

Datenbanksysteme



Thomas Neumann

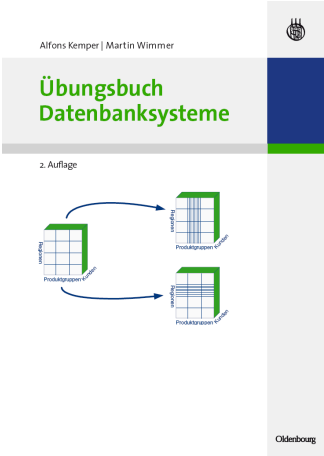
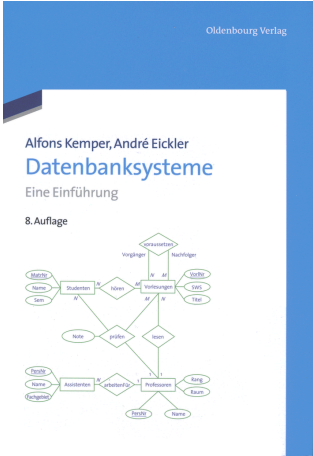
Einführung

Alfons Kemper und Andre Eickler
Datenbanksysteme – Eine Einführung
10. Auflage

Oldenbourg Verlag, München
(ca 50 Euro)

<http://www-db.in.tum.de/research/publications/books/DBMSeinf>

<http://www-db.in.tum.de>



- **A. Kemper, A. Eickler: Datenbanksysteme – Eine Einführung. Oldenbourg Verlag, 2015. 10. Auflage.**
- **A. Kemper, M. Wimmer: Übungsbuch Datenbanksysteme. Oldenbourg Verlag, 3. Auflage, 2011.**
- A. Silberschatz, H. F. Korth und S. Sudarshan: Database System Concepts, 4. Auflage, McGraw-Hill Book Co., 2002.
- R. Elmasri, S.B. Navathe: Fundamentals of Database Systems, Benjamin Cummings, Redwood City, Ca, USA, 2. Auflage, 1994
- R. Ramakrishnan, J. Gehrke: Database Management Systems, 3. Auflage, 2003.
- G. Vossen: Datenmodelle, Datenbanksprachen und Datenbank-Management-Systeme. Oldenbourg, 2001.

- D. Maier: The Theory of Relational Databases. Computer Science Press. 1983.
- S. M. Lang, P.C. Lockemann: Datenbankeinsatz. Springer Verlage, 1995.
- C. Batini, S. Ceri, S.B. Navathe: Conceptual Database Design, Benjamin Cummings, Redwood City, Ca, USA, 1992.
- C. J. Date: An Introduction to Database Systems. McGraw-Hill, 8. Aufl., 2003.
- J.D. Ullmann, J. Widom: A First Course in Database Systems, McGraw Hill, 2. Auflage, 2001.

- A. Kemper, G. Moerkotte: Object-Oriented Database Management: Applications in Engineering and Computer Science, Prentice Hall, 1994
- E. Rahm: Mehrrechner-Datenbanksysteme. Addison-Wesley, 1994.
- P. Dadam: Verteilte Datenbanken und Client/Server Systeme. Springer Verlag, 1996
- G. Weikum, G. Vossen: Transactional Information Systems: Theory, Algorithms, and the Practice of Concurrency Control. Morgan Kaufmann, 2001.
- T. Härder, E. Rahm: Datenbanksysteme – Konzepte und Techniken der Implementierung, 2001.

Einführung

- Was ist ein Datenbanksystem (DBS)?

Ein System zum Speichern und Verwalten von Daten.

- Warum kein herkömmliches Dateisystem verwenden?

Ausfallsicherheit und Skalierbarkeit nur mit hohem Aufwand erreichbar.

Traditionelle Anwendungsgebiete:

- Geschäftsdaten
- Buchhaltung
- Verwaltung
- ...

Heute sehr viel breiter:

- Wissenschaftliche/Medizinische Daten
- Data Mining
- Informationsintegration
- Websuche
- ...

