6. Relationaler Datenbankentwurf

Einordnung in den Vorlesungsverlauf

- ER-Modell
- Relationenmodell
- relationale Anfragesprachen
- SQL
- **Entwurfstheorie**
- Transaktionen

Überblick

- funktionale Abhängigkeiten
- Schema-Eigenschaften
- Transformationseigenschaften
- Entwurfsverfahren
- weitere Abhängigkeiten

Relationaler DB-Entwurf: Ziele

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

Sei R ein Relationenschema und $X, Y \subseteq R$ zwei Attributmengen.

In einer Relation besteht eine **funktionale Abhängigkeit** zwischen zwei Attributmengen X und Y, wenn in jedem Tupel der Relation der Attributwert unter den X-Komponenten den Attributwert unter den Y-Komponenten festlegt.

Schreibweise: $X \to Y$

Funktionale Abhängigkeit kurz: FD (von functional dependency)

FD formal

$$X \to Y :\Longleftrightarrow \forall r \in \textbf{REL}(R) \text{ gilt: } \forall t_1, t_2 \in r(R) : t_1(X) = t_2(X) \implies t_1(Y) = t_2(Y)$$

wobei $t(X)$ die Einschränkung eines Tupels t auf die Attribute in X ist.

Bücher-Relation mit Redundanzen

$ISBN \rightarrow Titel$
$ISBN \rightarrow Verlag$

zusammengefasst:
$ISBN \rightarrow Titel\ Verlag$

trivialerweise:
$ISBN \rightarrow ISBN$

gilt nicht:
$ISBN \rightarrow Autor$
$ISBN \rightarrow Stichwort$

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.

Schlüssel

Sei R ein Relationenschema und $X, \beta \subseteq R$ zwei Attributmengen.

Eine Attributmenge $X \subseteq R$ heißt **Superschlüssel**, wenn $X \rightarrow R$.

Schlüsseleigenschaft: Alle Attribute von R hängen funktional von X ab.

Eine Attributmenge $X \subseteq R$ heißt **Kandidatenschlüssel**, wenn

- X erfüllt die Schlüsseleigenschaft
- X ist *minimal*

minimal: Kein Attribut kann aus X entfernt werden, ohne die Schlüsseleigenschaft zu verletzen. Für alle Attribute $A \in X$ gilt: $X \setminus A \nrightarrow R$.

Primattribut: Ein Attribut heißt prim, falls es Teil von irgendeinem Kandidatenschlüssel ist.

Schlüssel

- für Beispiel auf folgender Folie

$$\begin{aligned} \mathit{PANr} \;\rightarrow\; & \mathit{Vorname},\ \mathit{Nachname},\ \mathit{PLZ},\ \mathit{Ort}, \\ & \mathit{Strasse},\ \mathit{Hausnummer},\ \mathit{Geburtsdatum} \end{aligned}$$

- es gilt immer: $\mathit{PANr} \rightarrow \mathit{PANr}$,
  damit gesamtes Schema auf rechter Seite
- da linke Seite minimal: *PANr* ist Kandidatenschlüssel

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	GebDatum
4711	Andreas	Heuer	18209	DBR	31.10.1958
5588	Gunter	Saake	39106	MD	05.10.1960
6834	Michael	Korn	39104	MD	24.09.1974
7754	Andreas	Möller	18209	DBR	25.02.1976
8832	Tamara	Jagellovsk	38106	BS	11.11.1973
9912	Antje	Hellhof	18059	HRO	04.04.1970
9999	Christa	Loeser	69121	HD	10.05.1969

Pers_Telefon

PANr	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 i

Notation

Ab jetzt schreiben wir AB anstatt $\{A, B\}$, also z.B. $AB \rightarrow BCD$ anstatt $\{A, B\} \rightarrow \{B, C, D\}$

R	A	B	C
	a_1	b_1	c_1
	a_2	b_1	c_1
	a_3	b_2	c_1
	a_4	b_1	c_1

- genügt $A \rightarrow B$ und $B \rightarrow C$
- dann gilt auch $A \rightarrow C$
- nicht ableitbar $C \rightarrow A$ oder $C \rightarrow B$

Ableitung von FDs ii

- Gilt für f über R $\mathbf{SAT}_R(F) \subseteq \mathbf{SAT}_R(f)$, dann impliziert F die FD f (kurz: $F \models f$)
- obiges Beispiel:

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

Hüllenbildung F^+:

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

$\mathbf{SAT}_R(F)$ ist die Menge der Relationen mit Schema R, die alle FDs in F erfüllen.

