

**Datenbank-Grundlagen**

SS 2005

**9. Datenbank-Schemaentwurf**  
**Prof. Dr. Stefan Böttcher**  
**Universität Paderborn**

mit Material von  
**Prof. Dr. Gregor Engels**

**Agenda:**

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 1

**Relationaler Datenbank-Entwurf**

- Beschreibt die zu verwaltenden Daten in Relationenschemata, unabhängig von einem konkretem SQL-Dialekt
- Transformiert diese, um Redundanz zu verringern.

**Schritte**

1. **E/R-Modell** → **Relationales Modell** (Ausgangsschemata)
  - aus integriertem Schemata des konzeptionellen Entwurfs (ER-Diagramm) abgeleitet oder *ad hoc* entworfen
2. **Normalisierung** → Normalisierte Schemata
  - durch wiederholte Analyse und Zerlegung (Normalisierung) aus Ausgangsschemata gewonnen

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 2

**Beispiel : Redundanz der universellen Relation**

Student	Adresse	Kurs	Professor	Fbr
Anna Arm	Hofweg 7, Ulm	ti1Mathe1	Fränkel	ti
Anna Arm	?	ti2Physik2	Beckmann	ti
Anna Arm	?	ti2Prog2	Kratzer	ti
Anna Arm	?	ti2Digital1	Frey	ti
Rita Reich	Hofweg 7, Ulm	ti1Mathe1	?	?
Rita Reich	?	ti2Physik2	?	?
Rita Reich	?	ti2Prog2	?	?
Norbert Neu	Umweg 3, Aalen	ti1Mathe1	?	?
Norbert Neu	?	ti2Physik2	?	?
Norbert Neu	?	ti2Prog2	?	?
Norbert Neu	?	ti2Digital1	?	?
Hugo Held	Heuweg 1, Ulm	ti2Mathe2	Fränkel	?

Jedes "?" steht für einen redundanten (errätbaren, schon bekannten) Wert

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 3

**Beispiel : Redundanz der universellen Relation**

Student	Adresse	Kurs	Professor	Fbr
Anna Arm	Hofweg 7, Ulm	ti1Mathe1	Fränkel	ti
Anna Arm	Hofweg 7, Ulm	ti2Physik2	Beckmann	ti
Anna Arm	Hofweg 7, Ulm	ti2Prog2	Kratzer	ti
Anna Arm	Hofweg 7, Ulm	ti2Digital1	Frey	ti
Rita Reich	Hofweg 7, Ulm	ti1Mathe1	Fränkel	ti
Rita Reich	Hofweg 7, Ulm	ti2Physik2	Beckmann	ti
Rita Reich	Hofweg 7, Ulm	ti2Prog2	Kratzer	ti
Norbert Neu	Umweg 3, Aalen	ti1Mathe1	Fränkel	ti
Norbert Neu	Umweg 3, Aalen	ti2Physik2	Beckmann	ti
Norbert Neu	Umweg 3, Aalen	ti2Prog2	Kratzer	ti
Norbert Neu	Umweg 3, Aalen	ti2Digital1	Frey	ti
Hugo Held	Heuweg 1, Ulm	ti2Mathe2	Fränkel	ti

Es würde genügen, zu jedem Student seine Adresse nur 1x zu speichern, ...

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 4

**Delete-Anomalie: Folge der Redundanz**

In der Regel werden die Schlüsselattribute (hier Student und Kurs) als NOT NULL definiert, d.h., sie dürfen nicht mit Nullwerten besetzt werden.

Beispiel für *Delete-Anomalie*: 'Rita Reich' meldet sich von allen Kursen ab (Urlaub)  
→ kein Tupel ist mehr vorhanden mit dem Attributwert 'Rita Reich' für Student.  
→ Auch die Adresse von 'Rita Reich' geht verloren!  
Das Löschen von Tupeln führt zu zusätzlichem unerwünschten Informationsverlust.

Student	Adresse	Kurs	Professor	Fbr
...	...	...	...	...
Rita Reich	Hofweg 7, Ulm	ti1Mathe1	Fränkel	ti
Rita Reich	Hofweg 7, Ulm	ti2Physik2	Beckmann	ti
Rita Reich	Hofweg 7, Ulm	ti2Prog2	Kratzer	ti
...	...	...	...	...

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 5

**Insert-/Update-Anomalie: Folgen der Redundanz**

Beispiel für *Insert-Anomalie*:  
Will man einen neuen Professor einstellen, der (noch) keinen Kurs anbietet, so ist das (ohne Nullwerte für Student, Adresse und Kurs) nicht möglich, insbesondere kann man nicht die Information speichern, in welchem Fbr der Professor eingestellt wird.

Student	Adresse	Kurs	Professor	Fbr
???	???	???	Neumann	ti
...	...	...	...	...
Rita Reich	Hofweg 7, Ulm	ti1Mathe1	Fränkel	ti
Rita Reich	Kamp 32, PB	ti2Physik2	Beckmann	ti
Rita Reich	Hofweg 7, Ulm	ti2Prog2	Kratzer	ti
...	...	...	...	...

Beispiel für *Update-Anomalie*: Rita Reich zieht um (nur 1x ändern)  
Anomalien vermeidbar, indem man die Relation in redundanzfreie Teile zerlegt.

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 6

### Beispiel: 1. Zerlegung der universellen Relation

Student --> Adresse		Student	Kurs --> Professor --> Fbr
Anna Arm	Hofweg 7, Ulm	Anna Arm	t1Mathe1 Fränkel ti
Rita Reich	Hofweg 7, Ulm	Anna Arm	t2Physik2 Beckmann ti
Norbert Neu	Umweg 3, Aalen	Anna Arm	t2Prog2 Kratzer ti
Hugo Held	Heuweg 1, Ulm	Anna Arm	t1Digital1 Frey ti
		Rita Reich	t1Mathe1 Fränkel ti
		Rita Reich	t2Physik2 Beckmann ti
		Rita Reich	t2Prog2 Kratzer ti
		Norbert Neu	t1Mathe1 Fränkel ti
		Norbert Neu	t2Physik2 Beckmann ti
		Norbert Neu	t2Prog2 Kratzer ti
		Norbert Neu	t1Digital1 Frey ti
		Hugo Held	t2Mathe2 Fränkel ti

Redundante Adress-Einträge sind beseitigt, aber es würde auch genügen, zu jedem Kurs den Prof und zu jedem Prof seinen Fbr nur 1x zu speichern.