Indirekt benutzen wir ständig Datenbanken:

- Websuche bei Google, Yahoo, ...
- Anfragen bei Amazon, EBay, ...
- Backend vieler großer Webseiten

Viele Spielarten (DB/IR, zentralisiert/dezentralisiert etc.)

Datenbanken werden fast immer eingesetzt wenn

- die Datenmengen groß sind
- die Daten wertvoll sind

Die großen kommerziellen Datenbanksysteme:

- Oracle
- IBM DB2
- Microsoft SQL Server
- Amazon Aurora
- Snowflake

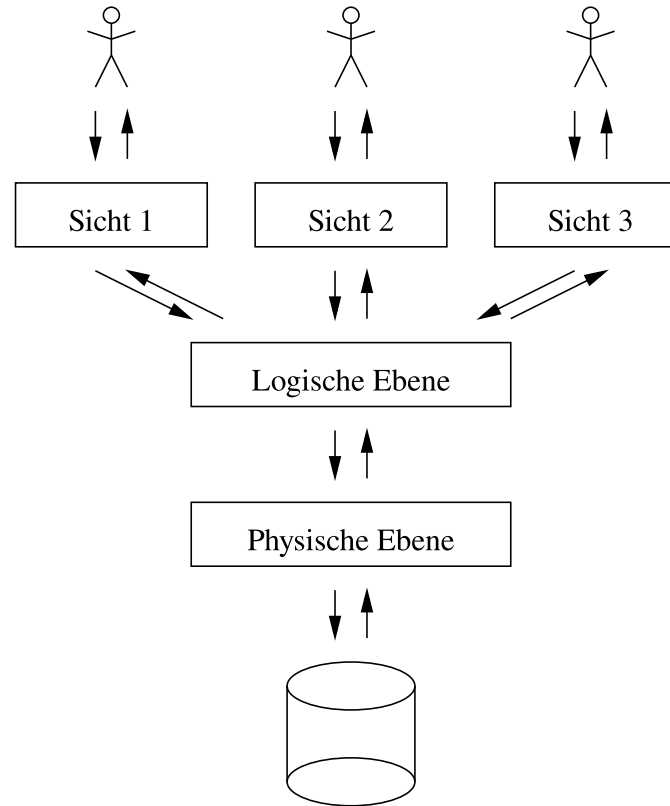
Einige freie Datebanksysteme:

- PostgreSQL
- MySQL
- DuckDB

Noch viele weitere, teils stark spezialisierte Systeme.

- Redundanz und Inkonsistenz
- Verschiedene Datenformate
- Probleme beim Mehrbenutzerbetrieb
- Verlust von Daten
- Integritätsverletzungen
- Sicherheitsprobleme
- Hohe Entwicklungskosten für Anwendungsprogramme

- Datenunabhängigkeit
- Deklarative Anfragesprachen
- Mehrbenutzersynchronisation
- Fehlerbehandlung
- Sicherstellung der Datenintegrität
- Effizienz und Skalierbarkeit



- Sicht: beschreibt wie ein Benutzer die Daten sieht
- Logische Ebene: beschreibt wie die Daten strukturiert sind
- Physische Ebene: beschreibt wie die Daten gespeichert werden

- DBS entkoppelt Anwendungen von der eigentlichen Struktur und Speicherung der Daten
- Logische Datenunabhängigkeit
 - Änderungen auf der logischen Ebene haben keinen Einfluß auf Anwendungen
- Physische Datenunabhängigkeit
 - Änderungen auf der physischen Ebene haben keinen Einfluß auf Anwendungen
 - Wird in fast allen modernen DBS durchgesetzt

