

Datenbanksysteme - Einleitung

Albert Weichselbraun <albert.weichselbraun@htwchur.ch>

Agenda

- Lernziele
- Motivation & einführendes Beispiel
- Modellierung

Weiterführende Literatur

- Meier, Andreas (2010): Relationale und postrelationale Datenbanken. Berlin / Heidelberg / New York: Springer (= eXamen.press).
- Übungsbeispiele: <https://weichselbraun.net/dbs>
- Zusammenfassung von Prof Panny

Lernziele des Moduls Datenbanksysteme

I. Funktionsweise

- Verständnis der Konzepte von Datenbanken
- Erkennen der Stärken und Schwächen des relationalen Modells

II. Datenbankentwurf

- Fähigkeit, Datenbankapplikationen methodisch korrekt zu entwerfen
- Modellierung von Datenbanken mittels
 - des Entity Relationship Modells und
 - Übertragung dieses Modells in das Relationale Modell
- Bewertung und Verbesserung des Datenbankdesigns (Normalformen)

III. Arbeiten mit Datenbanken

- Kennenlernen von relationalen Abfrage- und Manipulationssprachen
- Design und Abfrage von Datenbanken

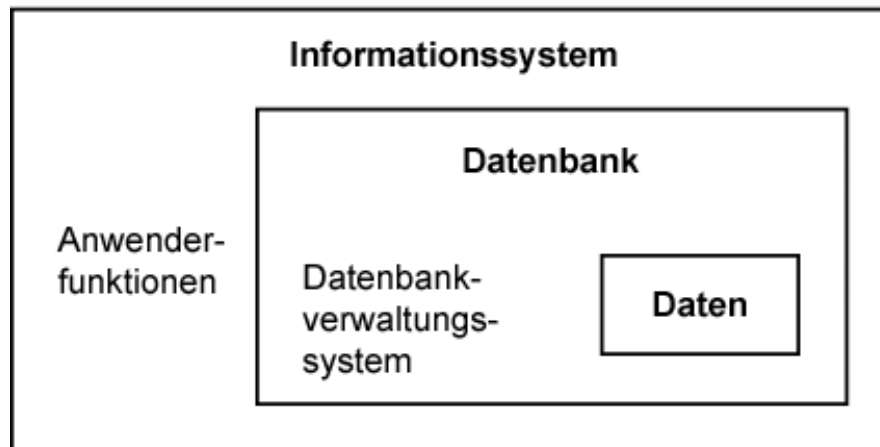
Informationssysteme

Ein System, welches informationsverarbeitende Prozesse realisiert, heisst **Informationssystem**. Es führt folgende Tätigkeiten aus:

- Informationen aufnehmen, verändern, löschen
- Informationen zwischenspeichern
- Informationen langfristig aufbewahren und zur Auskunftserteilung verfügbar machen
- mit anderen Informationssystemen kommunizieren
- Beispiele
 - Kontobuchungssysteme
 - Flugreservationssysteme
 - Bibliotheksausleihsysteme
 - Customer Care Systeme

