

Kapitel 11: Relationale Entwurfstheorie

11.1 Funktionalabhängigkeiten

11.2 Normalformen

11.3 Relationendekomposition

11.4 Relationensynthese

11.5 Ergänzungen zum algorithmischen Ansatz

Problem

Bestellungen (Datum, KNr, PNr, Bez, Preis, Gewicht, Menge)

"Typische" Ausprägung:

| Datum | KNr | PNr | Bez | Preis | Gewicht | Menge |
|--------------|------------|------------|------------|--------------|----------------|--------------|
| 16.7. | 1 | 1 | Papier | 20.00 | 2.000 | 100 |
| 21.7. | 1 | 1 | Papier | 20.00 | 2.000 | 200 |
| 26.10. | 2 | 1 | Papier | 20.00 | 2.000 | 100 |
| 26.10. | 2 | 5 | Disketten | 20.00 | 0.500 | 50 |

11.1 Funktionalabhängigkeiten (FAs, FDs)

Definition:

Für $X = \{X_1, \dots, X_m\} \subseteq \text{sch}(R)$ und $Y = \{Y_1, \dots, Y_n\} \subseteq \text{sch}(R)$ gilt die *Funktionalabhängigkeit* $X \rightarrow Y$, wenn jederzeit für je zwei Tupel $r, s \in \text{val}(R)$ gelten muß:

$$(r.X_1=s.X_1 \wedge \dots \wedge r.X_m=s.X_m) \Rightarrow (r.Y_1=s.Y_1 \wedge \dots \wedge r.Y_n=s.Y_n)$$

Beispiel:

Bestellungen (Datum, KNr, PNr, Bez, Preis, Gewicht, Menge)

Es gelten:

Datum, KNr, PNR \rightarrow Menge

PNr \rightarrow Bez, Preis, Gewicht

Es gelten *nicht*:

KNr, PNR \rightarrow Menge

PNr \rightarrow Datum

Entwurfsziel: Alle FAs in Primärschlüsselbedingungen ausdrücken

Ableitungsregeln für Funktionalabhängigkeiten

Definition:

Sei F eine Menge von FAs über $\text{sch}(R)$. Die **transitive Hülle** F^+ ist die Menge der aus F logisch ableitbaren FAs.

Ableitungskalkül (Armstrong-Regeln):

Seien $X \subseteq \text{sch}(R)$, $Y \subseteq \text{sch}(R)$, $Z \subseteq \text{sch}(R)$.

Regel 1 (Reflexivität): Wenn $Y \subseteq X$, dann gilt: $X \rightarrow Y$.

Regel 2 (Erweiterung): Wenn $X \rightarrow Y$, dann gilt: $XZ \rightarrow YZ$

Regel 3 (Transitivität): Wenn $X \rightarrow Y$ und $Y \rightarrow Z$, dann gilt: $X \rightarrow Z$.

Weitere Ableitungsregeln:

Regel 4 (Vereinigung): Wenn $X \rightarrow Y$ und $X \rightarrow Z$, dann gilt: $X \rightarrow YZ$.

Regel 5 (Pseudotrans.): Wenn $X \rightarrow Y$ und $WY \rightarrow Z$ mit $W \subseteq \text{sch}(R)$, dann gilt: $XW \rightarrow Z$.

Regel 6 (Zerlegung): Wenn $X \rightarrow Y$ und $Z \subseteq Y$, dann gilt: $X \rightarrow Z$.

Satz: Die Armstrong-Regeln bilden einen korrekten und vollständigen Ableitungskalkül für F^+ .

