$ Geräte- und Datenunabhängigkeit, redundanzfrei, konsistent



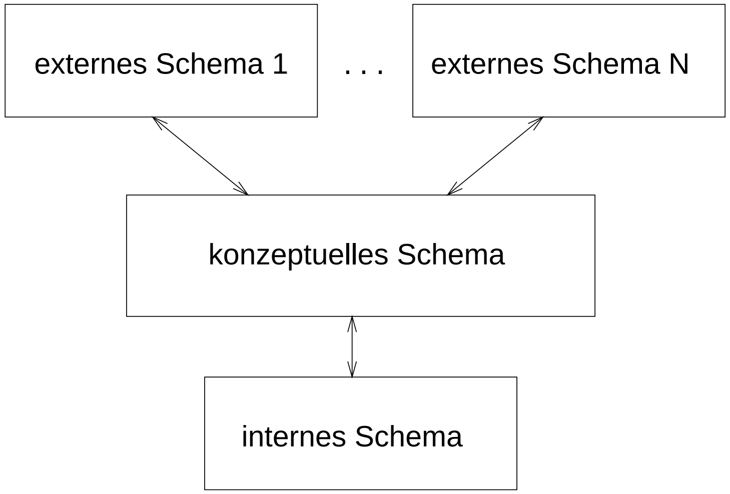

- DBMS: Datenbank-Management-System
- DBS: Datenbanksystem (DBMS + Datenbank)





# 1.2. Prinzipien und Architektur

- Grundprinzip moderner Datenbanksysteme
  - 3-Ebenen-Architektur
    $\rightsquigarrow$ physische Datenunabhängigkeit und logische Datenunabhängigkeit




  - Trennung zwischen
    – Schema (etwa Tabellenstruktur) und
    – Instanz (etwa Tabelleninhalt)

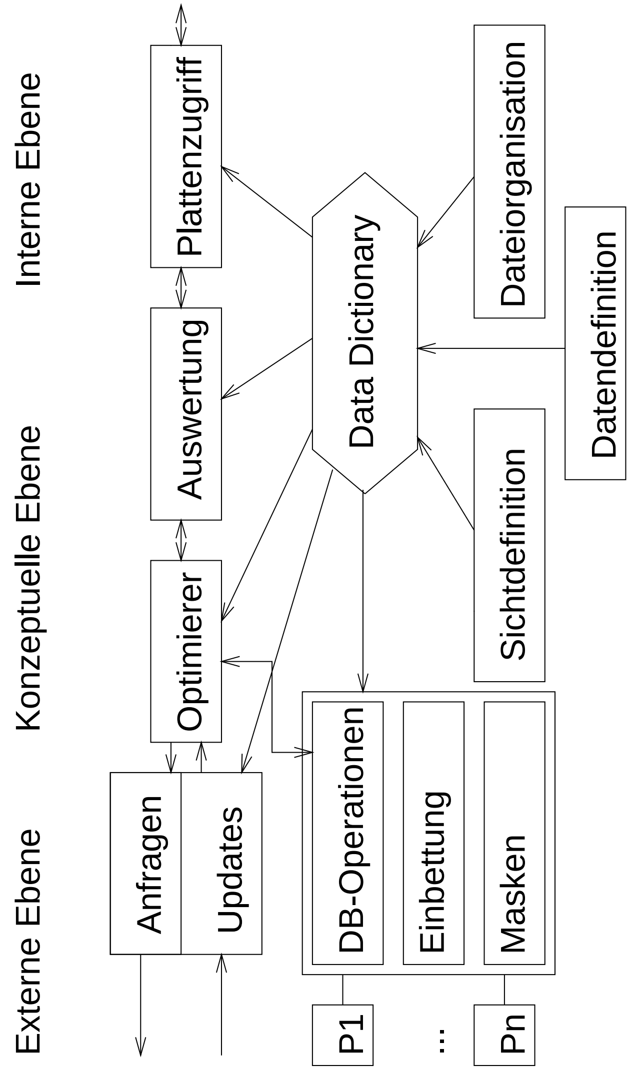
# Die neun Codd'schen Regeln

1. Integration
   - einheitliche, nichtredundante Datenverwaltung
2. Operationen
   - Speichern, Suchen, Ändern
3. Katalog
   - Zugriff auf DB-Beschreibungen im Data Dictionary
4. Benutzersichten
5. Konsistenzüberwachung
   - auch: Integritätssicherung (Korrektheit des DB-Inhalts)
6. Datenschutz
   - Ausschluß unauthorisierter Zugriffe
7. Transaktionen
   - Mehrere DB-Operationen als Funktionseinheit
8. Synchronisation
   - Parallele Transaktionen koordinieren
9. Datensicherung
   - Wiederherstellung von Daten nach Systemfehlern





# Architektur eines Datenbanksystems I



# Architektur eines Datenbanksystems II

- Dateiorganisation: Definition der (internen) Dateiorganisation und Zugriffspfade
- Datendefinition: Konzeptuelle Datendefinition (konzeptuelles Schema)
- Sichtdefinition: Definition von Benutzersichten (externe Schemata)
- Optimierer und Auswertung: Effiziente Umsetzung dieser Operationen
- Plattenzugriff: Plattenzugriffssteuerung (genauer: Fünf-Schichten-Architektur)
- Masken: Entwurf von Menüs und Masken
- DB-Operationen: Anfrage- und Änderungs-Operationen
- Einbettung: Einbettung dieser Operationen in Anwendungsprogramme
- P1, . . . , Pn: Verschiedene Anwendungsprogramme





# Einsatzgebiete und Grenzen

Klassische Einsatzgebiete:

- viele Objekte (15000 Bücher, 300 Benutzer, 100 Ausleihvorgänge pro Woche, . . . )
- wenige Objekttypen (BUCH, BENUTZER, AUSLEIHUNG)
- etwa Buchhaltungssysteme, Auftragserfassungssysteme, Bibliothekssysteme, . . .