Ableitungsregeln

Anforderungen an Ableitungssysteme:

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

Name	Regel		
$\boxed{\text{R}}$ Reflexivität	$\{\}$	$\Longrightarrow$	$X \rightarrow X$
$\boxed{\text{A}}$ Akkumulation	$\{X \rightarrow YZ, Z \rightarrow VW\}$	$\Longrightarrow$	$X \rightarrow YZV$
$\boxed{\text{P}}$ Projektivität	$\{X \rightarrow YZ\}$	$\Longrightarrow$	$X \rightarrow Y$

Weitere Ableitungsregeln i

R_1	Reflexivität:	$X \supseteq Y$	$\Longrightarrow$	$X \rightarrow Y$
R_2	Augmentation:	$\{X \rightarrow Y\}$	$\Longrightarrow$	$XZ \rightarrow YZ$
R_3	Transitivität:	$\{X \rightarrow Y, Y \rightarrow Z\}$	$\Longrightarrow$	$X \rightarrow Z$
R_4	Dekomposition:	$\{X \rightarrow YZ\}$	$\Longrightarrow$	$X \rightarrow Y$
R_5	Vereinigung:	$\{X \rightarrow Y, X \rightarrow Z\}$	$\Longrightarrow$	$X \rightarrow YZ$
R_6	Pseudotransitivität:	$\{X \rightarrow Y, WY \rightarrow Z\}$	$\Longrightarrow$	$WX \rightarrow Z$

R_1-R_3 bekannt als Armstrong-Axiome (sound, complete & independent)

Beweis: Reflexivität

- Annahme:
  $X \supseteq Y, X, Y \subset R, t_1, t_2 \in r(R)$ mit $t_1(X) = t_2(X)$
- dann folgt: $t_1(Y) = t_2(Y)$ wegen $X \supseteq Y$
- daraus folgt: $X \rightarrow Y$

Weitere Ableitungsregeln iii

Beweis: Agumentation

- Annahme: $X \rightarrow Y$ gilt in $r(R)$, jedoch nicht: $XZ \rightarrow YZ$
- dann müssen zwei Tupel $t_1, t_2 \in r(R)$ existieren, so dass gilt
  (1) $t_1(X) = t_2(X)$
  (2) $t_1(Y) = t_2(Y)$
  (3) $t_1(XZ) = t_2(XZ)$
  (4) $t_1(YZ) \neq t_2(YZ)$
- Widerspruch wegen $t_1(Z) = t_2(Z)$ aus (1) und (3), woraus mit (2) folgt: $t_1(YZ) = t_2(YZ)$

Weitere Ableitungsregeln iv

Beweis: Transitivität

- Annahme: in $r(R)$ gelten:
  (1) $X \rightarrow Y$
  (2) $Y \rightarrow Z$
- demzufolge für zwei beliebige Tupel $t_1, t_2 \in r(R)$ mit $t_1(X) = t_2(X)$ muss gelten:
  (3) $t_1(Y) = t_2(Y)$ (wegen (1))
  (4) $t_1(Z) = t_2(Z)$ (wegen (3) und (2))
- daher gilt: $X \rightarrow Z$

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

Gegeben Attributmenge X. Gilt $X \rightarrow Y \in F^+$?

1. Berechnung der Hülle X^* bzgl. FD-Menge F

   $X^0 := X$ ($\boxed{\text{R}}$-Regel)
   $X^{i+1} = X^i \cup \{A \mid \exists Y \rightarrow Z \in F \text{ mit } Y \subseteq X^i \text{ und } A \in Z\}$ ($\boxed{\text{A}}$-Regel)
   Wenn $X^{i+1} = X^i$, dann ist X^* erreicht

2. Prüfen, ob $Y \subseteq X*$ gilt oder nicht.

   Ist $Y \subseteq X^*$, dann gilt $X \rightarrow Y \in F^+$ ($\boxed{\text{P}}$-Regel)

Überdeckung und Äquivalenz

Gegeben zwei FD-Mengen F und G zum gleichen Relationsschema R.