Beispiel: Ableitung von Funktionalabhängigkeiten

Bestellungen (Datum, KNr, PNr, Bez, Preis, Gewicht, Menge)

$F = \{ \text{Datum, KNr, PNR} \rightarrow \text{Menge},$
 $\text{PNr} \rightarrow \text{Bez, Preis, Gewicht} \}.$

Es folgen:

Datum, KNr, PNR \rightarrow PNr

Bez, Preis, Gewicht \rightarrow Bez

PNr \rightarrow Bez

Datum, KNr, PNr \rightarrow Bez

nach der Reflexivitätsregel,

nach der Reflexivitätsregel,

nach der Transitivitätsregel,

nach der Transitivitätsregel

Ableitbarkeitstest für FAs: Xplus-Algorithmus

Definition:

Sei R eine Relation mit $\text{sch}(R)$ und FA-Menge F , und sei $X \subseteq \text{sch}(R)$. Die *Hülle* X^+ von X ist die Menge aller $A \in \text{sch}(R)$ mit $X \rightarrow A \in F^+$.

Xplus-Algorithmus:

```
procedure xplus ( $X$ : set of attribute): set of attribute;  
var H: set of attribute;  
    newattr: Boolean;  
begin  
    H := X; newattr := true;  
    while newattr do (* Invariante:  $X \rightarrow H$  *)  
        newattr := false;  
        for each  $Y \rightarrow Z \in F$  do  
            if  $Y \subseteq H$  then  
                if not ( $Z \subseteq H$ ) then H :=  $H \cup Z$ ; newattr := true; fi; fi;  
        od  
    od  
    return H;  
end xplus;
```

Beispiel: Xplus-Algorithmus

$\text{sch}(R) = \{A, B, C, D, E, G\}$

$F = \{AB \rightarrow C, ACD \rightarrow B,$
 $BC \rightarrow D, BE \rightarrow C,$
 $C \rightarrow A, CG \rightarrow BD, CE \rightarrow AG,$
 $D \rightarrow EG\}$

$X = \{B, D\}$

Bestimmung aller Schlüsselkandidaten

Definition:

Sei R eine Relation mit $\text{sch}(R)$ und FA-Menge F , und sei $X \subseteq \text{sch}(R)$.

$X \subseteq \text{sch}(R)$ ist ein *Schlüsselkandidat*, wenn $X \rightarrow \text{sch}(R) \in F^+$ gilt und es keine echte Teilmenge von X gibt, die diese Eigenschaft hat.

$A \in \text{sch}(R)$ ist Schl.attribut, wenn es in einem Schl.kandidaten vorkommt.

procedure findkeys: **set of set of** attribute;

var K: **set of set of** attribute; iskey: Boolean;

K := \emptyset ;

for each S $\in 2^{\text{sch}(R)}$ **do**

iskey := true;

if xplus (S) $\neq \text{sch}(R)$ **then** iskey := false **fi**;

for each A $\in S$ **while** iskey **do**

if xplus (S - {A}) = sch(R) **then** iskey := false **fi**

od;

if iskey **then** K := K \cup {S} **fi**

od;

return K;

11.2 Relationale Normalformen

Definition:

Sei R eine Relation mit $\text{sch}(R)$ und FA-Menge F .

R ist in **Boyce-Codd-Normalform (BCNF)**, wenn für jede FA $X \rightarrow A \in F^+$ mit $X \subseteq \text{sch}(R)$, $A \in \text{sch}(R)$ und $A \notin X$ gilt, daß X einen Schlüsselkandidaten enthält.

Definition:

Sei R eine Relation mit $\text{sch}(R)$ und FA-Menge F .

R ist in **3. Normalform (3NF)**, wenn für jede FA $X \rightarrow A \in F^+$ mit $X \subseteq \text{sch}(R)$, $A \in \text{sch}(R)$ und $A \notin X$ gilt, daß X einen Schlüsselkandidaten enthält oder A ein Schlüsselattribut ist.

Satz:

Jede Relation in BCNF ist auch in 3NF.