Informationssysteme

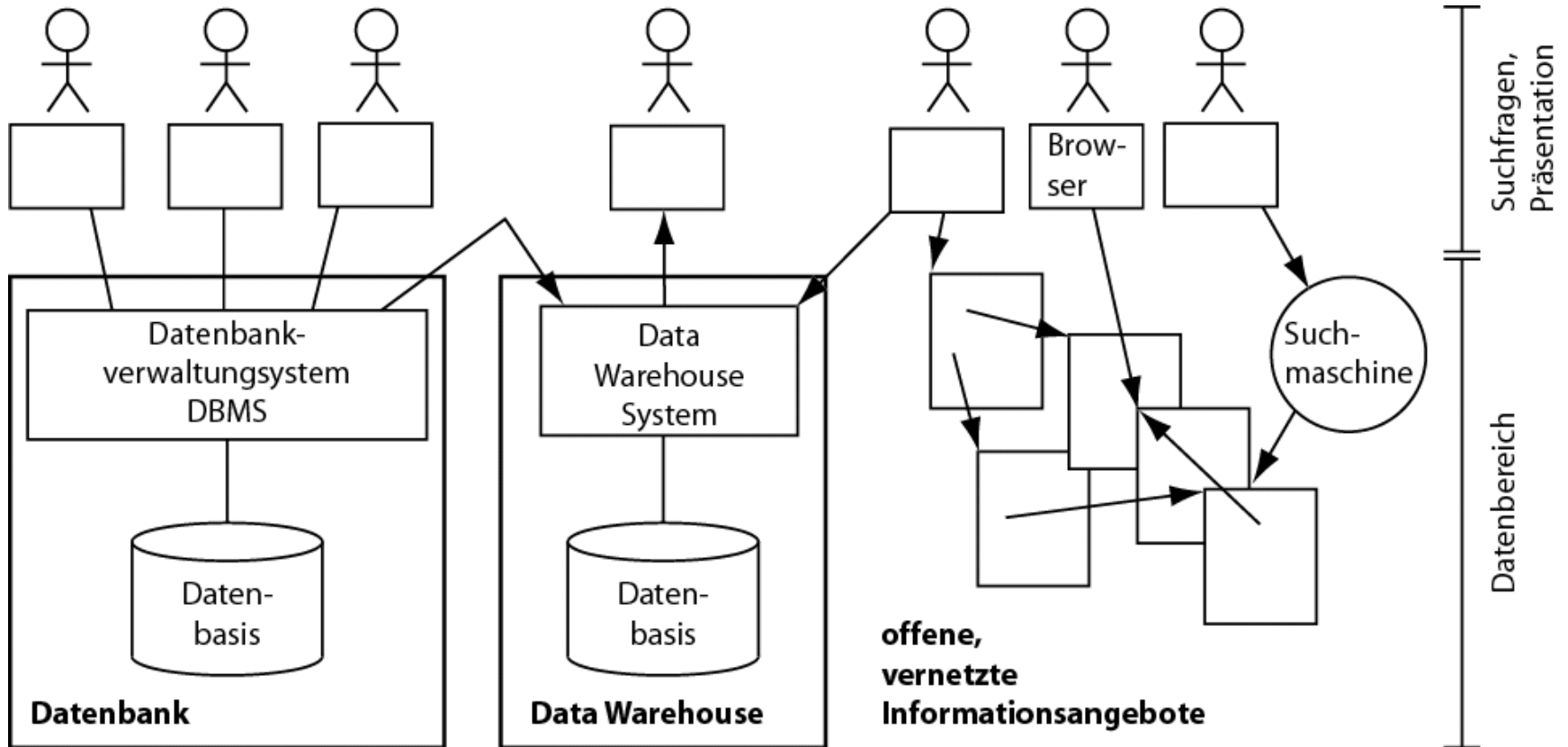
Schalenmodell des Informationssystems
Daten und Anwendungsfunktionen



Quelle: Bartelme (2000:16)

Informationssysteme

Beispiele für Informationssysteme

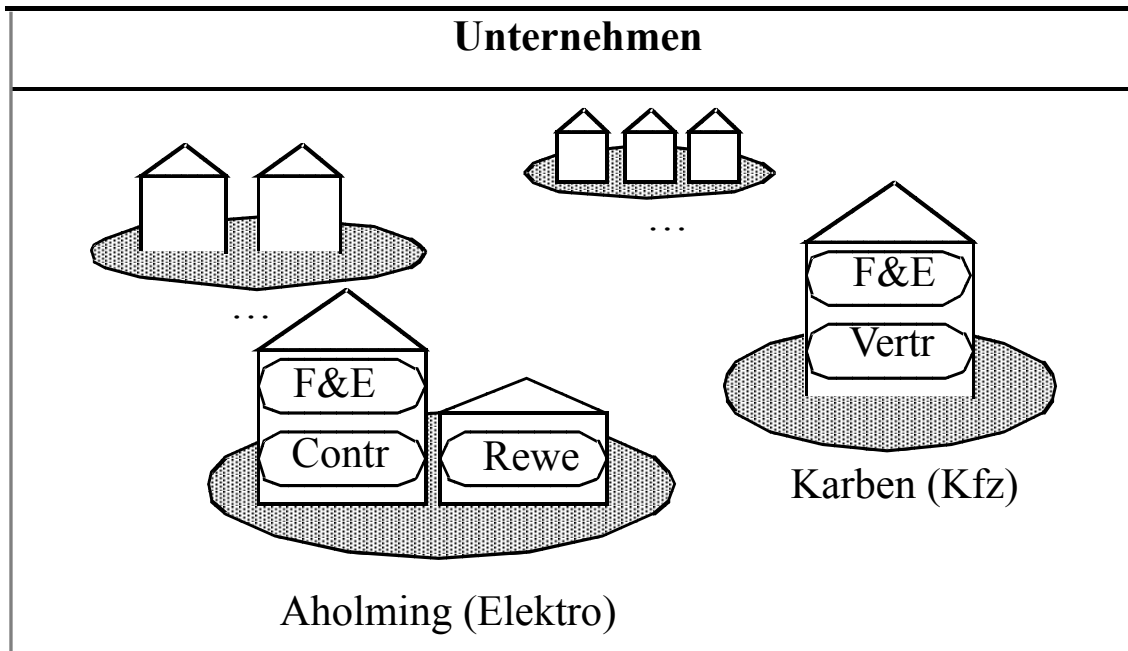


Einführendes Beispiel

Ein Unternehmen möchte ein Informationssystem aufbauen.