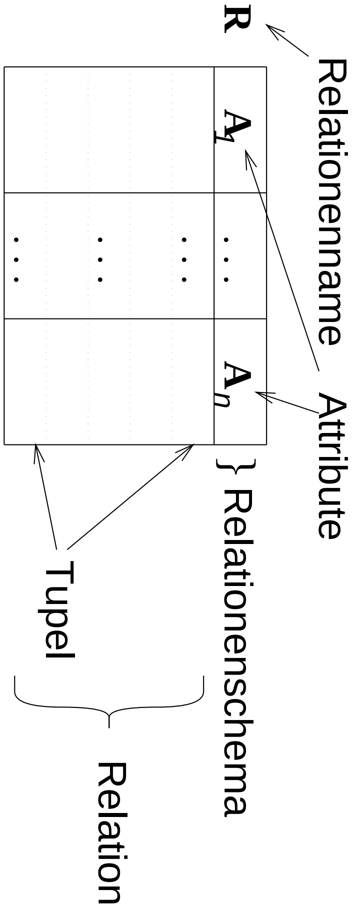
Normalerweise sind herkömmliche Datenbanksysteme überfordert mit:

- CAD- oder andere technische Anwendungen
  (viele Objekte, viele Objekttypen, sehr strukturierte Objekte)
  - *aber:* Objektorientierte Datenbanksysteme
- Expertensysteme (wenige Objekte, viele Objekttypen, kompliziertere Operationen)
  - *aber:* Deduktive Datenbanksysteme

# 1.3. Relationenmodell

[Codd, 1970]

**Veranschaulichung eines Relationenschemas und einer Relation**








## Zwei Relationen zur Darstellung von `Personen`

Personen

| PANr | Vorname | Nachname | PLZ | Ort | Straße | HNr | Geb.datum |
|---|---|---|---|---|---|---|---|
| 4711 | Andreas | Heuer | 18209 | DBR | BHS | 15 | 31.10.1958 |
| 5588 | Gunter | Saake | 39106 | MD | STS | 55 | 05.10.1960 |
| 6834 | Michael | Korn | 39104 | MD | BS | 41 | 24.09.1974 |
| 7754 | Andreas | Möller | 18209 | DBR | RS | 31 | 25.02.1976 |
| 8832 | Tamara | Jagellovsk | 38106 | BS | GS | 12 | 11.11.1973 |
| 9912 | Antje | Hellhof | 18059 | HRO | AES | 21 | 04.04.1970 |
| 9999 | Christa | Loeser | 69121 | HD | TS | 38 | 10.05.1969 |

Pers_Telefon

| PANr | Telefon |
|---|---|
| 4711 | 038203-12230 |
| 4711 | 0381-498-3401 |
| 4711 | 0381-498-3427 |
| 5588 | 0391-345677 |
| 5588 | 0391-5592-3800 |
| 9999 | 06221-400177 |

# Begriffe des Relationenmodells

| **Begriff** | **Informelle Bedeutung** |
|---|---|
| Attribut | Spalte einer Tabelle |
| Wertebereich | mögliche Werte eines Attributs (auch Domäne) |
| Attributwert | Element eines Wertebereichs |
| Relationenschema | Menge von Attributen |
| Relation | Menge von Zeilen einer Tabelle |
| Tupel | Zeile einer Tabelle |
| Datenbankschema | Menge von Relationenschemata |
| Datenbank | Menge von Relationen (Basisrelationen) |
| Schlüssel | minimale Menge von Attributen, deren Werte ein Tupel einer Tabelle eindeutig identifizieren |
| Primärschlüssel | ein beim Datenbankentwurf ausgezeichneter Schlüssel |
| Fremdschlüssel | Attributmenge, die in einer anderen Relation Schlüssel ist |
| Fremdschlüsselbedingung | alle Attributwerte des Fremdschlüssels tauchen in der anderen Relation als Werte des Schlüssels auf |





# Relationenalgebra (informell)

Operationen, die eine oder zwei Relationen als Eingabe nehmen und eine Ergebnisrelation prudzieren:

- Selektion

$$\sigma_{\texttt{Nachname}='\texttt{Meyer}'}(r(\texttt{Personen}))$$

$$\sigma_{\texttt{Nachname}=\texttt{Vorname}}(r(\texttt{Personen}))$$

- Projektion

$$\pi_{\texttt{Vorname},\texttt{PLZ}}(r(\texttt{Personen}))$$

- Verbund

$$\pi_{\texttt{Vorname},\texttt{Nachname},\texttt{Ort},\texttt{Straße},\texttt{Telefon}}( r(\texttt{Personen}) \bowtie r(\texttt{Pers\_Telefon}))$$

| Vorname | Nachname | Ort | Straße | Telefon |
|---|---|---|---|---|
| Andreas | Heuer | DBR | BHS | 038203-12230 |
| Andreas | Heuer | DBR | BHS | 0381-498-3401 |
| Andreas | Heuer | DBR | BHS | 0381-498-3427 |
| Gunter | Saake | MD | STS | 0391-345677 |
| Gunter | Saake | MD | STS | 0391-5592-3800 |
| Christa | Loeser | HD | TS | 06221-400177 |

- Mengenoperationen $\cap, \cup, -$
- Umbenennung $\beta_{\texttt{Wohnort}\leftarrow\texttt{Ort}}(r(\texttt{Personen}))$