- F heißt **Überdeckung** zu G, wenn gilt $G^+ \subseteq F^+$
  - Jede FD in G ist in der Hülle von F enthalten
    Für alle $X \rightarrow Y \in G$ gilt: $Y \subseteq X_F^*$
- F ist **äquivalent** zu G, wenn gilt $G^+ = F^+$
  - kurz: $F \equiv G$
  - $G^+ = F^+ \Leftrightarrow G^+ \subseteq F^+ \wedge F^+ \subseteq G^+$

6.2. 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 Speicherplatzes (eher unwichtig)
  - Information redundant $\longrightarrow$ 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 (FD, MVD verletzt; besser: auf Schlüsselabhängigkeiten zurückführen)

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

  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

Zweite Normalform i

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

- Zweite Normalform (2NF):
  1. Relation ist in 1NF
  2. Es gibt keine partielle Abhängigkeit zwischen einem Kandidatenschlüssel und einem Nicht-Primattribut
- partielle Abhängigkeit liegt vor, wenn ein Attribut funktional schon von einem *Teil* eines Kandidatenschlüssels abhängt
- *R* mit FD-Menge *F* ist in 2NF, gdw. für alle $\alpha \to \beta \in F$ mindestens eine der folgenden Bedingungen gilt:
  - $\alpha \to \beta$ ist trivial oder
  - alle Attribute in β sind prim oder
  - α ist keine echte Teilmenge eines Kandidatenschlüssels

Zweite Normalform ii

- Beispiel:

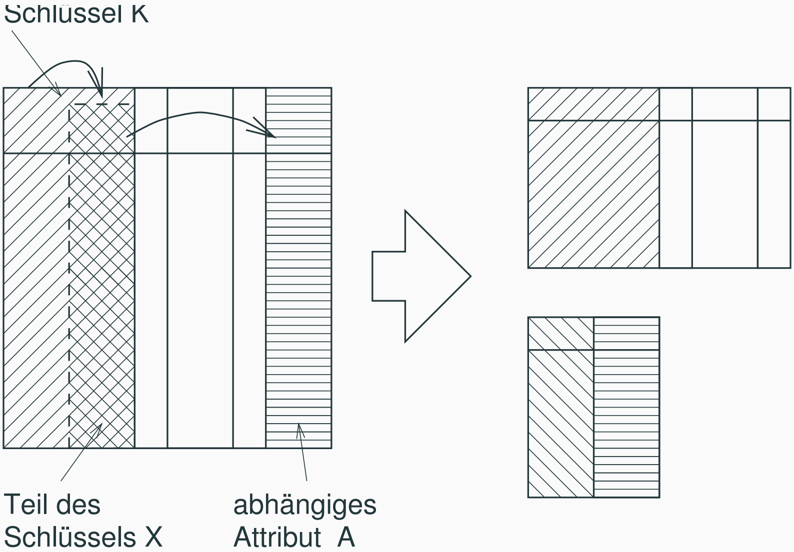
$$Invnr \rightarrow Titel$$

  und

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

  $Invnr$ und $Autor$ zusammen Schlüssel
  $Titel$ hängt aber allein von $Invnr$ ab
- 2NF erreichen durch Elimination der rechten Seite der partiellen Abhängigkeit und Kopie der linken Seite (siehe nächste Folie)

