

4. Relationaler Datenbankentwurf

- Funktionale Abhängigkeiten
- Schema-Eigenschaften
- Transformationseigenschaften
- Entwurfsverfahren

Relationaler Datenbankentwurf: Überblick

- Verfeinern des logischen Entwurfs
(wird später behandelt)
- **Ziel:** Vermeidung von Redundanzen durch Aufspalten von von Relationenschemata, ohne gleichzeitig
 - ◆ semantische Informationen zu verlieren (Abhängigkeitstreue)
 - ◆ die Möglichkeit zur Rekonstruktion der Relationen zu verlieren (Verbundtreue)
- Redundanzvermeidung durch *Normalformen*

4.1. Funktionale Abhängigkeiten

- Funktionale Abhängigkeit einer Relation zwischen Attributmengen X und Y , wenn in jedem Tupel der Relation der Attributwert unter den X -Komponenten den Attributwert unter den Y -Komponenten festlegt

- Funktionale Abhängigkeit (kurz: FD, von functional dependency)

Schreibweise: $X \rightarrow Y$

- Im Beispiel auf folgender Folie:
 $ISBN \rightarrow Titel, Verlag$
- Nicht: $ISBN \rightarrow Autor, Stichwort$
- Trivialerweise: $ISBN \rightarrow ISBN$

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.

[2 Autoren, 2 Versionen, 3 Stichwörter \rightsquigarrow 12 Kombinationen !]

Schlüssel als Spezialfall

- für Beispiel auf folgender Folie
 $PANr \rightarrow Vorname, Nachname, PLZ, Ort, Straße, Hausnummer, Geburtsdatum$
- Immer: $PANr \rightarrow PANr$
(zusammen dann gesamtes Schema auf rechter Seite)
- Wenn linke Seite minimal: Schlüssel
- **Formal:**
Schlüssel X liegt vor, wenn FD $X \rightarrow R$ für Relationenschema R gilt und X minimal

Ziel des Datenbankentwurfs:
alle gegebenen funktionalen Abhängigkeiten in „Schlüsselabhängigkeiten“ umformen, ohne dabei semantische Information zu verlieren

Schlüssel im Beispiel

Personen	PANr	Vorname	Nachname	PLZ	Ort	Straße	HINr	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	Loesser	69121	HD	TS	38	10.05.1969

Pers._Telefon	Telefon
4711	038203-12230
4711	0381-498-3401
4711	0381-498-3427
5588	0391-345677
5588	0391-5592-3800
9999	06221-400177

Ableitung von FDs

$$r \begin{array}{|c|c|c|} \hline A & B & C \\ \hline a_1 & b_1 & c_1 \\ a_2 & b_1 & c_1 \\ a_3 & b_2 & c_1 \\ a_4 & b_1 & c_1 \\ \hline \end{array}$$

- genügt $A \rightarrow B$ und $B \rightarrow C$
- dann gilt auch $A \rightarrow C$ (\sim Transitivität)
- nicht ableitbar $C \rightarrow A$ oder $C \rightarrow B$
- sei f eine FD und F eine Menge von FDs über R :
wenn alle Ausprägungen zu R , die alle FDs aus F erfüllen, auch f erfüllen, dann *impliziert* F die FD f (kurz: $F \models f$)
- obiges Beispiel:

$$\{A \rightarrow B, B \rightarrow C\} \models A \rightarrow C$$

Menge aller von F implizierten funktionalen Abhängigkeiten (Hülle F^+ zu F):

$$F^+ := \{ f \mid F \models f \}$$

Ableitungsregeln

- *gültig* (sound)
- *vollständig* (complete)
- *unabhängig* (independent) oder auch bzgl. \subseteq minimal

Eine solche Menge von Ableitungsregeln:

Name	Regel
Reflexivität	$\{ \} \implies X \rightarrow X$
Akkumulation	$\{ X \rightarrow YZ, Z \rightarrow VW \} \implies X \rightarrow YZV$
Projektivität	$\{ X \rightarrow YZ \} \implies X \rightarrow Y$

Abkürzungen:

- R** für Reflexivität
- A** für Akkumulation
- P** für Projektivität

Membership-Problem

Kann eine bestimmte FD $X \rightarrow Y$ aus der vorgegebenen Menge F abgeleitet werden, d.h. wird sie von F impliziert?