Algorithmus für BCNF-Test

```
procedure BCNFtest: Boolean;  
var isbcnf: Boolean;  
begin  
    isbcnf := true;  
    for each  $X \rightarrow A \in F$  while isbcnf do  
        if  $A \notin X$  then if  $XPlus(X) \neq sch(R)$  then isbcnf := false fi fi;  
    od  
    return isbcnf;  
end BCNFtest;
```

Beispiele: Normalformen

- 1) Best (Datum, KNr, PNr, Menge) mit
 $F = \{\text{Datum, KNr, PNr} \rightarrow \text{Menge}\}$
und
Prod (PNr, Bez, Preis, Gewicht) mit
 $F = \{\text{PNr} \rightarrow \text{Bez, Preis, Gewicht}\}$
- 2) Vorlesungen (Titel, Zeit, Raum) mit
 $F = \{\text{Titel} \rightarrow \text{Raum,}$
 $\text{Zeit, Raum} \rightarrow \text{Titel}\}$

Beispielausprägung:

| Titel | Zeit | Raum |
|-----------------|-------------|-------------|
| Datenbanken | Di 9-11 | HS 2 |
| Datenbanken | Do 9-11 | HS 2 |
| Compiler | Di 9-11 | HS 3 |
| Compiler | Mi 13-15 | HS 3 |
| Betriebssysteme | Mi 13-15 | HS 2 |

11.3 Relationendekomposition

Definition:

Sei R eine Relation mit $\text{sch}(R)$ und FA-Menge F .

Eine Zerlegung von R in $R_1 (X_1), \dots, R_k (X_k)$ mit $X_i \subseteq \text{sch}(R)$ und $X_1 \cup \dots \cup X_k = \text{sch}(R)$ heißt *abhängigkeitsbewahrend*, wenn gilt:

$$\left((F^+ /_{X_1})^+ \cup \dots \cup (F^+ /_{X_k})^+ \right)^+ = F^+$$

Definition:

Sei R eine Relation mit $\text{sch}(R)$ und FA-Menge F .

Eine Zerlegung von R in $R_1 (X_1), \dots, R_k (X_k)$ mit $X_i \subseteq \text{sch}(R)$ und $X_1 \cup \dots \cup X_k = \text{sch}(R)$ heißt (*projektion-join-*) *verlustfrei*, wenn für alle möglichen Ausprägungen, die F erfüllen, gilt:

$$\pi[X_1](R) \bowtie \dots \bowtie \pi[X_k](R) = R.$$

Beispiele: Abhängigkeitsbewahrung

1) Bestellungen (Datum, KNr, PNr, Bez, Preis, Gewicht, Menge) mit

$$F = \{\text{Datum, KNr, PNR} \rightarrow \text{Menge}, \\ \text{PNr} \rightarrow \text{Bez, Preis, Gewicht}\}$$

Zerlegung in

Best (Datum, KNr, PNr, Menge) mit

$$F1 = \{\text{Datum, KNr, PNR} \rightarrow \text{Menge}\}$$

und

Prod (PNr, Bez, Preis, Gewicht) mit

$$F2 = \{\text{PNr} \rightarrow \text{Bez, Preis, Gewicht}\}$$

2) Vorlesungen (Titel, Zeit, Raum) mit

$$F = \{\text{Titel} \rightarrow \text{Raum}, \\ \text{Zeit, Raum} \rightarrow \text{Titel}\}$$

Zerlegung in

Vorlesungsräume (Titel, Raum) mit

$$F1 = \{\text{Titel} \rightarrow \text{Raum}\}$$

und

Vorlesungszeiten (Titel, Zeit) mit $F2 = \emptyset$

Beispiele: Verlustfreiheit

Bestellungen (Datum, KNr, PNr, Bez, Preis, Gewicht, Menge) mit
 $F = \{\text{Datum, PNr, KNr} \rightarrow \text{Menge, PNr} \rightarrow \text{Bez, Preis, Gewicht}\}$

| Datum | KNr | PNr | Bez | Preis | Ge- wicht | Menge |
|--------------|------------|------------|------------|--------------|----------------------|--------------|
| 16.7. | 1 | 1 | Papier | 20.00 | 2.000 | 100 |
| 16.7. | 1 | 5 | Disketten | 20.00 | 0.500 | 50 |