Veranschaulichung zweite Normalform



Dritte Normalform

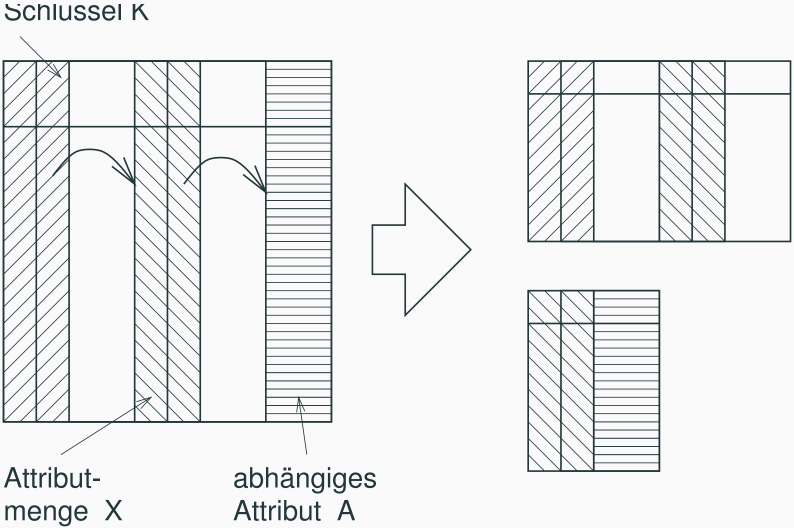
- Dritte Normalform (3NF):
  1. Relation ist in 2NF
  2. Es gibt keine transitive Abhängigkeit zwischen einem Kandidatenschlüssel und einem Nicht-Primattribut
- Es gibt eine transitive Abhängigkeit zwischen K und Y, wenn eine Attributmenge X existiert, sodass $K \rightarrow X$ und $X \rightarrow Y$ (also $K \rightarrow X \rightarrow Y$)
- R mit FD-Menge F ist in 3NF, gdw. für alle $\alpha \rightarrow \beta \in F$ mindestens eine der folgenden Bedingungen gilt:
  - $\beta \subseteq \alpha$ oder
  - alle Attribute in β sind prim oder
  - α ist ein Superschlüssel

Dritte Normalform

- Beispiel transitive Abhängigkeit:
  $PANr \rightarrow PLZ$ und $PLZ \rightarrow Ort$

  Information, dass zur PLZ ‘40225’ der Ort ‘Duesseldorf’ gehört, ist redundant
- 3NF erreichen durch Elimination von Y und Kopie von X (siehe nächste Folie)

Veranschaulichung dritte Normalform



Boyce-Codd-Normalform i

- nicht nur Nicht-Primattribute betrachten
- im aktuellen Postleitzahlsystem der Deutschen Post innerhalb der Attribute

$$\mathtt{PLZ},\ \mathtt{Ort},\ \mathtt{Strasse},\ \mathtt{Hausnummer}$$

folgende funktionale Abhängigkeiten:

$$\mathtt{Ort},\ \mathtt{Strasse},\ \mathtt{Hausnummer} \rightarrow \mathtt{PLZ},$$
$$\mathtt{PLZ} \rightarrow \mathtt{Ort}$$

Schlüssel:
$\mathtt{Ort},\ \mathtt{Strasse},\ \mathtt{Hausnummer}$
und $\mathtt{PLZ},\ \mathtt{Strasse},\ \mathtt{Hausnummer}$

alle Attribute nun Primattribute: $\rightsquigarrow$ 3NF

Boyce-Codd-Normalform ii

- trotzdem Redundanz:

$$\mathtt{PLZ,\ Strasse,\ Hausnummer} \rightarrow \mathtt{PLZ} \rightarrow \mathtt{Ort}$$

- partielle (oder transitive) Abhängigkeit

- Boyce-Codd-Normalform (BCNF):
  1. Relation ist in 3NF
  2. Es gibt kein *primes* Attribut mit einer partiellen/transitiven Abhängigkeit

- R mit FD-Menge F ist in BCNF, gdw. für alle $\alpha \rightarrow \beta \in F$ mindestens eine der folgenden Bedingungen gilt:
  - $\beta \subseteq \alpha$ oder
  - α ist ein Superschlüssel

Minimalität

- global Redundanzen vermeiden
- andere Kriterien (wie Normalformen) mit möglichst wenig Schemata erreichen
- Beispiel:
  Attributmenge ABC, FD-Menge $\{A \rightarrow B, B \rightarrow C\}$
- Datenbankschemata in 3NF:

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

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

Redundanzen in S'

Schema-Eigenschaften

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

6.3. Transformationseigenschaften

- Erreichen von Normalformen durch Zerlegung von Relationenschemata
- dabei beachten:
  1. nur semantisch sinnvolle und konsistente Anwendungsdaten darstellen (Abhängigkeitstreue)
  2. alle Anwendungsdaten sollen aus Basisrelationen hergeleitet werden können (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

Abhängigkeitstreue: Beispiel

- Attribute:
  $\mathtt{PLZ\ (P)}, \mathtt{Ort\ (O)}, \mathtt{Strasse\ (S)}, \mathtt{Hausnummer\ (H)}$
- 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*

Abhängigkeitstreue formal

- $S = \{(R_1, \mathcal{K}_1), \ldots, (R_p, \mathcal{K}_p)\}$ lokal erweitertes Datenbankschema, F Menge lokaler Abhängigkeiten
- S charakterisiert vollständig F (oder: ist abhängigkeitstreu bezüglich F) genau dann, wenn

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

Verbundtreue i

- 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.