Membership-Problem: „ $X \rightarrow Y \in F^+ ?$ “

Hülle einer Attributmenge X bzgl. F ist

$$X_F^* := \{ A \mid X \rightarrow A \in F^+ \}$$

Das Membership-Problem kann nun durch das modifizierte Problem

Membership-Problem (2): „ $Y \subseteq X_F^* ?$ “

in linearer Zeit gelöst werden.

RAP-Algorithmus

1. Bestimme X , setze $X^* := X$ (\boxed{R} -Regel für X)
2. Gibt es FD $f_1 := X_1 \rightarrow Y_1 \in F$ mit $X_1 \subseteq X^*$?
3. Wenn ja, dann wird X^* gemäß $X^* := X^* \cup Y_1$ vergrößert (\boxed{A} -Regel).
4. Führe Schritt 2 und 3 so lange aus, bis X^* stabil (Hülle)
5. Ist $Y \subseteq X^*$, dann ist $X \rightarrow Y \in F^+$ (\boxed{P} -Regel)

Überdeckungen:

F heißt *äquivalent* zu G falls $F^+ = G^+$

(oder: F *Überdeckung* von G ; kurz: $F \equiv G$)

4.2. Schema-Eigenschaften

Relationenschemata, Schlüssel und Fremdschlüssel so wählen, daß

1. alle Anwendungsdaten aus den Basisrelationen hergeleitet werden können,
2. nur semantisch sinnvolle und konsistente Anwendungsdaten dargestellt werden können und
3. die Anwendungsdaten möglichst nicht-redundant dargestellt werden.

Hier: Forderung 3

- Redundanzen innerhalb einer Relation: Normalformen
- globale Redundanzen: Minimalität

Update-Anomalien

Redundanzen in Basisrelationen unerwünscht:

- belegen unnötigen Speicherplatz (eher unwichtig)
- Information redundant
 - ↪ Änderung muß diese Information in allen ihren Vorkommen verändern (in relationalen Systemen nur schwer zu realisieren)

Beispiel **insert**-Anomalie:

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

in Bücher-Relation einfügen:

- eine weitere Version
- ein neuer Verlag

↪ Wieviele Tupel fehlen zur „Vervollständigung“?

[in diesem Beispiel liegt eine mehrwertige Abhängigkeit vor, d.h. zu einem Buch gibt es eine Menge von Autoren, eine Menge von Versionen, eine Menge von Stichwörtern, ...]

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

Zweite Normalform

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

■ Zweite Normalform:

Keine partiellen Abhängigkeiten zwischen einem Schlüssel und weiteren Nicht-Primattributen

[Primattribut: Attribut, das zu (irgendeinem) Schlüssel gehört]

■ *partielle Abhängigkeit* liegt vor, wenn ein oder mehrere Attribute funktional schon von einem *Teil* des Schlüssels abhängen

■ Beispiel:

$Invnr \rightarrow Titel$

und

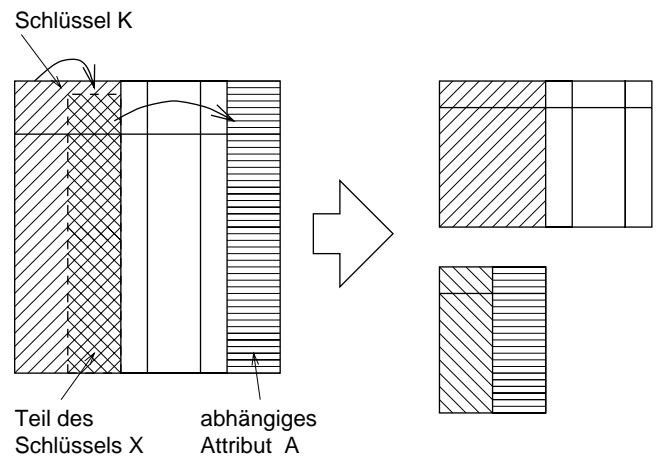
$Invnr, Autor \rightarrow Invnr, Titel, ISBN, Autor$

$\leadsto Invnr$ und $Autor$ zusammen Schlüssel

$Titel$ hängt aber allein von $Invnr$ ab

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

Veranschaulichung zweite Normalform



Dritte Normalform

■ Dritte Normalform:

Keine transitiven Abhängigkeiten zwischen einem Schlüssel und weiteren Nicht-Primattributen