- Das Unternehmen gliedert sich in mehrere Unternehmensbereiche (Elektro, Kfz, ...).
- Das Unternehmen gliedert sich in mehrere Betriebe an verschiedenen Standorten (Aholming, Karben, ...),
- an jedem Standort jedoch nur ein Betrieb.
- Jeder Betrieb ist eindeutig einem Unternehmensbereich zugeordnet.
- Zu jedem Betrieb gehören ein oder mehrere Gebäude.
- Abteilungen (F&E, Controlling, ...) sind immer in einem Gebäude untergebracht.

Einführendes Beispiel



Einführendes Beispiel

Informationsbedarfsanalyse:

I. Mitarbeiterinnen und Mitarbeiter:

- Name,
- Personalnummer,
- an welchem Ort und
- in welchem Unternehmensbereich sie/er arbeitet,
- Abteilung,
- Gebäude, in welchem sich der Arbeitsplatz befindet
- Gehalt.

II. Standort eines Betriebs:

- die geographischen Koordinaten (für Tourenplanung),
- Straße/Hausnummer, Ortsname und PLZ,
- den Namen des Betriebsleiters,
- die Höhe des Personalbudgets.

Der konventionelle Ansatz

Um die Aufgabe zu lösen, wird von der zentralen EDV-Abteilung ein COBOL-Programm erstellt, das die geforderten Daten über Bildschirmmasken abfragt und in die Dateien PERSONAL bzw. STANDORT schreibt.

Datei: „PERSONAL“, feste Satzlänge=80 Bytes						
Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Länge: 16	8	16	18	10	4	8
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	F&E	2	43.500
Jöndhard	739	Karben	Kfz	F&E	2	45.300
Frits	17	Fürth	Mechanik	Contr	4	90.000

Der konventionelle Ansatz

Datei: „STANDORT“, feste Satzlänge=81 Bytes					
Standort	PLZ	Straße	Koord	Leiter	PersBudget
Länge: 12	5	20	15	20	9
Aholming	94527	Bärengasse 22	48.47N:12.59E	Beutel	560.000
München	81523	Codd-Weg 9	48.07N:11.38E	Schmitz	900.000
Karben	61184	Nusshof 17	50.32N:08.71E	Dieler	120.000
Fürth	90763	Maierring 109	49.23N:10.61E	Gabler	389.200

Im Laufe der Zeit wurden weitere Programme geschrieben, die „PERSONAL“ und/oder „STANDORT“ benutzen.

Der konventionelle Ansatz

Die Aufteilung eines Satzes in Felder sowie die Länge der einzelnen Datenfelder sind in jedem COBOL-Programm, das die Datei benutzen soll, genau zu definieren (und damit auch die Satzlänge eines Datensatzes).

„File-Descriptor“ (FD)

...

FD STANDORT.