Redundanz und Schemazerlegung Funktionale Abhängigkeit Schema-Eigenschaften Transformationsseigenschaften Entwurfsaufgabe  
Datenbank Grundlagen SS2005 Kapitel 9 - Datenbank-Schema-Entwurf / 7

### Vollständige Zerlegung der universellen Relation

Student --> Adresse		Student	Professor
Anna Arm	Hofweg 7, Ulm	Anna Arm	Fränkel
Rita Reich	Hofweg 7, Ulm	Anna Arm	Beckmann
Norbert Neu	Umweg 3, Aalen	Anna Arm	Kratzer
Hugo Held	Heuweg 1, Ulm	Anna Arm	Frey
		Rita Reich	Fränkel
		Rita Reich	Beckmann
		Rita Reich	Kratzer
		Norbert Neu	Fränkel
		Norbert Neu	Beckmann
		Norbert Neu	Kratzer
		Norbert Neu	Frey
		Hugo Held	Fränkel

Welcher Student hört in welchem Fbr ?  
ist rekonstruierbar durch verlustfreien Join (Student,Professor) |X| (Professor, Fbr)  
Aber:  
Welcher Student hört welchen Kurs ist nicht durch verlustfreien Join rekonstruierbar  
(Student,Professor) |X| (Kurs,Professor) bringt zu viele Kurse (Informationsverlust)

Redundanz und Schemazerlegung Funktionale Abhängigkeit Schema-Eigenschaften Transformationsseigenschaften Entwurfsaufgabe  
Datenbank Grundlagen SS2005 Kapitel 9 - Datenbank-Schema-Entwurf / 8

### Verlustfreie Zerlegung der universellen Relation

Student --> Adresse		Student	Kurs
Anna Arm	Hofweg 7, Ulm	Anna Arm	t1Mathe1
Rita Reich	Hofweg 7, Ulm	Anna Arm	t2Physik2
Norbert Neu	Umweg 3, Aalen	Anna Arm	t2Prog2
Hugo Held	Heuweg 1, Ulm	Anna Arm	t1Digital1
		Rita Reich	t1Mathe1
		Rita Reich	t2Physik2
		Norbert Neu	t1Mathe1
		Norbert Neu	t2Physik2
		Norbert Neu	t2Prog2
		Norbert Neu	t1Digital1
		Hugo Held	t2Mathe2

Die universelle Relation (und folglich alle Projektionen davon) sind durch verlustfreien Join rekonstruierbar  
(Student,Adresse) |X| (Student,Kurs) |X| (Kurs,Professor) |X| (Professor, Fbr)

Redundanz und Schemazerlegung Funktionale Abhängigkeit Schema-Eigenschaften Transformationsseigenschaften Entwurfsaufgabe  
Datenbank Grundlagen SS2005 Kapitel 9 - Datenbank-Schema-Entwurf / 9

### Referentielle Integritätsbedingungen

Student --> Adresse		Student	Kurs
...	...	...	...

Zerlegung führt zu *referentiellen Integritätsbedingung* :

- Nur angemeldete Studenten können Kurse belegen:  
 $P_{Student} (Student,Kurs) \subseteq P_{Student} (Student,Adresse)$
- Kurse können nur von einem Fbr angehörigen Professoren gehalten werden  
 $P_{Professor} (Kurs,Professor) \subseteq P_{Professor} (Professor, Fbr)$
- Studenten können nur angebotene Kurse belegen:  
 $P_{Kurs} (Student,Kurs) \subseteq P_{Kurs} (Kurs,Professor)$

Redundanz und Schemazerlegung Funktionale Abhängigkeit Schema-Eigenschaften Transformationsseigenschaften Entwurfsaufgabe  
Datenbank Grundlagen SS2005 Kapitel 9 - Datenbank-Schema-Entwurf / 10

### Referentielle Integritätsbedingungen in SQL

Referentielle Integritätsbedingungen lassen sich in Algebra, ..., SQL formulieren

Nur angemeldete Studenten können Kurse belegen:  
 $P_{Student} (Student,Kurs) \subseteq P_{Student} (Student,Adresse)$   
 $P_{Student} (Student,Kurs) - P_{Student} (Student,Adresse) = \emptyset$

Assert NOT EXISTS ( select \* from StudentKurs SK  
where SK.Student NOT IN  
( select S from StudentAdresse )

Wenn gleichzeitig  
das referenzierte Attribut Schlüssel der referenzierten Relation ist,  
kann referentielle Integritätsbedingung  
auch durch Foreign Key-Constraint in SQL ausgedrückt werden.

Redundanz und Schemazerlegung Funktionale Abhängigkeit Schema-Eigenschaften Transformationsseigenschaften Entwurfsaufgabe  
Datenbank Grundlagen SS2005 Kapitel 9 - Datenbank-Schema-Entwurf / 11

### Referentielle Integritätsbedingungen in SQL (2)

Nur angemeldete Studenten können Kurse belegen:  
 $P_{Student} (Student,Kurs) \subseteq P_{Student} (Student,Adresse)$

create table StudentAdresse ( Student char(20) primary key , Adresse char(20) ) ;

create table StudentKurs ( Student char(20) , Kurs char(20) ,  
foreign key Student references StudentAdresse  
on delete set null,  
foreign key Kurs references KursProfessor  
on delete cascade ) ;

on delete set null - ersetzt Wert für Student in StudentKurs durch NULL  
on delete set default - ersetzt Wert für Student in StudentKurs durch Defaultwert  
on delete cascade - löscht auch Tupel in StudentKurs  
on delete no action - Tupel in StudentAdresse ist nicht löscher,  
(no action ist Default) solange Tupel in StudentKurs es referenzieren

ein no action-Eintrag hat Vorrang vor Kette kaskadierender deletes

Redundanz und Schemazerlegung Funktionale Abhängigkeit Schema-Eigenschaften Transformationsseigenschaften Entwurfsaufgabe  
Datenbank Grundlagen SS2005 Kapitel 9 - Datenbank-Schema-Entwurf / 12

### Bücher-Relation mit Redundanzen

Bücher

ISBN	Titel	Autor	Version	Stichwort	Verlagsname
0-8053-1753-8	Princ. of DBS	Elmasri	1,1989	RDB	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Navathe	1,1989	RDB	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Elmasri	2,1994	RDB	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Navathe	2,1994	RDB	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Elmasri	1,1989	Lehrbuch	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Navathe	1,1989	Lehrbuch	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Elmasri	2,1994	Lehrbuch	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Navathe	2,1994	Lehrbuch	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Elmasri	1,1989	ER	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Navathe	1,1989	ER	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Elmasri	2,1994	ER	Benj.Cumm.
0-8053-1753-8	Princ. of DBS	Navathe	2,1994	ER	Benj.Cumm.

**Problem:** insert-, delete- und update-Anomalien  
**Beispiel:** Einfügen einer neuen Version 3, 1996

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 13

### Funktionale Abhängigkeiten

(kurz: FD, von *functional dependency*)

Für eine Relation  $r(R)$  und Attributmengen  $X, Y \subseteq R$  heißt  $Y$  *funktional abhängig von X*, kurz  $X \rightarrow Y$ , falls für alle Tupel der Relation  $r$  die Werte der Attribute aus  $X$  diejenigen aus  $Y$  eindeutig festlegen.

Formal:  $\forall t_1, t_2 \in r: t_1(X) = t_2(X) \Rightarrow t_1(Y) = t_2(Y)$

Wir sagen auch:  
 $r$  erfüllt die *funktionale Abhängigkeit*  $X \rightarrow Y$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 14

### Beispiel zu Funktionalen Abhängigkeiten

In der Tabelle *Bücher* :  
 $ISBN \rightarrow Titel, Verlagsname$

Nicht:  
 $ISBN \rightarrow Autor, Stichwort$

Trivialerweise:  
 $ISBN \rightarrow ISBN$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 15

### Schlüssel als Spezialfall einer FD

Für Beispiel auf folgender Folie  
 $PANr \rightarrow Vorname, Nachname, PLZ, Ort, Straße, Hausnummer, Geburtsdatum$

- Immer:  $PANr \rightarrow PANr$ , damit gesamtes Schema auf rechter Seite
- Wenn linke Seite minimal: Schlüssel