■ *transitive Abhängigkeit*: Schlüssel K bestimmt Attributmenge X funktional, diese aber auch eine Attributmenge Y

transitive Abhängigkeit $K \rightarrow X \rightarrow Y$

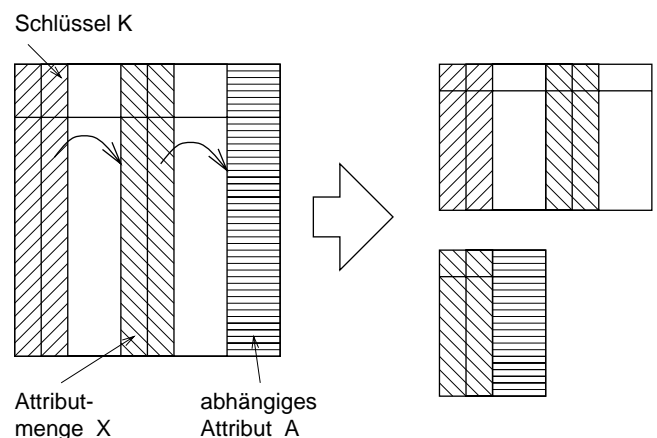
■ Beispiel: $PANr \rightarrow PLZ$ und $PLZ \rightarrow Ort$

Information, daß zur PLZ '18209' der Ort 'DBR' gehört, ist redundant

■ Dritte Normalform erreichen durch Zerlegung in zwei Relationenschemata (siehe nächste Folie):

- ◆ ursprüngliches Relationenschema ohne Y
- ◆ neues Relationenschema mit X und Y

Veranschaulichung dritte Normalform



Boyce-Codd-Normalform

- nicht nur Nicht-Primattribute betrachten
- Beispiel:
Im aktuellen Postleitzahlensystem der Deutschen Post innerhalb der Attribute

PLZ, Ort, Straße, Hausnummer

folgende funktionalen Abhängigkeiten:

Ort, Straße, Hausnummer \rightarrow PLZ,
PLZ \rightarrow Ort

Schlüssel: Ort, Straße, Hausnummer
und PLZ, Straße, Hausnummer

alle Attribute nun Primattribute:

\leadsto damit automatisch in dritter Normalform!

- trotzdem Redundanz:

PLZ, Straße, Hausnummer \rightarrow PLZ \rightarrow Ort

- partielle (oder transitive) Abhängigkeit

Boyce-Codd-Normalform (BCNF) definiert transitive Abhängigkeiten nicht nur über Nicht-Primattribute

Minimalität

- Global Redundanzen vermeiden
- andere Kriterien (wie Normalformen) mit möglichst wenigen Schemata erreichen

- Beispiel:

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

- Datenbankschemata in dritter Normalform:

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

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

Redundanzen in S'

Schema-Eigenschaften: Zusammenfassung

Kennung	Schema-eigenschaft	Kurzcharakteristik
	1NF	nur atomare Attribute
	2NF	keine partielle Abhängigkeit eines Nicht-Primattributes von einem Schlüssel
S1	3NF	keine transitive Abhängigkeit eines Nicht-Primattributes von einem Schlüssel
	BCNF	keine transitive Abhängigkeit eines Attributes von einem Schlüssel
S2	Minimalität	minimale Anzahl von Relationenschemata, die die anderen Eigenschaften erfüllt

4.3. Transformationseigenschaften

- Erreichen von Normalformen durch Zerlegung von Relationenschemata

- Dabei beachten:

1. nur semantisch sinnvolle und konsistente Anwendungsdaten darstellen
 \leadsto *Abhängigkeitstreue*
2. alle Anwendungsdaten sollen aus Basisrelationen hergeleitet werden können
 \leadsto *Verbundtreue*

Abhängigkeitstreue

- Abhängigkeitstreue ist folgende Forderung:
 - ◆ Allgemein: Menge der erfaßten Abhängigkeiten äquivalent zur Menge der im System darstellbaren Abhängigkeiten (etwa Schlüssel und Fremdschlüssel)
 - ◆ Hier spezieller: Menge der FDs 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 S : (OSHP, {OSH})
 - Menge der zugehörigen Schlüsselabhängigkeiten: { OSH \rightarrow OSHP }
- nicht äquivalent zu F ; S ist nicht *abhängigkeitstreu*

Verbundtreue

- Originalrelation soll aus zerlegten Relationen mit natürlichem Verbund zurückgewonnen werden können
- Beispiel: Relationenschema $R = ABC$ in $R_1 = AB$ und $R_2 = BC$ zerlegt; ist bei

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

nicht verbundtreu, bei

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

verbundtreu.