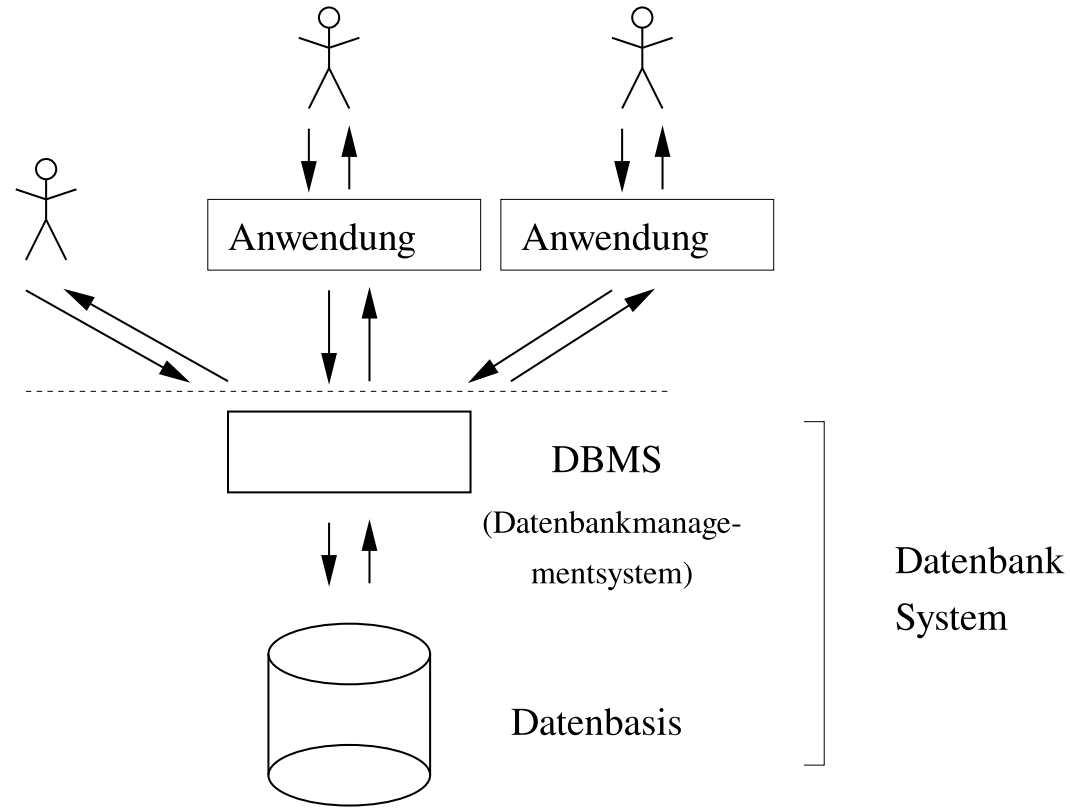
- Benutzer sagt DBS *was* für Daten geholt werden sollen ...
- ...und nicht *wie* die Daten geholt werden sollen
- Weniger fehleranfällig (beim Formulieren von Anfragen/Entwickeln von Anwendungen), da kein Wissen über die tieferen Schichten des DBS nötig sind

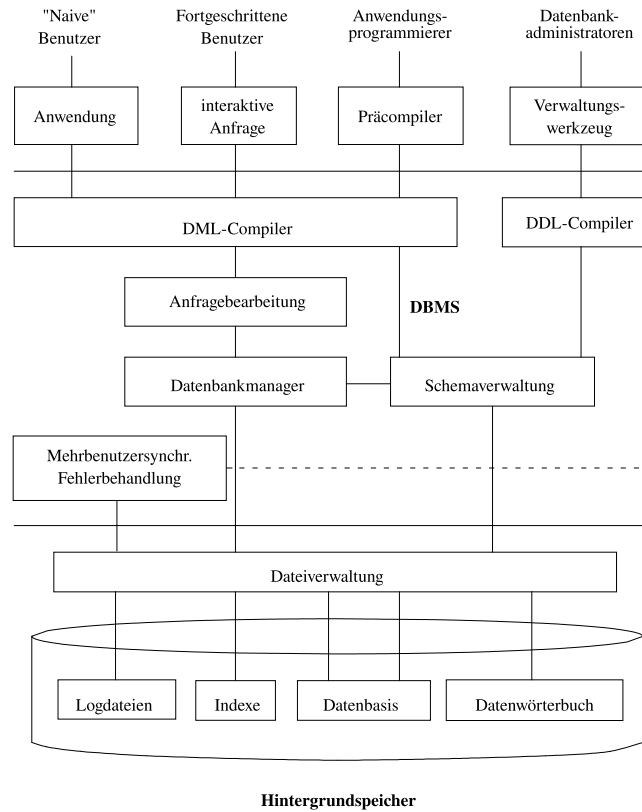
- Wenn mehrere Benutzer ohne jegliche Kontrolle gleichzeitig Daten ändern können, gibt es große Probleme
- DBS erlaubt gleichzeitigen Zugriff und verhindert schlimme Seiteneffekte