01 STANDORT-SATZ.

05 STANDORT-NAME PIC X(12).

05 PLZ PIC 9(05).

05 STRASSE PIC X(20).

05 KOORDINATEN PIC X(15).

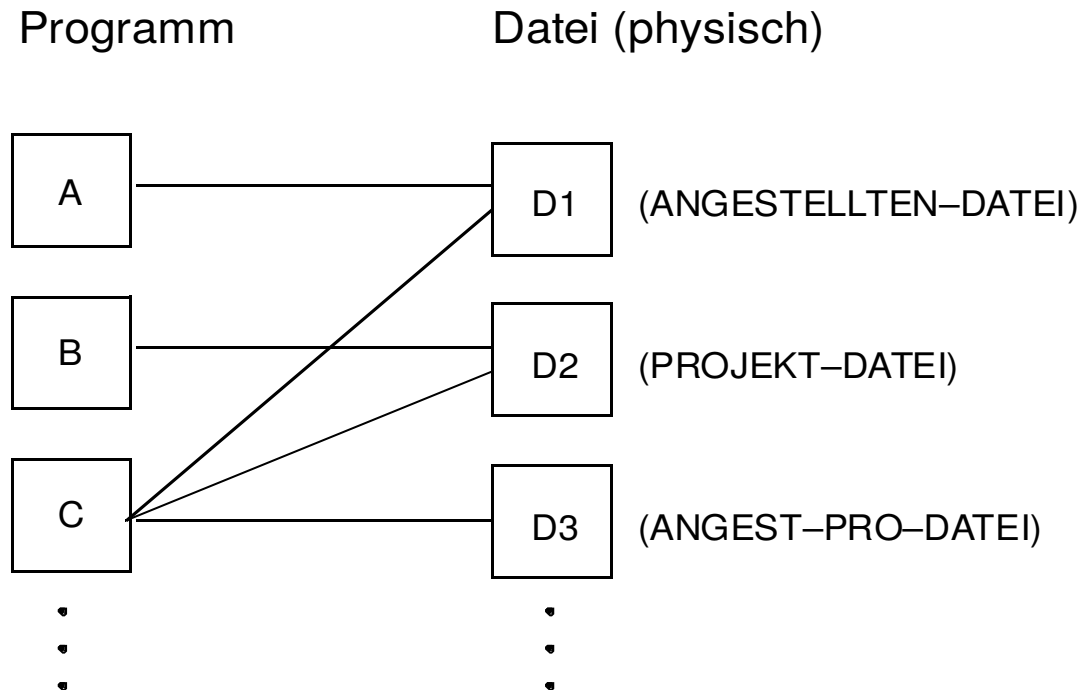
05 LEITER-NAME PIC X(20).

05 PERSBUDGET PIC 9(09).

...

Der konventionelle Ansatz

Programmierung im konventionellen Stil:
Enge Verflechtung zwischen Programm und Daten, d.h. Programm-Daten-
Abhängigkeit.



Der konventionelle Ansatz

Problem 1: Änderung

- Feldlänge
Beispiel: Seit 1.7.1993 gab es neue Postleitzahlen
- Entfernen oder Hinzufügen eines Feldes
Beispiel: In die PERSONAL-Datei soll für jeden Mitarbeiter das Geburtsdatum aufgenommen werden
- Veränderung der Anordnung der Felder

Folgen

- Änderung der Dateibeschreibung, damit Änderung und
- Neuinstallation aller Programme, die auf diese Datei zugreifen.

Aus versäumten Änderungen können Absturz des betreffenden Programms und (schlimmer) falsche Rechenergebnisse resultieren.

Der konventionelle Ansatz

Problem 2: Redundanz

Der Unternehmensvorstand „Forschung und Entwicklung“ möchte zum einfacheren Versenden von Rundschreiben eine Datei aller „F&E“-Mitarbeiter haben.

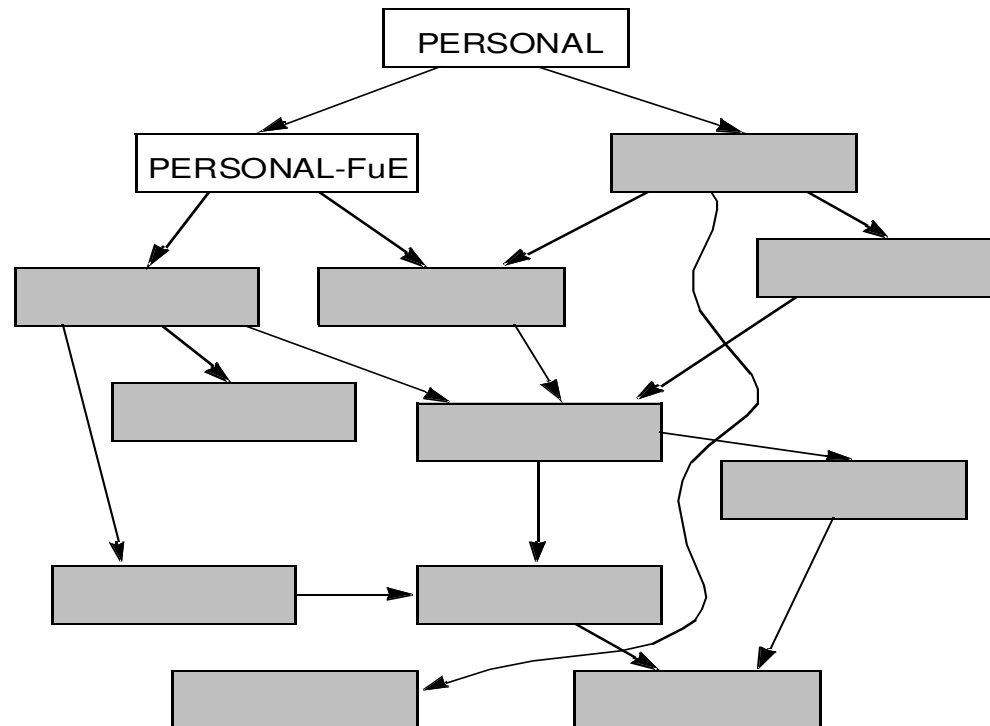
Er kopiert sich aus der Personaldatei PERSONAL (vgl. Seite 9) die Sätze aller „F&E“-Mitarbeiter und erstellt daraus die Datei PERSONAL–FuE. Diese enthält anstelle des Gehaltes ein zusätzliches Feld „Experte“, das angibt, worin sich der Mitarbeiter besonders gut auskennt:

Datei: „PERSONAL-FuE“, feste Satzlänge=86 Bytes						
Name	PersNr	StOrt	UBereich	Abt	GebNr	Expert
Länge: 16	8	16	18	10	4	14
Frits	17	Aholming	Elektro	F&E	11	VLSI
Enzian	17	München	Mechanik	F&E	2	Quanten
Truhel	54	Karben	Kfz	F&E	2	Sicherheit
Jöndhard	739	Karben	Kfz	F&E	2	Motor

Der konventionelle Ansatz

Andere Stellen im Unternehmen kopieren die PERSONAL-Datei ebenfalls und modifizieren ihre Kopien nach ihren Bedürfnissen.

Unter Umständen werden auch die modifizierten Kopien selbst wieder weitergegeben. So entsteht im Unternehmen mit der Zeit eine große Sammlung unterschiedlicher Abkömmlinge der PERSONAL-Datei:



Der konventionelle Ansatz

Herr Truhel aus F&E wird in den Vertrieb versetzt (bei gleichzeitiger Gehaltserhöhung)

Datei: „PERSONAL“, feste Satzlänge=80 Bytes						
Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Länge: 16	8	16	18	10	4	8
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	F&E	2	43.500
Jöndhard	739	Karben	Kfz	F&E	2	45.300

↓
Änderung durch die Personalabteilung

Datei: „PERSONAL“, feste Satzlänge=80 Bytes						
Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Länge: 16	8	16	18	10	4	8
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	Vertr.	8	44.000
Jöndhard	739	Karben	Kfz	F&E	2	45.300

Der konventionelle Ansatz

Problem 3: Inkonsistenzen

Alle von PERSONAL abgeleiteten Dateien enthalten nun falsche Angaben über Herrn Truhel → Inkonsistenzen.

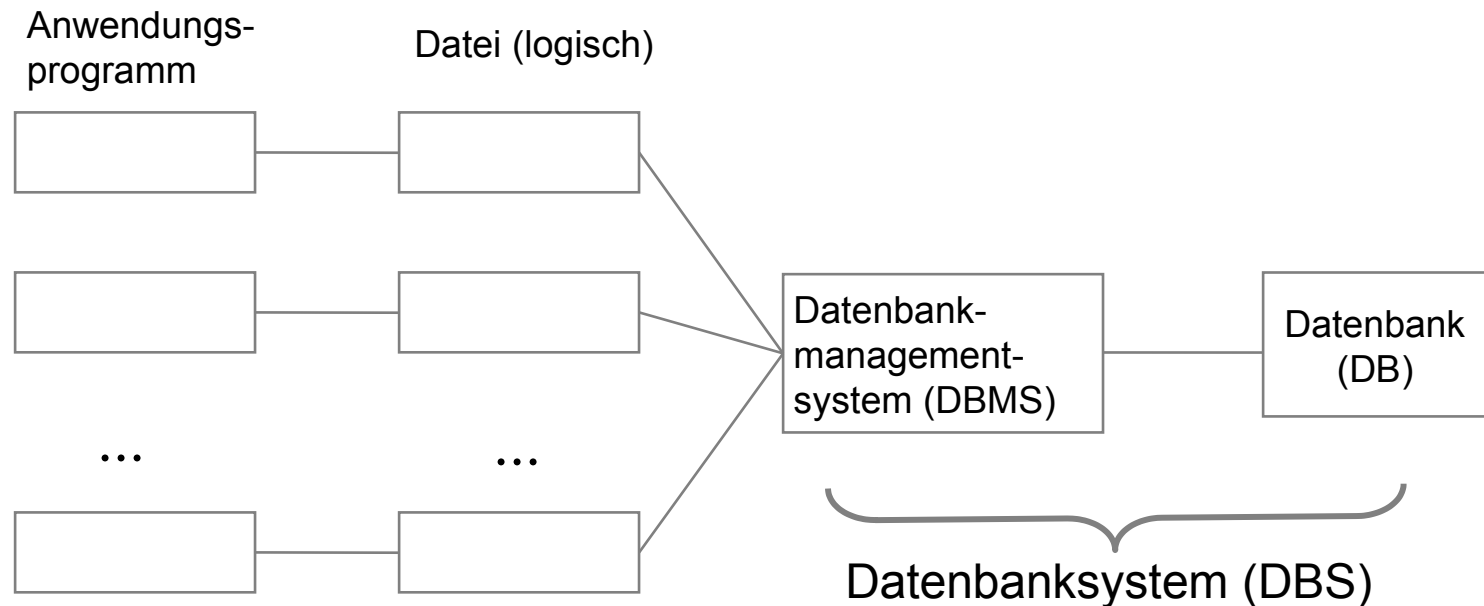
Beispiel: Der F&E-Vorstand (vgl. Seite 13) wird wohl auch weiterhin seine Rundschreiben an Herrn Truhels alten Arbeitsplatz schicken.