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

Beispielrelationen zur Verbundtreue

- Originalrelation:

A	B	C
1	2	3
4	2	5
- Dekomposition:

A	B	C
1	2	2
4	2	2
- Verbund (nicht verbundtreu):

A	B	C
1	2	3
4	2	5

- Originalrelation:

A	B	C
1	2	3
4	2	3
- Dekomposition:

A	B	C
1	2	2
4	2	2
- Verbund (verbundtreu):

A	B	C
1	2	3
4	2	3

Transformationseigenschaften: Zusammenfassung

Kennung	Transformations-eigenschaft	Kurzcharakteristik
T1	Abhängigkeitstreue	alle gegebenen Abhängigkeiten sind durch Schlüssel repräsentiert
T2	Verbundtreue	die Originalrelationen können durch den Verbund der Basisrelationen wiedergewonnen werden

4.4. Entwurfsverfahren

- Universum \mathcal{U} (Menge aller Attribute) und FD-Menge F gegeben
- lokal erweitertes Datenbankschema $S = \{(R_1, \mathcal{K}_1), \dots, (R_p, \mathcal{K}_p)\}$ berechnen mit
 - T1 – S charakterisiert vollständig F
 - S1 – S ist in 3NF bezüglich F
 - T2 – Dekomposition von \mathcal{U} in R_1, \dots, R_p ist verbundtreu bezüglich F
 - S2 – Minimalität, d.h. $\nexists S' : S'$ erfüllt T1, S1, T2 und $|S'| < |S|$

Entwurfsverfahren II

- Datenbankschemata schlecht entworfen, wenn nur eins dieser vier Kriterien nicht erfüllt
- Beispiel:

$$S = \{(AB, \{A\}), (BC, \{B\}), (AC, \{A\})\}$$
 erfüllt T1, S1 und T2 bezüglich

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

aber:
in dritter Relation AC -Tupel redundant oder inkonsistent
- korrekt: $S' = \{(AB, \{A\}), (BC, \{B\})\}$

BCNF und Abhängigkeitstreue unvereinbar

- Attribute:
PLZ (P), Ort (O), Straße(S), Hausnummer(H)
- funktionale Abhängigkeiten F

$$OSH \rightarrow P, P \rightarrow O$$
- Datenbankschema S

$$(OSH, \{OSH, PSH\})$$
- PSH auch Schlüssel, da $PSH \rightarrow OSHP$ mit PSH minimal
- Schema in 3NF, da alle Attribute Primattribute

BCNF und Abhängigkeitstreue unvereinbar II

- Schema nicht in BCNF, da

$$\{ PSH \rightarrow P \rightarrow O \}$$
 transitive Abhängigkeit des Primattributs O
- Jede Zerlegung von OSHP zerstört Abhängigkeit

$$OSH \rightarrow P$$
- Abhängigkeitstreue nicht gewährleistet

Dekomposition: Start

Start: initiales Relationenschema R mit allen Attributen und einer von erfaßten Abhängigkeiten implizierten Schlüsselmenge

- Attributmenge \mathcal{U} und eine FD-Menge F
- suche alle $K \rightarrow \mathcal{U}$ mit K minimal, für die $K \rightarrow \mathcal{U} \in F^+$ gilt ($\mathcal{K}(F)$)
- $(\mathcal{U}, \mathcal{K}(F))$ initiales Relationenschema

Dekomposition: Normalisierung

Normalisierungsschritt: falls $K \rightarrow X \rightarrow Y$, aus R Attributmenge Y eliminieren und mit X in ein neues Relationenschema stecken

- $\mathcal{R} = (R, \mathcal{K})$ und F über R gegeben
- Falls \mathcal{R} in 3NF ist: fertig
- Sonst: für Schlüssel K existiert

$$K \rightarrow X, X \not\rightarrow K, X \rightarrow Y, Y \cap KX = \emptyset$$

Wähle dann

$$\begin{aligned} R_1 &:= R - Y & R_2 &:= XY \\ \mathcal{R}_1 &:= (R_1, \mathcal{K}) & \mathcal{R}_2 &:= (R_2, \mathcal{K}_2 = \{X\}) \end{aligned}$$

Bewertung des Dekompositionsverfahrens:

- Vorteile: 3NF, Verbundtreue
- Nachteile: restliche Kriterien nicht, reihenfolgeabhängig, NP-vollständig (Schlüsselsuche)