Verbundtreue ii

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

Beispielrelationen zur Verbundtreue i

1. Originalrelation:

A	B	C
1	2	3
4	2	5

   Dekomposition:

A	B
1	2
4	2

B	C
2	3
2	5

   Verbund (nicht verbundtreu):

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

Beispielrelationen zur Verbundtreue ii

2. Originalrelation:

A	B	C
1	2	3
4	2	3

Dekomposition:

A	B
1	2
4	2

B	C
2	3

Verbund (verbundtreu):

A	B	C
1	2	3
4	2	3

Verbundtreue formal i

- *Definition:*
  Dekomposition einer Attributmenge X in $X_1, \ldots, X_p$ mit $X = \bigcup_{i=1}^{p} X_i$ heißt verbundtreu ($\pi \bowtie$-treu, lossless) bezüglich einer Menge von Abhängigkeiten G über X genau dann, wenn

$$\forall r \in \mathbf{SAT}_X(G) : \pi_{X_1}(r) \bowtie \cdots \bowtie \pi_{X_p}(r) = r$$

  gilt
- *einfaches Kriterium für zwei Relationenschemata:*
  Dekomposition von X in X_1 und X_2 ist verbundtreu bzgl. F, wenn $X_1 \cap X_2 \rightarrow X_1 \in F^+$ oder $X_1 \cap X_2 \rightarrow X_2 \in F^+$

- *allgemeineres Kriterium:*
  G Menge funktionaler Abhängigkeiten

$$\exists i \in \{1, \ldots, p\} : X_i \rightarrow X \in G^+ \quad \Longrightarrow$$
$$\text{Dekomposition von } X \text{ in } X_1, \ldots, X_p$$
$$\text{ist verbundtreu bezüglich } G$$

- minimale Teilmenge von X_i: Universalschlüssel
- Beispiel erster Fall: AC der einzige Universalschlüssel, in keinem Relationenschema enthalten
- Beispiel zweiter Fall: Universalschlüssel A

Transformationseigenschaften

Kennung	Transformations-eigenschaft	Kurzcharakteristik
$\boxed{T}\boxed{1}$	Abhängigkeits-treue	alle gegebenen Abhängigkeiten sind durch Kandidatenschlüssel repräsentiert
$\boxed{T}\boxed{2}$	Verbundtreue	die Originalrelationen können duch den Verbund der Basisrelationen wiedergewonnen werden

Entwurfsverfahren i

- Universum $\mathcal{U}$ und FD-Menge F gegeben
- lokal erweitertes Datenbankschema $S = \{(R_1, \mathcal{K}_1), \ldots, (R_p, \mathcal{K}_p)\}$ berechnen mit

  T 1 S charakterisiert vollständig F

  S 1 S ist in 3NF bezüglich F

  T 2 Dekomposition von $\mathcal{U}$ in $R_1, \ldots, R_p$ ist verbundtreu bezüglich F

  S 2 Minimalität, d.h.

  $\nexists S' : S'$ erfüllt $\boxed{\text{T}}\boxed{1}$, $\boxed{\text{S}}\boxed{1}$, $\boxed{\text{T}}\boxed{2}$ und $|S'| < |S|$

Entwurfsverfahren ii

- Datenbankschemata schlecht entworfen, wenn nur eines dieser vier Kriterien nicht erfüllt
- Beispiel:
  $S = \{(AB, \{A\}), (BC, \{B\}), (AC, \{A\})\}$ erfüllt $\boxed{\text{T}}\boxed{1}$, $\boxed{\text{S}}\boxed{1}$ und $\boxed{\text{T}}\boxed{2}$ bezüglich
  $F = \{A \rightarrow B, B \rightarrow C, A \rightarrow C\}$
  in dritter Relation AC-Tupel redundant oder inkonsistent
- korrekt: $S' = \{(AB, \{A\}), (BC, \{B\})\}$