Formal: Schlüssel  $X$  liegt vor, wenn für Relationenschema  $R$   $FD X \rightarrow R$  gilt und  $X$  minimal  
Ziel des Datenbankentwurfs: alle gegebenen funktionalen Abhängigkeiten in „Schlüsselabhängigkeiten“ umformen, ohne dabei semantische Information zu verlieren

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 16

### Schlüssel im Beispiel

Personen

PANr	Vorname	Nachname	PLZ	Ort	Straße	HNr	Geb.datum
4711	Andreas	Heuer	18209	DBR	BHS	15	31.10.58
5588	Gunter	Saake	39106	MD	STS	55	05.10.60
6834	Michael	Korn	39104	MD	BS	41	24.09.74
7754	Andreas	Möller	18209	DBR	RS	31	25.02.76
8832	Tamara	Jagellovsk	38106	BS	GS	12	11.11.73
9912	Anje	Heilhof	18059	HRO	AES	21	04.04.70
9999	Christa	Loeser	69121	HD	TS	38	10.05.69

Pers\_Telefon

PANr	Telefon
4711	038203-12230
4711	0381-498-3401
4711	0381-498-3427
5588	0391-345677
5588	0391-4592-3800
9999	06221-400177

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 17

### Partielle und transitive Abhängigkeiten

Partielle funktionale Abhängigkeit:  
linke Seite von  $X \rightarrow Y$  ist nicht minimal, d.h. es gibt  $FD X_0 \rightarrow Y$  mit  $X_0 \subset X$

Beispiel:  $ISBN, Autor, Version, Stichwort \rightarrow Titel, Verlagsname, ISBN \rightarrow Titel, Verlagsname$   
aber auch

Transitive funktionale Abhängigkeit:  
 $X \rightarrow Y$  als indirekter Zusammenhang, d.h. es gibt  $FD X \rightarrow Z$  und  $Z \rightarrow Y$

Beispiel:  $PANr \rightarrow PLZ \rightarrow Ort$

Wenn eines davon innerhalb eines Relationenschemas vorkommt,  
 $\rightarrow$  Anzeichen von Redundanz!

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 18

### Ableitung von FDs

r	A	B	C
	a <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>
	a <sub>2</sub>	b <sub>1</sub>	c <sub>1</sub>
	a <sub>3</sub>	b <sub>2</sub>	c <sub>1</sub>
	a <sub>4</sub>	b <sub>1</sub>	c <sub>1</sub>

- genügt  $A \rightarrow B$  und  $B \rightarrow C$
- dann gilt auch (transitiv)  $A \rightarrow C$ 
  - ➔ Vorteil:  $A \rightarrow C$  muss man nicht mehr testen
- nicht ableitbar  $C \rightarrow A$  oder  $C \rightarrow B$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 19

### Ableitung von FDs

**Bezeichnungen:**

- $F$  Menge von funktionalen Abhängigkeiten
- $f$  funktionale Abhängigkeit
- $SAT_R(F)$  Menge aller Relationen über dem Relationenschema  $R$ , die die funktionalen Abhängigkeiten aus  $F$  erfüllen.

Gilt für  $f$  über  $R$ :  $SAT_R(F) \subseteq SAT_R(f)$ ,  
dann impliziert  $F$  die funktionale Abhängigkeit  $f$ ; kurz:  $F \models f$

Obiges Beispiel:  
 $F = \{A \rightarrow B, B \rightarrow C\} \not\models A \rightarrow C$

**Bemerkung:** Falls  $F = \{\}$ , gilt  $SAT_R(\{\}) = REL(R)$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 20

### Transitive Hülle einer Menge von FDs

Die (transitive) Hülle einer Menge von funktionalen Abhängigkeiten  $F$  ist definiert durch

$$F^+ := \{f \mid f \text{ funktionale Abhängigkeit} \wedge F \models f\}$$

**Im vorigen Beispiel:**

$$\{A \rightarrow C, AB \rightarrow C, A \rightarrow BC, AB \rightarrow AB\} \subseteq F^+$$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 21

### Berechnungsproblem für $F^+$

Ist FD  $X \rightarrow Y$  in der Hülle  $F^+$  einer Menge  $F$ ?     $X \rightarrow Y \in F^+$   
Äquivalent: Wird  $X \rightarrow Y$  von  $F$  impliziert?     $F \models X \rightarrow Y$

Problem: Komplette Berechnung von  $F^+$  zu aufwendig.  
Idee: Die Menge aller von der Attributmenge  $X$  aus „erreichbaren“ (abhängigen) Attribute bestimmen und auf Enthaltensein von  $Y$  testen.

Menge  $X^*$ : aller mit  $F$  von  $X$  abhängigen Attribute:  
Hülle von  $X$  bzgl.  $F$ :  $X^* := \{A \mid X \rightarrow A \in F^+\}$

Es gilt:  $X \rightarrow Y \in F^+$  gdw.  $Y \subseteq X^*$ . Damit kann das Problem in linearer Zeit mit dem RAP-Algorithmus gelöst werden.

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 22

### RAP-Algorithmus

**Frage:**  $X \rightarrow Y \in F^+$  ?

1. Initialisierung (es gilt Reflexivität):  
 $X^* := X$
2. repeat (es gilt Akkumulation):  
Wähle FD  $X_i \rightarrow Y_i \in F$  mit  $X_i \subseteq X^*$ ;  $X^* := X^* \cup Y_i$   
until  $Y_i \subseteq X^*$  or  $X^*$  stabil (Hülle) ist.
3. Projektion:  
Antworte JA wenn  $Y \subseteq X^*$ , ansonsten NEIN.

Algorithmus ist  
korrekt (JA impliziert  $X \rightarrow Y \in F^+$ ) und  
vollständig ( $X \rightarrow Y \in F^+$  impliziert JA).

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 23

### Beispiel zum RAP-Algorithmus

Menge  $F$  gegeben als

- a) Ort, Straße, HNr  $\rightarrow$  PLZ
- b) PANr  $\rightarrow$  Vorname, Nachname, Geb.datum
- c) PANr  $\rightarrow$  Ort, Straße, HNr

Frage: PANr  $\rightarrow$  PLZ  $\in F^+$  ?  
Rechnung:

PANr	(R)
$\rightarrow$ c) PANr, Ort, Straße, HNr	(A)
$\rightarrow$ a) PANr, Ort, Straße, HNr, PLZ	(A)
$\supseteq$ PLZ	(P)

Antwort: JA

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 24

### Hülle einer Attributmenge

Frage:  
Welche Attribute sind bei einer gegebenen Menge  $F$  von FD von Attributen aus  $X$  abhängig (mit transitiven Abhängigkeiten) ?

```
repeat (Akkumulation):
   $X_{alt} := X$  ;
  if ( exists FD  $X_1 \rightarrow Y_1 \in F$  mit  $X_1 \subseteq X$  und  $Y_1 \not\subseteq X$  )
     $X := X \cup Y_1$ 
until  $X_{alt} = X$  ; // d.h.  $X$  ist stabil (Hülle).
return  $X$ .
```

Die durch diesen Algorithmus berechnete Attributmenge heißt  
Attributhülle von  $X$  bezüglich  $F$   
Notation:  $\text{Attributhülle}(F, X)$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 25

### Ableitbarkeit - alternative Formulierung

$X_1, \dots, X_n \twoheadrightarrow Y_1, \dots, Y_m$  ist ableitbar aus  $F$ ,  
genau dann wenn  
 $\{Y_1, \dots, Y_m\} \subseteq \text{Attributhülle}(F, \{X_1, \dots, X_n\})$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 26