- DBS kann Zustand zum Zeitpunkt eines Absturzes rekonstruieren
- Dafür werden Logdateien vom DBS angelegt und verwaltet

- Datenverarbeitung in einer Anwendung läuft nicht völlig zufällig ab, sondern folgt gewissen Prinzipien
- DBS befolgt (angegebene) Prinzipien und schützt so vor:
 - Benutzerfehlern
 - Programmfehlern

- DBSe sind für groß angelegte Anwendungen konzipiert
- In DBSen sind Techniken integriert, die mit großen Datenvolumen umgehen können



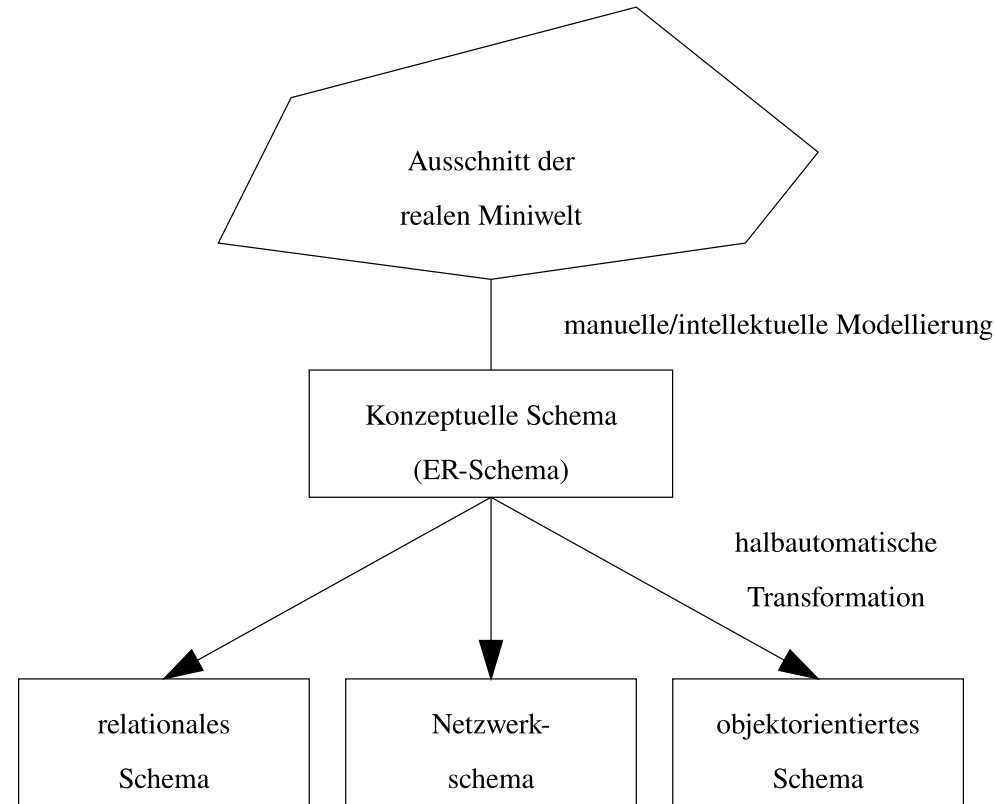


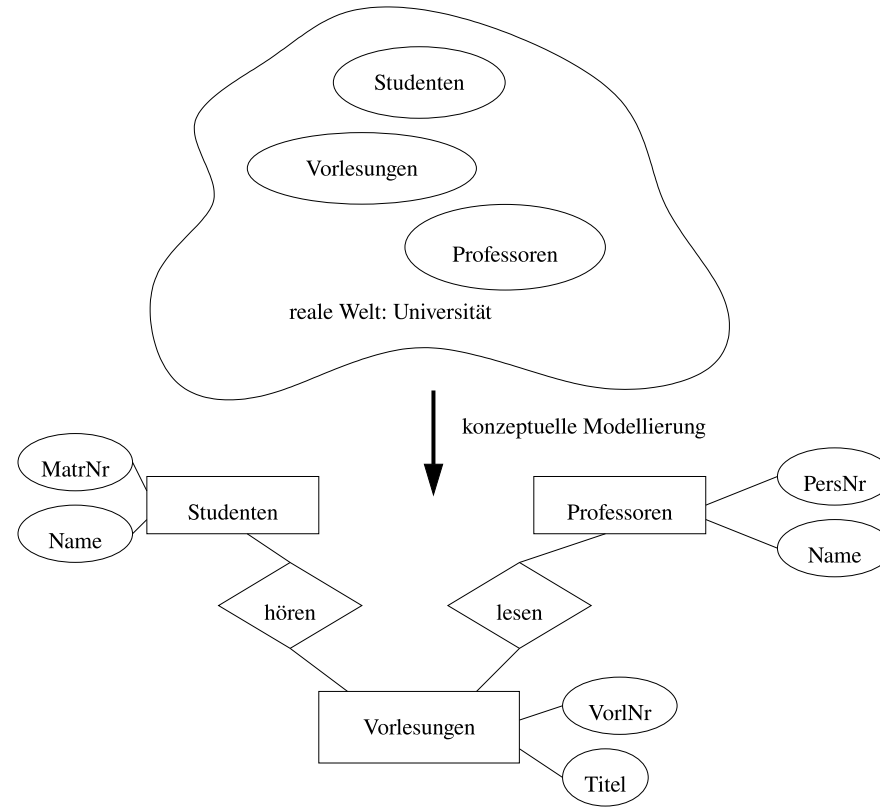
- Auch wenn DBS vieles kann, es kann nicht alles!
- Ein Benutzer muß immer noch die

Anforderungen einer Anwendung ...

- ...und die Art von Daten die gespeichert werden sollen spezifizieren
- Zwei wichtige Konzepte beim Entwurf:
 - Datenmodell: legt fest, welche Konstrukte zum Beschreiben der Daten existieren
 - Schema: eine konkrete Beschreibung einer bestimmten Datensammlung (unter Verwendung eines Datenmodells)

- Konzeptuelle Modelle:
 - Entity-Relationship-Modell (ER-Modell)
 - Unified Modeling Language (UML)
- Logische Modelle:
 - Hierarchisches Modell