BCNF und Abhängigkeitstreue i

BCNF und Abhängigkeitstreue unvereinbar

- Attribute $\mathtt{PLZ\ (P),\ Ort\ (O),\ Strasse\ (S),\ Hausnummer\ (H)}$
- funktionale Abhängigkeiten F

$$OSH \rightarrow P,\ P \rightarrow O$$

- Datenbankschema S

$$(OSHP, \{OSH,\ \ PSH\})$$

- PSH auch Kandidatenschlüssel, da $PSH \rightarrow OSHP$ mit PSH minimal
- Schema in 3NF, da alle Attribute Primattribute

BCNF und Abhängigkeitstreue 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

6.4. Entwurfsverfahren

- Dekomposition
- Syntheseverfahren

Dekomposition: Start

Start: initiales Relationenschema R mit allen Attributen und einer von den 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: existiert für Kandidatenschlüssel K mit $K \rightarrow Y, Y \not\rightarrow K, Y \rightarrow A, A \notin KY$ wähle dann:

$$R_1 := R - A \qquad R_2 := YA$$
$$\mathcal{R}_1 := (R_1, \mathcal{K}) \qquad \mathcal{R}_2 := (R_2, \mathcal{K}_2 = \{Y\})$$

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

Syntheseverfahren

- Prinzip: Synthese formt Original-FD-Menge F in resultierende Menge von Schlüsselabhängigkeiten G so um, dass $F \equiv G$ gilt
- „Abhängigkeitstreue“ im Verfahren verankert
- 3NF und Minimalität wird auch erreicht, reihenfolgeunabhängig
- Zeitkomplexität: quadratisch

Syntheseverfahren i

Syntheseverfahren

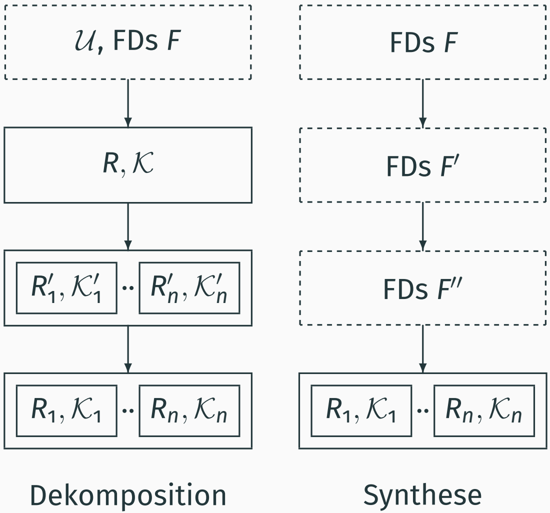
- gegeben: FD-Menge F
- berechne minimale Überdeckung F' ($\equiv F$) durch
  - *Linksreduktion* und *Rechtsreduktion*
- fasse FDs aus F' zu „Äquivalenzklassen“ zusammen
  - FDs in eine Klasse, die gleiche oder äquivalente linke Seiten haben
  - pro Äquivalenzklasse ein Relationenschema mit allen Attributen der zugeordneten FDs
- Falls kein Kandidatenschlüssel zu $\mathcal{U}$ vollständig in einem der Relationenschemata, füge für einen Schlüssel ein Relationenschema hinzu ($\leadsto$ Verbundtreue!)
  - *oder:* erweitere Original-FD-Menge F um $\mathcal{U} \rightarrow \delta$;
    δ Dummy-Attribut, das nach Synthese entfernt wird

Syntheseverfahren ii

Bestimmung einer minimalen Überdeckung

- FD-Menge F ist linksreduzierbar, falls $(X \to Y) \in F$ mit
  $$Z \subset X \ \wedge \ (F - \{X \to Y\}) \cup \{Z \to Y\} \equiv F$$
  - Linksreduktion: Eliminieren von „überflüssigen“ Attributen auf der linken Seite von FDs
- FD-Menge F ist rechtsreduzierbar, falls $(X \to Y) \in F$ mit
  $$Z \subset Y \ \wedge \ (F - \{X \to Y\}) \cup \{X \to Z\} \equiv F$$
  - Rechtsreduktion: Eliminieren von „überflüssigen“ Attributen auf der rechten Seite von FDs
  - Spezialfall: $Z = \emptyset \ \rightsquigarrow$ Eliminieren der FD $X \to Y$, da diese aus $F - \{X \to Y\}$ ableitbar.
- F ist eine minimale Überdeckung, wenn F weder linksreduzierbar noch rechtsreduzierbar ist.