Zerlegung in:

| Datum | KNr | PNr | Menge |
|--------------|------------|------------|--------------|
| 16.7. | 1 | 1 | 100 |
| 16.7. | 1 | 5 | 50 |

und

| Datum | Bez | Preis | Gewicht |
|--------------|----------------|--------------|----------------|
| 16.7. | Papier | 20.00 | 2.000 |
| 16.7. | Disket- ten | 20.00 | 0.500 |

Verlustfreie BCNF-Dekomposition

Satz: Zu jeder Relation R gibt es eine abhängigkeitsbewahrende und verlustfreie Zerlegung von R in 3NF-Relationen.

Satz: Zu jeder Relation R gibt es eine verlustfreie Zerlegung von R in BCNF-Relationen..

Satz: Sei R eine Relation mit $\text{sch}(R)$ und einer FA-Menge F.

Sei $X \rightarrow Y \in F^+$ mit $X \cap Y = \emptyset$. Die Zerlegung von R in $R' (X, Y)$ und $R'' (X, \text{sch}(R) - Y)$ ist verlustfrei.

Algorithmus:

Teste, ob R in BCNF ist

Falls R nicht in BCNF ist

(* es gibt eine Funktionalabhängigkeit $X \rightarrow Y \in F^+$ mit $X \cap Y = \emptyset$ und $X \rightarrow \text{sch}(R) \notin F^+$ *)

dann

zerlege R in $R' (X, Y)$ und $R'' (X, \text{sch}(R) - Y)$

wiederhole den Zerlegungsalgorithmus für R' und R''

Beispiel: Relationendekomposition

R (FlugNr, Datum, Abflugzeit, FSNr, SitzNr, TicketNr, Name, Adresse, GNr, Gewicht)

F = { FlugNr, Datum \rightarrow Abflugzeit, FSNr,
FlugNr, Datum, TicketNr \rightarrow SitzNr,
TicketNr \rightarrow Name, Adresse,
GNr \rightarrow Gewicht }

Schlüsselkandidat:

FlugNr, Datum, TicketNr, GNr

Zerlegungsschritte:

1. Zerlegung von R entlang von FlugNr, Datum \rightarrow Abflugzeit, FSNr:
R1 (FlugNr, Datum, Abflugzeit, FSNr) und
R2 (FlugNr, Datum, SitzNr, TicketNr, Name, Adresse, GNr, Gewicht)
2. Zerlegung von R2 entlang von FlugNr, Datum, TicketNr \rightarrow SitzNr:
R21 (FlugNr, Datum, TicketNr, SitzNr) und
R22 (FlugNr, Datum, TicketNr, Name, Adresse, GNr, Gewicht)
3. Zerlegung von R22 entlang von TicketNr \rightarrow Name, Adresse:
R221 (TicketNr, Name, Adresse) und
R222 (TicketNr, FlugNr, Datum, GNr, Gewicht)
4. Zerlegung von R222 entlang von GNr \rightarrow Gewicht:
R2221 (GNr, Gewicht) und
R2222 (GNr, FlugNr, Datum, TicketNr)

11.4 Relationensynthese

Definition:

Sei eine Relation mit $\text{sch}(R)$ und FA-Menge F . Eine FA-Menge F' heißt *Überdeckung* (engl. cover) von F , wenn gilt: $(F')^+ = F^+$.

Definition:

Eine Überdeckung F' von F heißt *minimal*, wenn gilt:

- 1) In jeder FA von F' hat die rechte Seite ein Attribut.
- 2) Keine echte Teilmenge von F' ist eine Überdeckung von F .
- 3) Für jede FA $X \rightarrow A \in F'$ und jede echte Teilmenge X' von X
 $F'' := F' - \{ X \rightarrow A \} \cup \{ X' \rightarrow A \}$ keine Überdeckung mehr von F .