### Linksreduktion

Gegeben: eine Menge  $F$  von FDs

Linksreduktion:  
repeat  
if ( exists  $X \twoheadrightarrow Y \in F$  mit  $X_i \in X$  und  $Y \subseteq \text{Attributhülle}(F, X - \{X_i\})$   
//  $X_i$  ist überflüssig in  $X \twoheadrightarrow Y$   
 $F = F \setminus \{X \twoheadrightarrow Y\} \cup \{X - \{X_i\} \twoheadrightarrow Y\}$   
// entferne  $X_i$  aus  $X$  in  $X \twoheadrightarrow Y$  in  $F$   
until  $F$  ändert sich nicht mehr ;

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 27

### Rechtsreduktion

Gegeben: eine Menge  $F$  von FDs

Rechtsreduktion:  
repeat  
if ( exists  $X \twoheadrightarrow Y \in F$  mit  $Y_i \in Y$  und  $Y_i \in \text{Attributhülle}(F \setminus \{X \twoheadrightarrow Y\} \cup \{X \twoheadrightarrow (Y - \{Y_i\})\})$   
//  $Y_i$  ist überflüssig in  $X \twoheadrightarrow Y \in F$   
 $F = F \setminus \{X \twoheadrightarrow Y\} \cup \{X \twoheadrightarrow (Y - \{Y_i\})\}$   
// entferne  $Y_i$  aus  $Y$  in  $X \twoheadrightarrow Y$  in  $F$   
until  $F$  ändert sich nicht mehr ;

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 28

### Minimale Menge von FDs (kanonische Überdeckung)

Gegeben ist eine Menge  $F$  von FDs

$F_{\text{Min}}$  ist minimale Menge von FDs zu  $F$ , wenn

- $F_{\text{Min}}^+ = F^+$
- $F_{\text{Min}}$  keine überflüssigen Attribute enthält (d.h. Linksreduktion und Rechtsreduktion durchgeführt wurde)
- Jede linke Seite einer FD einzigartig ist.

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 29

### Berechnung einer minimalen Menge von FDs

Gegeben ist eine Menge  $F$  von FDs

- Führe Linksreduktion auf  $F$  durch
- Führe Rechtsreduktion auf dem Ergebnis von 1. durch
- entferne aus dem Ergebnis von 2. FDs der Form  $X \twoheadrightarrow \{ \}$
- Fasse  $X \twoheadrightarrow Y_1, \dots, X \twoheadrightarrow Y_n$  zu  $X \twoheadrightarrow Y_1, \dots, Y_n$  zusammen

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 30

### Mehrere minimale Mengen von FDs

Im allgemeinen gibt es mehrere minimale Mengen von FDs zu einer Menge F von FDs.

Beispiel:

$F = \{ A \rightarrow B, B \rightarrow A, B \rightarrow C, C \rightarrow B, A \rightarrow C, C \rightarrow A, A \rightarrow BC \}$

$F1 = \{ A \rightarrow B, B \rightarrow A, B \rightarrow C, C \rightarrow B \}$

$F2 = \{ A \rightarrow B, B \rightarrow C, C \rightarrow A \}$

$F3 = \{ A \rightarrow C, C \rightarrow B, B \rightarrow A \}$

Es gilt:  $F^+ = F1^+ = F2^+ = F3^+$

und alle 3 Mengen sind minimale Mengen von FDs, die äquivalent zu F sind

### Äquivalenz einer Menge von FDs

F heißt äquivalent zu G (kurz:  $F \equiv G$ )

falls  $F^+ = G^+$

$f \in F$  heißt redundant (=ableitbar aus  $F \setminus \{f\}$ )

falls  $F \setminus \{f\} \equiv F$

### Schema-Eigenschaften (Prioritätsliste)

Relationenschemata, Schlüssel und Fremdschlüssel so wählen, dass

1. alle Anwendungsdaten durch verlustfreien Join aus den Basisrelationen hergeleitet werden können (verlustfreie Zerlegung),
2. nur semantisch sinnvolle und konsistente Anwendungsdaten dargestellt werden können und
3. die Anwendungsdaten möglichst redundanzfrei dargestellt werden.

Thema in diesem Abschnitt: 3. Forderung

- Redundanzen der Daten innerhalb einer Relation: Normalformen
- globale Redundanz des DB-Schemas und der FDs: Minimalität

### Update-Anomalien (Wdh.)

Redundanzen in Basisrelationen unerwünscht:

- Belegen unnötigen Speicherplatz (eher unwichtig)
- Information redundant  $\rightarrow$  Änderung muss diese Information in allen ihren Vorkommen verändern (in relationalen Systemen nur schwer zu realisieren)

Beispiel insert-Anomalie:

Bücher	ISBN	Titel	Autor	Version	Stichwort	Verlagsname
	0-8053-1753-8	Princ. of DBS	Elmasri	3, 1996	RDB	null

Weitere Beispiele für delete-, insert-, update-Anomalien?

### Erste Normalform

führt zunächst Redundanzen ein

Erste Normalform (1NF): nur atomare Attribute in Relationenschemata

Invnr	Titel	ISBN	Autoren
0007	Dr. No	3-125	James Bond
1201	Objektbanken	3-111	Heuer, Scholl
4711	Datenbanken	3-765	Vossen, Witt
4712	Datenbanken	3-891	Ullman
4717	Pascal	3-999	Wirth, Dijkstra

wäre in erster Normalform

Invnr	Titel	ISBN	Autor
0007	Dr. No	3-125	James Bond
1201	Objektbanken	3-111	Heuer
1201	Objektbanken	3-111	Scholl
4711	Datenbanken	3-765	Vossen
4711	Datenbanken	3-765	Witt
4712	Datenbanken	3-891	Ullman
4717	Pascal	3-999	Wirth
4717	Pascal	3-999	Dijkstra

### Redundanz und Anomalie

InvNr	Titel	ISBN	Autor
0007	Dr. No	3-125	James Bond
1201	Objektbanken	3-111	Heuer
1201	Objektbanken	3-111	Scholl
4711	Datenbanken	3-765	Vossen
4711	Datenbanken	3-765	Witt
4712	Datenbanken	3-891	Ullman
4717	Pascal	3-999	Wirth
4717	Pascal	3-999	Dijkstra

Relationenschema, funktionale Abhängigkeiten und Schlüssel:

$R = \{InvNr, Titel, ISBN, Autor\}$

$F = \{InvNr \rightarrow ISBN, InvNr, Autor \rightarrow InvNr, Titel, ISBN, Autor\}$

$\rightarrow K_R = \{InvNr, Autor\}$

Eine weitere Art von Anomalie (hier Verletzung einer FD):

4717	Petlitz	4-100	Rozenberg
------	---------	-------	-----------

- Neue Wertekombination (4717, Rozenberg) zum Schlüssel (InvNr, Autor).
- Zu Nicht-Schlüsselattribut ISBN existieren die Werte 3-999 und 4-100.

$\Rightarrow$  Anomalie: die Abhängigkeit  $InvNr \rightarrow ISBN$  ist verletzt.