Die Konsistent-Haltung aller Abkömmlinge von PERSONAL über einen längeren Zeitraum ist in der Realität kaum zu gewährleisten.

Datenbanksysteme

Konzept des Datenbanksystems:

Integration des Datenbestandes, zentrale Verwaltung der Daten; Trennung von Programm und physischer Datenorganisation:



→ Datenunabhängigkeit

Datenbanksysteme

Dateien werden durch **Tabellen (Relationen)** ersetzt

- diese können jederzeit modifiziert werden
- das Datenbanksystem übernimmt:
 1. allfällige Änderungen der Datenstruktur
 2. Integrität der Daten (semantische Integritätsbedingungen)
 3. Zugriffsberechtigungen

Tabelle: „PERSONAL“

Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	F&E	2	43.500
Jöndhard	739	Karben	Kfz	F&E	2	45.300
Frits	17	Fürth	Mechanik	Contr	4	90.000

Datenbanksysteme – Integritätsbedingungen

I. Semantische Integritätsbedingungen – Wertebereiche (Domäne)

Tabelle: „STANDORT“					
Standort	PLZ	Straße	Koord	Leiter	PersBudget
Aholming	94527	Bäregasse 22	48.47N:12.59E	Beutel	560.000
München	81523	Codd-Weg 9	48.07N:11.38E	Schmitz	900.000
Karben	61184	Nusshof 17	50.32N:08.71E	Dieler	120.000
Fürth	90763	Maierring 109	49.23N:10.61E	Gabler	389.200

Beispiele für semantische Integritätsbedingungen

- Innerhalb einer Tabellenzeile:
Wertebereichsangaben - z.B. PLZ muss zwischen 01000 und 99999 liegen
(statische Bedingung)
- Gehalt eines Angestellten darf bei Änderungen nie kleiner werden
(dynamische Bedingung)

Datenbanksysteme – Integritätsbedingungen

II. Semantische Integritätsbedingungen – Innerhalb der Tabelle

Zellenübergreifen:

- Alle Angestellten, die am selben Standort in derselben Abteilung arbeiten, haben ihren Arbeitsplatz im selben Gebäude
- Je Standort gehören alle Mitarbeiter demselben Unternehmensbereich an.

Tabelle: „PERSONAL“

Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	F&E	2	43.500
Jöndhard	739	Karben	Kfz	F&E	2	45.300
Frits	17	Fürth	Mechanik	Contr	4	90.000

Datenbanksysteme – Integritätsbedingungen

Bsp. (für semantische Integritätsbedingungen):

III. Semantische Integritätsbedingungen – Tabellenübergreifend

Die Summe der Gehälter aller Mitarbeiter an einem Standort darf das Personalbudget dieses Standorts nicht überschreiten

Tabelle: „PERSONAL“						
Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	F&E	2	43.500
Jöndhard	739	Karben	Kfz	F&E	2	45.300
Frits	17	Fürth	Mechanik	Contr	4	90.000

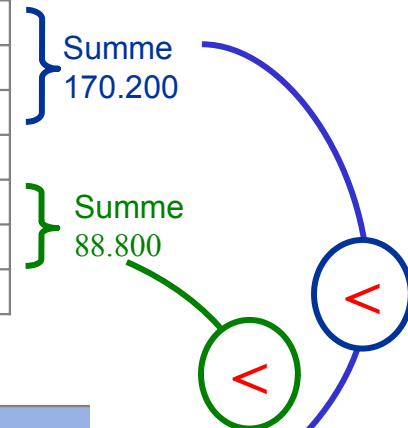


Tabelle: „STANDORT“					
Standort	PLZ	Straße	Koord	Leiter	PersBudget
Aholming	94527	Bärengasse 22	48.47N:12.59E	Beutel	560.000
München	81523	Codd-Weg 9	48.07N:11.38E	Schmitz	900.000
Karben	61184	Nusshof 17	50.32N:08.71E	Dieler	120.000
Fürth	90763	Maierring 109	49.23N:10.61E	Gabler	389.200

Datenbanksysteme – Integritätsbedingungen

III. Semantische Integritätsbedingungen – Tabellenübergreifend

In der PERSONAL-Tabelle dürfen nur solche Standorte auftreten, die in der Tabelle Standort vorkommen (Fremdschlüsselbedingung).