Beispiel: minimale Überdeckung

$\text{sch}(\mathbf{R}) = \{ \text{Ang}(\text{estellten})\text{Nr}, \text{Name}, \text{Gehalt}, \text{Leistung}(\text{sbeurteilung}), \text{Tarif}(\text{klasse}), \text{Abt}(\text{eilungs})\text{Nr}, \text{ProjektNr}, \text{Projektleiter}, \text{Abt}(\text{eilungs})\text{leiter} \}$

$F = \{ \text{AngNr} \rightarrow \text{Name},$
 $\text{AngNr} \rightarrow \text{Gehalt},$
 $\text{AngNr} \rightarrow \text{Tarif},$
 $\text{AngNr} \rightarrow \text{AbtNr},$
 $\text{AngNr} \rightarrow \text{Abtleiter},$
 $\text{ProjektNr} \rightarrow \text{Projektleiter},$
 $\text{AbtNr} \rightarrow \text{Abtleiter},$
 $\text{AngNr}, \text{Abtleiter} \rightarrow \text{Leistung},$
 $\text{Tarif}, \text{Leistung} \rightarrow \text{Gehalt} \}$

Schritt 1: $\rightarrow \{ \text{AngNr} \rightarrow \text{Name},$
 $\text{AngNr} \rightarrow \text{Gehalt},$
 $\text{AngNr} \rightarrow \text{Tarif},$
 $\text{AngNr} \rightarrow \text{AbtNr},$
 $\text{AngNr} \rightarrow \text{Abtleiter},$
 $\text{ProjektNr} \rightarrow \text{Projektleiter},$
 $\text{AbtNr} \rightarrow \text{Abtleiter},$
 $\text{AngNr} \rightarrow \text{Leistung},$
 $\text{Tarif}, \text{Leistung} \rightarrow \text{Gehalt} \}$

Schritt 2: $\rightarrow \{ \text{AngNr} \rightarrow \text{Name},$
 $\text{AngNr} \rightarrow \text{Tarif},$
 $\text{AngNr} \rightarrow \text{AbtNr},$
 $\text{ProjektNr} \rightarrow \text{Projektleiter},$
 $\text{AbtNr} \rightarrow \text{Abtleiter},$
 $\text{AngNr} \rightarrow \text{Leistung},$
 $\text{Tarif}, \text{Leistung} \rightarrow \text{Gehalt} \} = F'$

Algorithmus zur Berechnung der Min-Cover

```
procedure mincover: set of FDs;  
  var G: set of FDs;  
  procedure xplus (arg1: set of FDs, arg2: set of attribute): set of attribute;  
  procedure reduce (arg: set of FDs): set of FDs;  
    var H: set of FDs;  
    H := arg;  
    for each  $X \rightarrow A \in H$  do  
      for each  $C \in X$  do  
        Hred :=  $H - \{ X \rightarrow A \} \cup \{ (X - \{ C \}) \rightarrow A \}$ ;  
        if  $A \in \text{xplus}(H, X - \{ C \})$  then  $H := \text{Hred}$  fi;  
      od;  
    od;  
  return H;
```

```
G := reduce(F);  
for each  $X \rightarrow A \in G$  do  
  G :=  $G - \{ X \rightarrow A \}$ ;  
  if not  $(A \in \text{xplus}(G, X))$  then  $G := G \cup \{ X \rightarrow A \}$  fi;  
od;  
return G;
```

Algorithmus zur Relationensynthese

Schritt 1: Berechne eine minimale Überdeckung F' von F .

Schritt 2: Partitioniere die FAs in F' nach gleichen linken Seiten, d.h. für jedes X_i fasse alle $X_i \rightarrow A_1, \dots, X_i \rightarrow A_{k_i}$ zusammen

Schritt 3: Bilde für jede Partition von Schritt 2 eine Relation R_i mit $\text{sch}(R_i) = X_i \cup \{A_1, \dots, A_{k_i}\}$.