### Zweite Normalform

Zweite und weitere Normalformen: aufgrund der Struktur von Abhängigkeiten Redundanzen entdecken

**Zweite Normalform:** Keine *partiellen Abhängigkeiten* zwischen einem Schlüssel und Nicht-Schlüsselattributen (d.h. Attributen, die in *keinem* Schlüssel vorkommen)

**Formal:**  $\neg \exists$  Schlüssel  $K$ , Attributmengemenge  $X \subset K$ , Nicht-Schlüsselattribut  $Y$ , so dass  $X \rightarrow Y$

**Im Beispiel:**

- $Invnr \rightarrow ISBN$  und  $Invnr, Autor \rightarrow Invnr, Titel, ISBN, Autor$
- $Invnr$  und  $Autor$  zusammen Schlüssel
- $ISBN$  hängt aber **allein** von  $Invnr$  ab, also Bücher nicht in 2. Normalform!

Zweite Normalform erreichen wir durch Elimination der rechten Seite der partiellen Abhängigkeit und Kopie der linken Seite (siehe nächste Folie)

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 37

### Veranschaulichung: zweite Normalform

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 38

### Redundanz durch partielle Abhängigkeiten

InvNr	Titel	ISBN	Autor
0007	Dr. No	3-125	James Bond
1201	Objektbanken	3-111	Heuer
1201	Objektbanken	3-111	Scholl
4711	Datenbanken	3-765	Vossen
4711	Datenbanken	3-765	Witt
4712	Datenbanken	3-891	Ullman
4717	Pascal	3-999	Wirth
4717	Pascal	3-999	Dijkstra

Relationenschema, funktionale Abhängigkeiten und Schlüssel:

$R = \{InvNr, Titel, ISBN, Autor\}$   
 $F = \{InvNr \rightarrow ISBN, ISBN \rightarrow Titel, Invnr, Autor \rightarrow Invnr, Titel, ISBN, Autor\}$   
 $K_R = \{InvNr, Autor\}$

➡ Redundanz: die Information über die Abhängigkeit  $ISBN \rightarrow Titel$  ist in der Relation mehrfach abgelegt. Dies führt zu update-Anomalien

Problem: zu derselben ISBN gibt es mehrere verschiedene Einträge für  $InvNr$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 39

### Dritte Normalform

**Dritte Normalform:** Keine *transitiven Abhängigkeiten* zwischen einem Schlüssel und Nicht-Schlüsselattributen

**Formal:**  $\neg \exists$  Schlüssel  $K$ , Attributmengemenge  $X \rightarrow K$ , Nicht-Schlüsselattribut  $Y \in X \cup K$ , so dass  $X \not\rightarrow Y$

**Im Beispiel:**

- $ISBN \rightarrow Titel$  und  $Invnr, Autor \rightarrow Invnr, Titel, ISBN, Autor$
- $Invnr$  und  $Autor$  zusammen Schlüssel
- $Titel$  hängt über  $ISBN$  vom Schlüssel ab ( $Invnr, Autor \rightarrow ISBN \rightarrow Titel$ ), also Bücher nicht in 3. Normalform!

Dritte Normalform erreichen durch Elimination von  $Y$  und Kopie von  $X$  (siehe nächste Folie)

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 40

### Veranschaulichung: dritte Normalform

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 41

### Beispiel: Zerlegung in Normalform (1)

Ausgangsrelation:  
 $R = \{InvNr, Titel, ISBN, Autor\}$   
 $F = \{InvNr \rightarrow ISBN, ISBN \rightarrow Titel\}$   
 $K_R = \{InvNr, Autor\}$   
**Nicht 3NF:**  $InvNr, Autor \rightarrow ISBN \rightarrow Titel$

Zerlegung:

InvNr	ISBN	Autor
0007	3-125	James Bond
1201	3-111	Heuer
1201	3-111	Scholl
4711	3-765	Vossen
4711	3-765	Witt
4712	3-891	Ullman
4717	3-999	Wirth
4717	3-999	Dijkstra

ISBN	Titel
3-125	Dr. No
3-111	Objektbanken
3-111	Objektbanken
3-765	Datenbanken
3-765	Datenbanken
3-891	Datenbanken
3-999	Pascal
3-999	Pascal

Neue Relationen:  
 $R1 = \{InvNr, ISBN, Autor\}$   
 $F1 = \{InvNr \rightarrow ISBN\}$   
 $\rightarrow K_{R1} = \{InvNr, Autor\}$   
 $R2 = \{ISBN, Titel\}$   
 $F2 = \{ISBN \rightarrow Titel\}$   
 $\rightarrow K_{R2} = \{ISBN\}$

Nicht 2NF!

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 42

### Beispiel: Zerlegung in Normalform (2)

Ausgangsrelationen:  
 $R_1 = \{InvNr, ISBN, Autor\}$   
 $F_1 = \{InvNr \rightarrow ISBN\}$   
 $\rightarrow K_{R_1} = \{InvNr, Autor\}$

$R_2 = \{ISBN, Titel\}$   
 $F_2 = \{ISBN \rightarrow Titel\}$   
 $\rightarrow K_{R_2} = \{ISBN\}$

Nicht 2NF:  $InvNr \rightarrow ISBN$   
**Zerlegung:**

InvNr	Autor
0007	Jepins Bone
1201	Heuer
1201	Scholl
4711	Vossen
4711	Witt
4712	Ullman
4717	Wirth
4717	Dijkstra

InvNr	ISBN
0007	3-125
1201	3-111
1201	3-111
4711	3-765
4711	3-765
4712	3-891
4717	3-999
4717	3-999

Neue Relationen:  
 $R_1' = \{InvNr, Autor\}$   
 $K_{R_1'} = \{InvNr, Autor\}$   
 $R_1'' = \{InvNr, ISBN\}$   
 $F_1'' = \{InvNr \rightarrow ISBN\}$   
 $K_{R_1''} = \{InvNr\}$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
 Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 43

### Boyce-Codd-Normalform

Einschränkung auf Nicht-Schlüsselattribute fallenlassen:  
 Im Postleitzahlensystem der Deutschen Post existieren innerhalb der Attribute *PLZ, Ort, Straße, Hausnummer* folgende funktionale Abhängigkeiten:  
  $Ort, Straße, Hausnummer \rightarrow PLZ$   
  $PLZ \rightarrow Ort$

Schlüssel:  
  $Ort, Straße, Hausnummer$   
  $PLZ, Straße, Hausnummer$

Alle Attribute nun Schlüsselattribute, also dritte Normalform  
 Trotzdem Redundanz:  $PLZ, Straße, Hausnummer \rightarrow PLZ \rightarrow Ort$   
 partielle (oder transitive) Abhängigkeit

**Boyce-Codd-Normalform (BCNF)** schließt auch transitive Abhängigkeiten über Schlüsselattribute aus

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
 Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 44

### Minimalität eines Datenbankschemas

**Idee:**

- Globale Redundanzen vermeiden
- andere Kriterien (wie Normalformen) mit möglichst wenig Schemata erreichen

**Beispiel:**

- Attributmenge  $ABC$
- FD-Menge  $\{A \rightarrow B, B \rightarrow C\}$

Datenbankschemata in dritter Normalform:

- $S = \{(AB, \{A\}), (BC, \{B\})\}$
- $S' = \{(AB, \{A\}), (BC, \{B\}), (AC, \{A\})\}$

$\rightarrow S'$  ist nicht minimal.

