



Relationale Datenbanken - Grundlagen

Stephan Karrer

Verwaltung von Daten ohne Datenbanksysteme

- **Große Datenmengen:** Andere Software-Systeme (Dateisystem, Tabellenkalkulation, ...) können große Mengen nicht effizient verwalten
- **Paralleler Zugriff:** Einschränkungen im Mehrbenutzer-Betrieb
- **Redundanz:** Verschiedene Anwendungen speichern dieselben Daten
- **Datenschutz:** Häufig fehlen Mechanismen zur Verhinderung unberechtigter Zugriffe
- **Verfügbarkeit:** Bei Software- oder Hardware-Fehlern gehen Daten verloren
- **Unabhängigkeit vom internen Format:** Anwendungen oder Benutzer sind vom Speicherungsformat der Daten abhängig

Grundmerkmale von Datenbanksystemen

- Verwalten persistente (langfristig zu haltende) Daten
- Verwalten große Datenmengen effizient
- Datenbankmodell, mit dessen Konzepten alle Daten einheitlich beschrieben werden (Integration)
- Operationen und Sprachen deskriptiv, getrennt von einer Programmiersprache
- Transaktionskonzept, Concurrency Control: logisch zusammenhängende Operationen atomar (unteilbar), Auswirkungen langlebig, können parallel durchgeführt werden
- Datenschutz, Datenintegrität (Konsistenz), Datensicherheit

Relationale Datenbanksysteme

- Sind der heute übliche Standard
- Einfaches Datenmodell (relationales Modell):
 - Alle Daten in Tabellen
 - Beziehungen werden durch Integritätsbedingungen abgebildet
 - Mathematisch fundierte Grundlage (die der Endbenutzer nicht kennen muss)
- Einfache Datenbanksprache (SQL)
 - Benutzer beschreibt, was er haben möchte (deskriptiv): **Was statt Wie !**
 - System interpretiert die Anfrage, so dass die Kosten möglichst minimal (Optimierer)
 - Integritäts- und Konsistenzprüfungen durch das System
- Wichtige Vertreter:
 - IBM DB2, Oracle, MS SQL-Server, Sybase, MySQL, PostgreSQL,

Alle Daten logisch in Tabellen

EMPLOYEE_ID	LAST_NAME	SALARY	MANAGER_ID
202	Fay	6000	201
200	Whalen	4400	101
205	Higgins	12000	101
101	Kochhar	17000	100
102	De Haan	17000	100
124	Mourgos	5800	100
149	Zlotkey	10500	100
201	Hartstein	13000	100

- Jede Zeile (row) enthält einen Datensatz (record)
- Jede Spalte (column) hat einen Datentyp und einen Namen
- Die Attribute (Felder) des Datensatzes sind vom jeweiligen Spaltentyp
- In der Regel wird ein Attribut (oder auch Attributkombination) mit seinem Wert die jeweilige Zeile eindeutig identifizieren: **Primärschlüssel (primary key)**
- Ein Attributwert kann leer sein: **NULL-Wert**

Beziehungen zwischen Tabellen

Tabelle: EMPLOYEES

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	DEPARTMENT_ID
174	Ellen	Abel	80
142	Curtis	Davies	50
102	Lex	De Haan	90
104	Bruce	Ernst	60
202	Pat	Fay	20
206	William	Gietz	110

↑
Primärschlüssel

↑
Fremdschlüssel

Tabelle: DEPARTMENTS

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	Administration	200	1700
20	Marketing	201	1800
50	Shipping	124	1500
60	IT	103	1400
80	Sales	149	2500
90	Executive	100	1700
110	Accounting	205	1700
190	Contracting		1700

↑
Primärschlüssel

- Die Daten werden in der Regel auf mehrere Tabellen verteilt, um Redundanzen zu vermeiden (sog. Normalisierung)
- Der Wert in der Fremdschlüsselspalte der Tabelle "EMPLOYEES" verweist auf den zugehörigen Datensatz (Primärschlüssel) in der Tabelle "DEPARTMENTS"
- Wenn wir dem System dies mitteilen, kann es diese Beziehung überwachen (referentielle Integrität)