 - Netzwerkmodell
 - Relationales Modell
 - Objekt-orientiertes Modell
 - Objekt-relationales Modell
 - Semistrukturiertes Modell
 - Graphstrukturiertes Modell





Relationales Schema

Studenten	
MatrNr	Name
26120	Fichte
25403	Jonas
...	...

hören	
MatrNr	VorlNr
25403	5022
26120	5001
...	...

Vorlesungen	
VorlNr	Titel
5001	Grundzüge
5022	Glaube und Wissen
...	...

- Viele Anwendungen haben ähnliche Erfordernisse was die Datenverwaltung angeht
- Ein DBS bildet eine gemeinsame Basis und bietet eine Infrastruktur die diese Anwendungen unterstützt
- Für eine Übersicht über das was geboten wird siehe Folie “Gründe für den DBS-Einsatz”

- Wir versuchen zwei Seiten abzudecken:
 - Wie benutzt man ein DBMS?
 - Was passiert hinter den Kulissen?

- Inhalt:
 - Datenbankentwurf (Kapitel 2)
 - Relationales Modell (Kapitel 3)
 - SQL (Kapitel 4)
 - Relationale Entwurfstheorie (Kapitel 5)
 - Physische Datenorganisation (Kapitel 6)
 - Anfragebearbeitung (Kapitel 7)
 - Transaktionsverwaltung (Kapitel 8)
 - Fehlerbehandlung (Kapitel 9)
 - Mehrbenutzersynchronisation (Kapitel 10)
 - Verteilte Datenbanksysteme (Kapitel 11)