Vergleich Dekomposition — Synthese



- Mehrwertige Abhängigkeit (kurz: MVD, multivalued dependency) $X \rightarrow\!\!\!\rightarrow Y$
- innerhalb einer Relation r wird einem Attributwert von X eine Menge von Y-Werten zugeordnet, unabhängig von den Werten der restlichen Attribute von r
- Beispiel: in $\mathit{Bücher}$

$$\mathit{ISBN} \rightarrow\!\!\!\rightarrow \mathit{Autor}$$
$$\mathit{ISBN} \rightarrow\!\!\!\rightarrow \mathit{Version}$$
$$\mathit{ISBN} \rightarrow\!\!\!\rightarrow \mathit{Stichwort}$$

Mehrwertige Abhängigkeiten

- Schwierigkeiten bei MVDs:
  $R = \{\, \mathtt{Student},\ \mathtt{Fach},\ \mathtt{Vorlesung} \,\}$
  mit FD $\mathtt{Student} \rightarrow \mathtt{Fach}$ und MVD $\mathtt{Fach} \rightarrow\!\!\rightarrow \mathtt{Vorlesung}$
  - zusätzliches Attribut (SWS)
  - zusätzliche FD $\mathtt{Vorlesung} \rightarrow \mathit{SWS}$
  - unsinnige FD $\mathtt{Fach} \rightarrow \mathit{SWS}$ ableitbar
- MVD falsch spezifiziert
- $\mathtt{Fach} \rightarrow\!\!\rightarrow \mathtt{Vorlesung}$ (Werte zum Attribut $\mathtt{Vorlesung}$ unabhängig von den Werten aller restlichen Attribute?)
- sinnvoll
  $\mathtt{Student} \rightarrow \mathtt{Fach}, \mathtt{Vorlesung} \rightarrow \mathit{SWS}$
  und MVD $\mathtt{Fach} \rightarrow\!\!\rightarrow \mathtt{Vorlesung},\ \mathit{SWS}$

Vierte Normalform i

Name	Kind	Hobby
James Bond	Hugo	Autos
James Bond	Egon	Autos
James Bond	Hugo	Action
James Bond	Egon	Action
James Bond	Hugo	Klettern
James Bond	Egon	Klettern

$$Name \twoheadrightarrow Kind, Name \twoheadrightarrow Hobby$$

vierte Normalform (4NF) durch

- Elimination der rechten Seite einer der beiden mehrwertigen Abhängigkeiten,
- linke Seite mit dieser rechten Seite in neue Relation kopiert

Name	Kind
James Bond	Hugo
James Bond	Egon

Name	Hobby
James Bond	Autos
James Bond	Action
James Bond	Klettern

Transformationseigenschaften bei MVDs

„Unabhängigkeit" der Attributmengen Y und Z voneinander: pro X-Wert bildet kartesisches Produkt der Y- und Z-Werte den YZ-Wert

$$
\begin{aligned}
X \rightarrow\!\!\!\rightarrow Y \qquad \Longleftrightarrow \qquad & \forall X\text{-Werte } x : \\
& \pi_{YZ}(\sigma_{X=x}(r)) = \pi_Y(\sigma_{X=x}(r)) \bowtie \pi_Z(\sigma_{X=x}(r))
\end{aligned}
$$

genau für alle $r \in \mathbf{SAT}_R(\{X \rightarrow\!\!\!\rightarrow Y\})$ gilt verbundtreue Dekomposition

$$
r = \pi_{XY}(r) \bowtie \pi_{XZ}(r)
$$

Inklusionsabhängigkeiten

Verallgemeinerung von Fremdschlüsseln

- auf der rechten Seite einer Fremdschlüsselabhängigkeit nicht unbedingt der Primärschlüssel einer Relation: Inklusionsabhängigkeit (kurz: IND, von inclusion dependency)
- X-Werte in einer Relation $r_1(R_1)$ kommen auch als Y-Werte in einer Relation $r_2(R_2)$ vor: Inklusionsabhängigkeit $R_1[X] \subseteq R_2[Y]$