Tabelle: „PERSONAL“						
Name	PersNr	StOrt	UBereich	Abt	GebNr	Gehalt
Frits	17	Aholming	Elektro	F&E	11	44.000
Frans	9133	Aholming	Elektro	Contr	11	88.200
Lubbe	321	Aholming	Elektro	Vertr	8	38.000
Enzian	17	München	Mechanik	F&E	2	53.000
Truhel	54	Karben	Kfz	F&E	2	43.500
Jöndhard	739	Karben	Kfz	F&E	2	45.300
Frits	17	Fürth	Mechanik	Contr	4	90.000

Tabelle: „STANDORT“					
Standort	PLZ	Straße	Koord	Leiter	PersBudget
Aholming	94527	Bäregasse 22	48.47N:12.59E	Beutel	560.000
München	81523	Codd-Weg 9	48.07N:11.38E	Schmitz	900.000
Karben	61184	Nusshof 17	50.32N:08.71E	Dieler	120.000
Fürth	90763	Maierring 109	49.23N:10.61E	Gabler	389.200

Datenbanksysteme – Mehrbenutzerbetrieb

I. Transaktionsverwaltung

Dient dazu, mehreren Benutzern ein konfliktfreies Arbeiten zu ermöglichen.

Transaktion (transaction)

Unter dem Begriff der Transaktion versteht man an Integritätsregeln gebundene Datenbankoperationen, die Datenbestände konsistenzhaltend nachführen.

Datenbanksysteme sind in der Regel **ACID konform**.

1. Dies garantiert jedem Anwender konsistente Datenbankzustände. Zwischenzeitlich inkonsistente Zustände bleiben nach aussen unsichtbar und werden im Fehlerfall rückgängig gemacht.
2. NOSQL Datenbanken verzichten oft auf die Einhaltung von ACID Konformität und gleichen dies zum Beispiel durch eine redundante Datenhaltung aus.

Datenbanksysteme – Mehrbenutzerbetrieb

1. **Atomarität (Atomicity)**

Eine Transaktion wird entweder komplett durchgeführt, oder sie hinterlässt keine Spuren ihrer Wirkung auf die Datenbank.

2. **Konsistenz (Consistency)**

Während der Transaktion können einzelne Konsistenzbedingungen zeitweise verletzt sein, bei Transaktionsende müssen jedoch alle wieder erfüllt sein.

3. **Isolation (Isolation)**

Das Prinzip der Isolation verlangt, dass gleichzeitig ablaufende Transaktionen dieselben Resultate wie im Falle einer Einbenutzerumgebung erzeugen müssen.

4. **Dauerhaftigkeit (Durability)**

Datenbankzustände müssen so lange gültig sein und erhalten bleiben, bis sie von Transaktionen verändert werden.

Datenbanksysteme – Mehrbenutzerbetrieb

II. Serialisierbarkeit

Im Mehrbenutzerbetrieb kommt es vor, dass durch gleichzeitigen Zugriff mehrerer Benutzer auf die selben Daten Konflikte entstehen.

Ein System gleichzeitig ablaufender Transaktionen heisst, **korrekt synchronisiert**, wenn es eine serielle Ausführung gibt, die denselben Datenbankzustand erzeugt.

Bei parallel ablaufenden Transaktionen garantiert das **Prinzip der Serialisierbarkeit**, dass die Resultate auf den Datenbanken identisch sind, gleichgültig ob die Transaktion streng nacheinander ausgeführt worden sind oder nicht.

Parallele
Prozesse,
Transaktionen
oder Benutzer
gleichzeitig



Serielle Ausführung



Datenbanksysteme – Mehrbenutzerbetrieb

Locking

Eine Transaktion kann sich gegenüber anderen absichern, indem sie durch Sperren die lesenden oder zu verändernden Objekte vor weiteren Zugriffen schützt.

Exklusives Sperren (exclusive locks)

- Ein bestimmtes Objekt wird ausschliesslich von einer Transaktion bearbeitet, die übrigen konkurrierenden Transaktionen müssen warten bis das Objekt wieder frei ist.