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
 Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 45

### Schema-Eigenschaften: Zusammenfassung

Schemaeigenschaft	Kurzcharakteristik
1 NF	Nur atomare Attribute
2 NF	keine partielle Abhängigkeit eines Nicht-Primattributes von einem Schlüssel
3 NF	keine transitive Abhängigkeit eines Nicht-Primattributes von einem Schlüssel
BCNF	keine transitive Abhängigkeit eines Attributes von einem Schlüssel
Minimalität	minimale Anzahl von Relationenschemata, die die anderen Eigenschaften erfüllt

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
 Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 46

### Transformationseigenschaften

Erreichen von Normalformen durch Zerlegung von Relationenschemata

Dabei beachten:

- Nur semantisch sinnvolle und konsistente Anwendungsdaten darstellen (**Abhängigkeitstreue**)
- Alle Anwendungsdaten sollen aus Basisrelationen hergeleitet werden können (**Verbundtreue**)

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
 Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 47

### FD-erhaltend zerlegen (=Abhängigkeitstreue)

Relationenschema  $\{Kurs, Professor, Zeit\}$  hat folgende minimale Menge von FDs:  
 $Kurs \rightarrow Professor$   
 $Professor, Zeit \rightarrow Kurs$

Kurs	Professor	Zeit
SWE1	Szwilius	Di 14-16
SWE1	?	Do 11-13

Redundante "Zerlegung"

Kurs	Professor	Zeit
SWE1	Szwilius	Di 14-16
SWE1	Szwilius	Do 11-13

Redundanzfreie Zerlegung

Kurs	Zeit
SWE1	Di 14-16
SWE1	Do 11-13

FD-erhaltend (=FD treu) =  
 Es gibt minimalen Menge F von FDs, so dass jede FD aus F innerhalb einer einzigen Relation, also ohne Join, testbar ist.

Nicht FD-erhaltend (=nicht FD treu) =  
 In jeder minimalen Menge F von FDs, gibt FD= (Professor, Zeit  $\rightarrow$  Kurs), deren Test folgenden Join benötigt: (Kurs, Professor) |X| (Kurs, Zeit)

Ist nicht beides gleichzeitig "FD-erhaltende Zerlegung" und "völlig redundanzfreie Zerlegung" möglich ist, wird "FD-erhaltende Zerlegung" bevorzugt.

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
 Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 48

### Abhängigkeitstreue

Allgemein: Menge der erfassten Abhängigkeiten äquivalent zur Menge der im System darstellbaren Abhängigkeiten (etwa Schlüssel und Fremdschlüssel)

Hier spezieller: Menge der FD's äquivalent zur Menge der Schlüsselabhängigkeiten

**Beispiel:**

- Attribute:  $PLZ (P), Ort (O), Straße (S), Hausnummer (H)$
- Funktionale Abhängigkeiten:  $F = \{ OSH \rightarrow P, P \rightarrow O \}$
- Datenbankschema:  $R = (OSHP, \{OSH\})$
- Menge der Schlüsselabhängigkeiten:  $K = \{ OSH \rightarrow OSHP \}$
- $K$  nicht äquivalent zu  $F$ , also ist das Schema  $R$  nicht abhängigkeittreu

### Abhängigkeitstreue formal

Sei  $S = \{ (R_1, \underline{K}_1), \dots, (R_p, \underline{K}_p) \}$  lokal erweitertes Datenbankschema,  $F$  Menge lokaler funktionaler Abhängigkeiten.

$S$  ist abhängigkeittreu bezüglich  $F$  genau dann, wenn

$$F \equiv \{ K \rightarrow R \mid (R, \underline{K}) \in S, K \in \underline{K} \}$$

Das heißt:  $F$  wird durch Schlüsselabhängigkeiten der gewählten Zerlegung äquivalent dargestellt.

### Verbundtreue (=verlustfreie Zerlegung)

**Idee:** Originalrelation soll aus zerlegten Relationen mit natürlichem Verbund zurückgewonnen werden können.

**Beispiel:**

- Relationenschema:  $R = ABC$
- Zerlegung in  $R_1 = AB$  und  $R_2 = BC$
- nicht verbundtreu bei:  $F = \{ A \rightarrow B, C \rightarrow B \}$
- verbundtreu bei:  $F' = \{ A \rightarrow B, B \rightarrow C \}$

**Definition:** Dekomposition einer Attributmenge  $X$  in  $X_1, \dots, X_p$  mit  $X = \bigcup_{i=1}^p X_i$  heißt *verbundtreu* bezüglich einer Menge von Abhängigkeiten  $G$  über  $X$  gdw.  $\forall$  Relationen  $r$ , die  $G$  erfüllen, gilt:  $\pi_{X_1}(r) \bowtie \dots \bowtie \pi_{X_p}(r) = r$

### (Gegen-) Beispiel zur Verbundtreue

Originalrelation:

A	B	C
1	2	3
4	2	5

$$F = \{ A \rightarrow B, C \rightarrow B \}$$

Dekomposition:

A	B	B	C
1	2	2	3
4	2	2	5

Verbund (nicht verbundtreu):

A	B	C
1	2	3
4	2	5
1	2	5
4	2	3

### Beispiel zur Verbundtreue

Originalrelation:

A	B	C
1	2	3
4	2	3

$$F' = \{ A \rightarrow B, B \rightarrow C \}$$

Dekomposition:

A	B	B	C
1	2	2	3
4	2	2	3

Verbund (verbundtreu):

A	B	C
1	2	3
4	2	3

### Kriterium für Verbundtreue

Attributmenge im Schnitt der entstandenen Relationenschemata (hier:  $B$ ) bestimmt eines der beiden Relationenschemata (hier:  $BC$ ) funktional (ist also Schlüssel)

**Formal:**

Eine Dekomposition von  $X$  in  $X_1$  und  $X_2$  ist verbundtreu bzgl.  $F$ , wenn

$$X_1 \cap X_2 \rightarrow X_1 \in F^+ \text{ oder } X_1 \cap X_2 \rightarrow X_2 \in F^+$$

### Verbundtreue formal

**Allgemeineres Kriterium:** Sei  $G$  Menge funktionaler Abhängigkeiten über  $X$ . Eine abhängigkeitstreue Dekomposition  $X_1, \dots, X_p$  von  $X$  ist verbundtreu bezüglich  $G$  wenn

$$\exists i \in \{1, \dots, p\} : X_i \rightarrow X \in G^+$$

Minimale Teilmenge von  $X_i$ : **Universalschlüssel**

Beispiel erster Fall: Da weder  $AB \rightarrow ABC$  noch  $BC \rightarrow ABC$  abgeleitet werden kann, ist das Kriterium nicht erfüllt und die Zerlegung nicht verbundtreu  
Beispiel zweiter Fall: Universalschlüssel  $A$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 55

### Transformationseigenschaften Zusammenfassung

Transformationseigenschaft	Kurzcharakteristik
Abhängigkeitstreue	Alle gegebenen Abhängigkeiten sind durch Schlüssel repräsentiert
Verbundtreue	Die Originalrelationen können durch den Verbund der Basisrelationen wiedergewonnen werden

Kein Join nötig, um FD zu testen

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 56

### Aufgabe: "Gutes Datenbankschema" entwerfen

**Gegeben:** Universum  $U$  und FD-Menge  $F$   
**Gesucht:** lokal erweitertes Datenbankschema  $S = \{(R_1, K_1), \dots, (R_p, K_p)\}$  so dass gilt

- Abhängigkeitstreue: Schlüsselabhängigkeiten in  $S$  sind äquivalent zu  $F$
- 3NF:  $S$  ist in 3. Normalform bezüglich  $F$
- Verbundtreue: Dekomposition von  $U$  in  $R_1, \dots, R_p$  ist verbundtreu bezüglich  $F$
- Minimalität: Für alle Zerlegungen  $S'$  von  $U$ , die abhängigkeittreu, in 3NF und verbundtreu sind, gilt  $|S'| \leq |S|$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 57