Datenbankentwurf: Entity-Relationship-Modell

P.P. Chen im Jahre 1976

- **Entity**: Objekt der realen oder der Vorstellungswelt, über das Informationen zu speichern sind, z.B. Vorlesungsveranstaltungen, Buch, Lehrperson, ...
Auch Informationen über Ereignisse: Prüfungen, ...
- **Relationship**: Beziehungen zwischen Entities, z.B. eine Lehrperson hält eine Vorlesung
- **Attribut**: Eigenschaft von Entities oder Beziehungen, z.B. die ISBN eines Buches, der Titel einer Vorlesung, oder das Semester, in dem eine Vorlesung gehalten wird

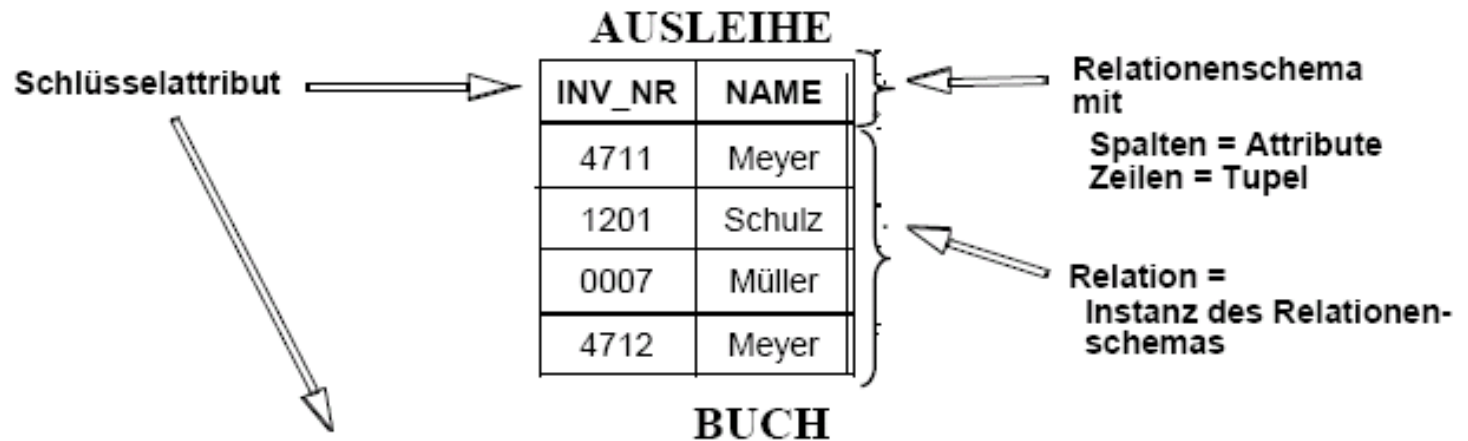
Datenbankentwurf: Entity-Relationship-Modell



- ein Mitarbeiter arbeitet in höchstens einem Raum
- in einem Raum arbeiten höchstens 3 Mitarbeiter
- ein Professor hat maximal 6 Vorlesungstermine (pro Woche)
- eine Vorlesung wird maximal dreimal (pro Woche) gehalten
- an einem Termin können beliebig viele Vorlesungen gehalten werden

Quelle: Prof. Dr. Gregor Engels
Universität Paderborn, Vorlesung

Datenbankentwurf: Relationenschema



INV_NR	TITEL	ISBN	AUTOR
0007	Dr. No	3-125...	James Bond
1201	Objektbanken	3-111...	Heuer
4711	Datenbanken	3-765...	Vossen
4712	Datenbanken	3-891...	Ullmann
4717	PASCAL	3-999...	Wirth

Datenbank (DB) = Menge von Relationen

DB-Schema = Menge von Relationenschemata

Datenbankentwurf: Relationenschema und Integritätsbedingungen

lokale Integritätsbedingungen:

- ☐ zu jedem Attribut in jedem Relationenschema gibt es eine Typdefinition, die die zulässigen Werte der entsprechenden Spalte festlegt (String, Integer, ...)
- ☐ Attribut INV_NR ist **(Primär-)Schlüssel** von BUCH
 - ☐ in BUCH keine zwei Tupel mit demselben INV_NR-Wert
- ☐ Attribut INV_NR ist **(Primär-)Schlüssel** von AUSLEIHE
 - ☐ in AUSLEIHE keine zwei Tupel mit demselben INV_NR-Wert

globale Integritätsbedingungen:

- ☐ jeder INV_NR-Wert des „Fremdschlüssels“ von AUSLEIHE muß auch in BUCH auftreten

Quelle: Prof. Dr. Gregor Engels
Universität Paderborn, Vorlesung