Schritt 4: Falls $\text{sch}(R) - (\cup_i \text{sch}(R_i))$ nichtleer ist, bilde eine weitere Relation R' mit $\text{sch}(R') = \text{sch}(R) - (\cup_i \text{sch}(R_i))$.

Schritt 5: Falls keine der Relationen R_i, R' einen Schlüssel von R enthält, erzeuge eine Relation R'' , deren Schema ein Schlüssel von R ist.

Satz:

Der Algorithmus zur Relationensynthese erzeugt eine abhängigkeitsbewahrende und verlustfreie 3NF-Zerlegung von R .

Beispiel: Relationensynthese

$\text{sch}(R) = \{ \text{Ang(estellten)Nr, Name, Gehalt,} \\ \text{Leistung(sbeurteilung), Tarif(klasse),} \\ \text{Abt(eilungs)Nr, ProjektNr, Projektleiter, Abt(eilungs)leiter} \}$

$F' = \{ \text{AngNr} \rightarrow \text{Name,} \\ \text{AngNr} \rightarrow \text{Tarif,} \\ \text{AngNr} \rightarrow \text{AbtNr,} \\ \text{ProjektNr} \rightarrow \text{Projektleiter,} \\ \text{AbtNr} \rightarrow \text{Abtleiter,} \\ \text{AngNr} \rightarrow \text{Leistung,} \\ \text{Tarif, Leistung} \rightarrow \text{Gehalt} \}$

Ergebnis der Relationensynthese:

Angestellte (AngNr, Name, Tarif, Leistung, AbtNr)

Projekte (ProjektNr, Projektleiter)

Abteilungen (AbtNr, Abtleiter)

Lohn (Tarif, Leistung, Gehalt)

Projektmitarbeit (AngNr, ProjektNr)

11.5 Ergänzungen (1)

1) weitere Zerlegung von BCNF-Relationen:

Angestellte (ANr, Informatikkenntnisse, Kinder) mit $F = \emptyset$

| ANr | Informatikkenntnisse | Kinder |
|------------|-----------------------------|---------------|
| 1 | Leda | Sabrina |
| 1 | Oracle | Sabrina |
| 2 | Leda | Pierre |
| 2 | Leda | Sabrina |

sollte weiter zerlegt werden in

AngInfKenntnisse (ANr, Informatikkenntnisse) und
AngKinder (ANr, Kinder)

2) unnötig feine Zerlegungen:

Produkte (PNr, Bez, Preis) mit $F = \{PNr \rightarrow Bez, PNr \rightarrow Preis\}$

sollte nicht zerlegt werden in

Produktbez (PNr, Bez) und
Produktpreise (PNr, Preis)

11.5 Ergänzungen (2)

3) Zerlegung aufgrund irrelevanter Funktionalabhängigkeiten:

Bestellungen (Datum, KNr, PNr, Bez, Preis, Gewicht, Menge, Summe)

mit $F = \{ \text{Datum, KNr, PNr} \rightarrow \text{Menge},$

$\text{PNr} \rightarrow \text{Bez, Preis, Gewicht},$

$\text{Preis, Menge} \rightarrow \text{Summe} \}$

sollte nicht zur Relatione führen

Preise (Preis, Menge, Summe)

4) Nicht-BCNF-Relationen, die unproblematisch sind:

Kunden (KNr, Name, Postleitzahl, Stadt, Strasse)

mit $F = \{ \text{KNr} \rightarrow \text{Name, Postleitzahl, Stadt, Strasse},$

$\text{Postleitzahl} \rightarrow \text{Stadt} \}$

muß nicht notwendigerweise zerlegt werden in

Kundenadressen (KNr, Name, Postleitzahl, Strasse) und

Postleitzahlenverzeichnis (Postleitzahl, Stadt)

5) M:N-Relationships ohne Attribute:

Sitze (FlugNr, Datum, SitzNr)

muß bei der Relationendekomposition ggf. hinzugefügt werden