### Entwurfsaufgabe: Beispiel

Datenbankschemata schlecht entworfen, wenn nur eins dieser vier Kriterien nicht erfüllt ist

**Beispiel:**

- $S = \{(AB, \{A\}), (BC, \{B\}), (AC, \{A\})\}$  ist abhängigkeittreu, in 3NF und verbundtreu, bezüglich  $F = \{A \rightarrow B, B \rightarrow C, A \rightarrow C\}$
- in dritter Relation  $AC$ -Tupel redundant oder inkonsistent
- korrekt:  $S' = \{(AB, \{A\}), (BC, \{B\})\}$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 58

### Grenze guten Datenbankentwurfs: BCNF und Abhängigkeitstreue unvereinbar

**Beispiel:**

- Attribute:  $PLZ (P)$ , Ort ( $O$ ), Straße ( $S$ ), Hausnummer ( $H$ )
- Funktionale Abhängigkeiten:  $F = \{OSH \rightarrow P, P \rightarrow O\}$
- Datenbankschema:  $S = (OSHP, \{OSH, PSH\})$

**Analyse:**

- $PSH$  auch Schlüssel, da  $PSH \rightarrow OSHP$  mit  $PSH$  minimal
- Schema in 3NF, da alle Attribute Schlüsselattribute
- Schema nicht in BCNF, da  $\{PSH \supseteq P \rightarrow O\}$  partielle Abhängigkeit des Schlüsselattributs  $O$
- Jede Zerlegung von  $OSHP$  zerstört Abhängigkeit  $OSH \rightarrow P$

Abhängigkeitstreue nicht gewährleistet

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 59

### Dekompositionsverfahren: Aufgabe

**Gegeben:** Attributmenge  $R$  und FD-Menge  $F$   
**Zu bestimmen** (im Verlauf des Algorithmus):

- Hülle  $F^+$  durch RAP-Algorithmus
- Schlüsselmenge  $K(F)$  als Menge aller  $K \rightarrow R \in F^+$  mit minimalem  $K$ , d.h. so dass kein  $K' \rightarrow R$  existiert mit  $K' \subset K$  (d.h.  $K' \neq K$ )

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 60

### Schema-Synthese-Verfahren: Aufgabe

**Gegeben:** Attributmenge  $R$  und FD-Menge  $F$   
**Zu bestimmen** (im Verlauf des Algorithmus):

- Minimale FD-Menge  $MINF$  mit  $F^+ = MINF^+$
- Relationenmenge, so dass  $MINF$  erhaltend zerlegt wird

### Schema-Synthese-Verfahren: Algorithmus

**Bestimme eine minimale Menge funktionaler Abhängigkeiten  $MINF$  ;**  
 $i := 0$  ;  
**Für jede funktionale Abhängigkeit  $A_1, \dots, A_n \rightarrow A_x$  in  $MINF$**   
**(beginnend mit den Funktionalen Abhängigkeiten mit der längsten linken**  
**Seiten) tue**  
 { **if** (bisher enthält kein Schema  $R_i$  das Schema  $A_1, \dots, A_n, A_x$ )  
 {  $i := i+1$  ;  
 $R_i := A_1, \dots, A_n, A_x$  ;  
 }  
**if** (kein Schema enthält einen Schlüssel der universellen Relation  $R$ )  
 {  $i := i+1$  ;  
 $R_i :=$  irgendein Schlüssel der universellen Relation  $R$  ;  
 }  
**Das gesuchte Datenbankschema ist  $(R_1, \dots, R_i)$ .**

### Schema-Synthese-Verfahren: Beispiel

**Gegeben ist eine minimale Menge  $MINF$  funktionaler Abhängigkeiten:**

$MINF = \{ A \rightarrow B ; C \rightarrow D ; B, E \rightarrow A \}$

Die Schleife ergibt 2 Relationen

$R_1 = \{ B, E, A \}$   
 $R_2 = \{ C, D \}$

Es fehlt noch ein Schema,  
das einen Schlüssel der universellen Relation  $R$  enthält, z.B.

$R_3 = \{ A, C, E \}$

Ein mögliches Datenbankschema ist  $(R_1, R_2, R_3)$ .

### Dekomposition in BCNF

**gegeben:** universelles Schema  $R$  und minimale Menge  $F$  von FDs

```

generateBCNF(R, F)
{
  if R in BCNF output R
  else {
    select FD  $X \rightarrow Y \in F$  mit  $X \not\rightarrow R$  und  $X \cap Y = \emptyset$  ;
     $R_1 := R - Y$  ;
     $F_1 := \{fd \in F \mid \text{alle Attribute in fd sind in } R_1\}$  ;
     $R_2 := X \cup Y$  ;
     $F_2 := \{fd \in F \mid \text{alle Attribute in fd sind in } R_2\}$  ;
    generateBCNF( $R_1, F_1$ ) ;
    generateBCNF( $R_2, F_2$ ) ;
  }
}
    
```

Vorteile: BCNF, Verbundtreue

Nachteile: i.A. nicht abhängigkeittreu; abhängig von der Wahl ( $\rightarrow$  select);

### Dekompositionsverfahren: Normalisierung

```

normalize(R, F)
begin
  if R in 3NF
    output R,  $K(F)$ 
  else select Schlüssel  $K \in K(F)$  und FD  $Y \rightarrow A \in F^+$ 
    mit  $Y \not\rightarrow K$  und Attribüt  $A \notin K \cup Y$  ;
     $R_1 := R - \{A\}$  ;
     $F_1 := \{fd \in F^+ \mid \text{alle Attribute in fd sind in } R_1\}$  ;
     $R_2 := Y \cup A$  ;
     $F_2 := \{fd \in F^+ \mid \text{alle Attribute in fd sind in } R_2\}$  ;
    normalize( $R_1, F_1$ ) ;
    normalize( $R_2, F_2$ ) ;
end
    
```

garantiert noch: 3NF, Verbundtreue  
 Nachteile: i.A. nicht abhängigkeittreu und minimal; abhängig von der Wahl des Ansatzes ( $\rightarrow$  select); NP-vollständig (Schlüsselsuche)

### Beispiel (1)

Ausgangsrelation:

$R = \{InvNr, Titel, ISBN, Autor\}$   
 $F = \{InvNr \rightarrow ISBN, ISBN \rightarrow Titel\}$   
 $\rightarrow K(F) = \{\{InvNr, Autor\}\}$

Ansatz für  $R$ :

□  $InvNr, Autor \rightarrow ISBN \rightarrow Titel$ ;  
 nicht  $ISBN \rightarrow InvNr, Autor$  ;  
 $Titel \notin \{InvNr, Autor, ISBN\}$

Zerlegung:

□  $R_1 = \{InvNr, Autor, ISBN\}$  ;  
 $F_1 = \{InvNr \rightarrow ISBN\}$   
 $\rightarrow$  nicht 3NF  
 □  $R_2 = \{ISBN, Titel\}$  ;  
 $F_2 = \{ISBN \rightarrow Titel\}$   
 $\rightarrow$  3NF; output  $R_2, K(F_2) = \{\{ISBN\}\}$

### Beispiel (2)

Ansatz für R1:

- $InvNr, Autor \rightarrow InvNr \rightarrow ISBN;$   
nicht  $InvNr \rightarrow InvNr, Autor$   
 $ISBN \notin \{InvNr, Autor\}$

Zerlegung:

- $R11 = \{InvNr, ISBN\}$   
 $F11 = \{InvNr \rightarrow ISBN\}$   
→ 3NF; output R11,  $K(F11) = \{InvNr\}$
- $R12 = \{InvNr, Autor\}$   
 $F12 = \emptyset$   
→ 3NF; output R12,  $K(F12) = \{InvNr, Autor\}$

Analyse:

- abhängigkeitsrelation:  $K(F11) \cup K(F12) \cup K(F2) = F$
- verbundtreue: Eigenschaft des Algorithmus  
(hier R12 Schlüssel der Gesamterrelation)

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 67

### Mehrwertige Abhängigkeiten (Beispiel)

Mnr	Seminar	Hobby
1000	Electronic Commerce	Gitarre spielen
1000	Mobile Agenten	Gitarre spielen
1000	Electronic Commerce	Figurentheater
1000	Mobile Agenten	Figurentheater
1004	Electronic Commerce	Segeln

Die Tabelle enthält Redundanz durch mehrwertige Abhängigkeiten (MVDs):  
**1. Ein Student besucht (mehrere) Seminare und hat (mehrere) Hobbies**  
**2. a) Menge der Seminare hängt nur vom Student ab, nicht von Hobbies**  
**2. b) Menge der Hobbies hängt nur vom Student ab, nicht von Seminare.**

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 68

### Mehrwertige Abhängigkeiten (Beispiel)

Mehrwertige Abhängigkeit liegt vor, wenn für jeden Studenten (Mnr) gilt: Seminare sind nur abhängig von Mnr, aber unabhängig von Hobbies.

Mnr	Seminar	Hobby
1000	Electronic Commerce	Gitarre spielen
1000	Mobile Agenten	Gitarre spielen
1000	Electronic Commerce	Figurentheater
1000	Mobile Agenten	Figurentheater
1004	Electronic Commerce	Segeln

Für jeden Student mit Matrikelnr m gilt: Wenn R für ihn die Werte  $s_1, \dots, s_n$  im Attribut Seminar und die Werte  $h_1, \dots, h_m$  im Attribut Hobbies hat, dann sind alle Tupel  $\{m\} \times \{s_1, \dots, s_n\} \times \{h_1, \dots, h_m\}$ , also alle Kombinationen aus seinen Seminaren und seinen Hobbies, auch in der Relation enthalten.

Wir schreiben  $Mnr \twoheadrightarrow Seminar$  bzw.  $Mnr \twoheadrightarrow Hobby$  für die beiden mehrwertigen Abhängigkeiten (MVDs).

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 69

### Mehrwertige Abhängigkeiten (Definition)

$X_1, \dots, X_n \twoheadrightarrow Y_1, \dots, Y_m$

Mnr	Seminar	Hobby
1000	Electronic Commerce	Gitarre spielen
1000	Mobile Agenten	Gitarre spielen
1000	Electronic Commerce	Figurentheater
1000	Mobile Agenten	Figurentheater
1004	Electronic Commerce	Segeln

bedeutet folgendes:

Sei  $\{Z_1, \dots, Z_k\} = R - \{X_1, \dots, X_n\} - \{Y_1, \dots, Y_m\}$ .  
 Dann gilt für alle Paare  $t, u \in R$ :  
 $t(X_1, \dots, X_n) = u(X_1, \dots, X_n) \implies t(X_1, \dots, X_n) \text{ conc } u(Y_1, \dots, Y_m) \text{ conc } t(Z_1, \dots, Z_k) \in R$

also die Werte für die Y's von t kann man gegen die Werte für die Y's von u austauschen, und man erhält wieder ein Tupel aus R.

Es gilt:  $X_1, \dots, X_n \twoheadrightarrow Y_1, \dots, Y_m \iff X_1, \dots, X_n \twoheadrightarrow Z_1, \dots, Z_k$

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 70

### Verlustfreie Zerlegung bei mehrwertigen Abhängigkeiten

Jede Funktionale Abhängigkeit ist auch eine mehrwertige Abhängigkeit.  
 Relationen mit mehrwertigen Abhängigkeiten kann man verlustfrei zerlegen.

Im Beispiel:

Student_besucht_Seminar		Student_hat_Hobby	
Mnr	Seminar	Mnr	Hobby
1000	Electronic Commerce	1000	Gitarre spielen
1000	Mobile Agenten	1000	Figurentheater
1004	Electronic Commerce	1004	Segeln

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 71

### 4. Normalform

Eine Relation R heißt in 4. Normalform, wenn für jede mehrwertige Abhängigkeit (also auch für jede funktionale Abhängigkeit)  $A_1, \dots, A_n \twoheadrightarrow B_1, \dots, B_m$  zwischen Attributen aus R gilt:  
 $A_1, \dots, A_n$  ist ein Superkey von R oder die mehrwertige Abhängigkeit ist trivial (d.h.  $\{B_1, \dots, B_m\} \subseteq \{A_1, \dots, A_n\}$  oder  $\{B_1, \dots, B_m\} = R \setminus \{A_1, \dots, A_n\}$ )

Da jede funktionale Abhängigkeit auch eine mehrwertige Abhängigkeit ist, ist jede Relation in 4. Normalform auch in Boyce-Codd-Normalform.

Relationen in 4. Normalform schließen Redundanz durch MVDs aus.

Redundanz und Schemazerlegung    Funktionale Abhängigkeit    Schema-Eigenschaften    Transformationseigenschaften    Entwurfsaufgabe  
Datenbank Grundlagen    SS2005    Kapitel 9 - Datenbank-Schema-Entwurf / 72

### Dekomposition in 4.NF

gegeben: universelles Schema  $R$  und minimale Menge  $M$  von MVDs

```
generateBCNF(R, M)
{
  if R in BCNF output R
  else { select MVD  $X \twoheadrightarrow Y \in M$  mit  $X \not\rightarrow R$  und  $X \cap Y = \emptyset$  und  $Y \neq R-X$ ;
         $R1 := R - Y$ ;
         $R2 := X \cup Y$ ;
        generateBCNF(R1, M);
        generateBCNF(R2, M);
      }
}
```

Vorteile: a) 4. NF  $\implies$  BCNF  
b) Verbundtreue

Nachteile: i.A. nicht abhängigkeittreu; abhängig von der Wahl ( $\rightarrow$  select);  
Wie bei BCNF gilt auch hier: i.d.R. hat Abhängigkeitstreue Vorrang

### Ausblick: Normalisierung und Software-Qualität

Vermeidung von Redundanz beeinflusst (unmittelbar)

- interne Qualitäten:
  - bessere **Wartbarkeit**, **Wiederverwendbarkeit**, **Verständlichkeit** des Datenbank- (SQL-) Code, insbesondere beim Einfügen und Ändern von Tupeln
- externe Qualitäten:
  - bessere **Effizienz** durch geringeren Speicherplatzbedarf und Vermeidung von Update-Anomalien
  - höhere **Benutzerfreundlichkeit** bei ad-hoc-Benutzung
  - u.U. schlechtere **Effizienz** bei einzelnen Anfragen, da mehr Verbund-Operationen erforderlich sind

Transformationseigenschaften (Abhängigkeits- und Verbundtreue)

- sichern Korrektheit des relationalen Schemas bzgl. des konzeptionellen (ER) Schemas