Sperrprotokoll (locking protokoll)

- In einem Sperrprotokoll wird festgehalten, auf welche Art und Weise Sperren verhängt bzw. aufgehoben werden.
- Falls die Sperrung zu früh zurückgegeben wird, können nicht-serialisierbare Abläufe entstehen.
- Blockieren sich mehrere Transaktionen gegenseitig entsteht eine deadlock Situation.

Mehrbenutzerbetrieb

- Für das **exklusive Sperren** von Objekten sind die beiden Operationen Lock und Unlock notwendig.
- Grundsätzlich muss jedes Objekt gesperrt werden, bevor eine Transaktion darauf zugreift.
- Zweiphasen-Sperrprotokoll (two-phase locking protocol)
Das Zweiphasen-Sperrprotokoll untersagt einer Transaktion, nach dem ersten UNLOCK (entsperren) ein weiteres LOCK (sperren) zu verlangen.

Datenbanksysteme

Beurteilung des Datenbank-Ansatzes

1. Vermeidung von Redundanz:
 - Trennung von Programm und Daten
 - integrierte Verwaltung der Daten, um redundante Datenspeicherung zu vermeiden (Duplikate nicht notwendig).
2. Sicherstellung der Aktualität für alle:
Zentrale Verwaltung bewirkt aktuellen und gleichen Stand der Daten für alle Benutzer.
3. Infrastruktur für den parallelen Zugriff durch mehrere Benutzer und Prozesse (Transaktionen, Locking, etc.)

Datenbanksysteme

3. Flexibler Gebrauch von Daten :

- Daten werden dem Benutzer vom DBMS in der von ihm gewünschten Form übergeben (Views).
- Neue Anwendungen sind leichter zu implementieren, da Datenverwaltung nicht auf einzelne Anwendungen zugeschnitten ist.
- Grad der Flexibilität eines DBS ist allerdings abhängig vom Datenmodell und den verwendeten Speichertechniken.

4. Sicherung der Integrität:

- Integrität der Datenbank – Korrektheit und Vollständigkeit der gespeicherten Daten.
- zentrale Kontrollmöglichkeit durch spezielle Programme des DBMS.

Datenbankmodellierung

Modellierung – Betrachtungsebenen

Reale Welt

Teile der realen Welt



Informationen

über Teile der realen Welt, Modelle

BIBEL

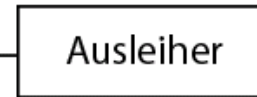
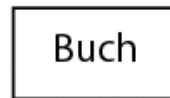
an



A. Hug

Logisches Datensystem

Schema



Physisches Datensystem

physische Datensätze und Dateien

Computer

Hard- und Software

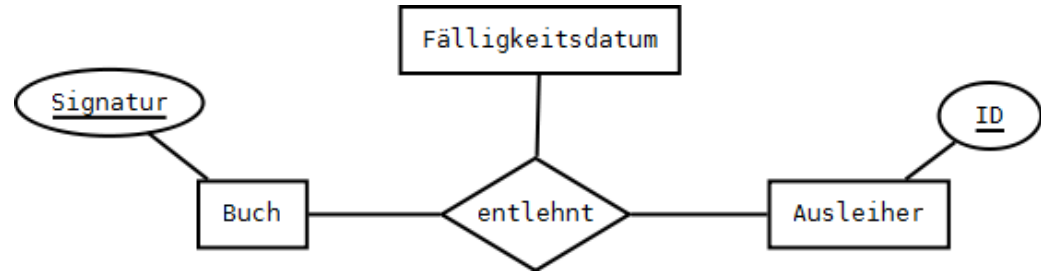
Speichermedien

Datenbank

Datenbankmodellierung – Workflow

I. Entity Relationship (ER)

Model



II. Relationale Model

Signatur	Titel	Autor	...
78912A	SQL	W. Panny	...
12931A	Semantic Web	J. Tusek	...



III. Structured Query Language (SQL)

```
CREATE TABLE Buch (  
  signatur CHAR(12) PRIMARY KEY,  
  title VARCHAR NOT NULL,  
  ... );
```

Quellenangabe

- Foliensatz Prof Stucky (Karlsruhe Institute of Technology – Institut für Angewandte Informatik und Formale Beschreibungsverfahren)
- Foliensatz Prof Panny & Prof Weichselbraun (Wirtschaftsuniversität Wien - Institut für Informationswirtschaft)
- Foliensatz Prof Bischof & Prof Studer (HTW Chur)
- Meier, Andreas (2010): *Relationale und postrelationale Datenbanken*. Berlin / Heidelberg / New York: Springer