5 Relationaler Datenbank-Entwurf

Problem: Zu modellierender Ausschnitt der realen Welt durch unterschiedlichste DB-Schemata darstellbar

- Welche Schemata sind **äquivalent**, d.h. stellen die gleiche Information dar? Welche sind **gute** Schemata?

- Gesucht: Gütekriterien und Äquivalenzkriterien; Methoden zur Schemaverbesserung

Grundlage: Relationenschemata mit **funktionalen Abhängigkeiten (FAen)**; spezielle, für das relationale Datenmodell typische Integritätsbedingungen

Erwünscht: Vermeidung von

- Redundanz

- Anomalien
  Änderungs-, Einfüge- und Löschanomalien

Entwurfsziele

- Redundanzen und Anomalien beseitigen

- Äquivalenz garantieren

- Vorgehen: Normalisierung, d.h. Umwandlung von gegebenen Relationenschemata zu Relationenschemata in Normalform

Beispiel: KAL (Kunde-Auftrag-Lieferant)

S1 KAL(KName, KAdr, Kto, Ware, Menge, LName, LAdr, Preis)

universelle Relation: alle Attribute

S2 Original-KAL

KUNDE (KName, KAdr, Kto)

AUFTRAG(KName, Ware, Menge)

LIEFERANT(LName, LAdr, Ware, Preis)

S3 KUNDE(KName, KAdr, Kto)

AUFTRAG(KName, Ware, Menge)

LIEFERANT'(LName, LAdr)

ANGEBOT(LName, Ware, Preis)

S4 wie S3, jedoch zusätzlich:

BEZUG(KName, Ware, LName)

S5-S7 wie S2-S4, jedoch KUNDE zerlegt in

KUNADR(KName, KAdr)

KUNKTO(KName, Kto)

Nachteile (anhand von Original-KAL)

- Redundanz: Adresse LAdr eines Lieferanten wird zu jeder Ware aufgeführt

- Änderungs-Anomalien (Einfüge-, Update-, Lösch-Anomalien)

  a) Adresse kann nicht ohne Ware eingefügt werden

  b) Adresse kann beim Update einer Ware ebenfalls geändert werden

  c) Adresse wird mit der letzten Ware gelöscht

Verbesserung

- LIEFERANT'(LName, LAdr)
  ANGEBOT(LName, Ware, Preis)
  also Zerlegung der Relation LIEFERANT

- Problem: Ist LIEFERANT' zusammen mit ANGEBOT äquivalent zu LIEFERANT?

- neuer Nachteil: zum Teil aufwendigere Anfragen, z.B. Lieferantenadressen zu Waren erfordert Verbundbildung

Beispiele für funktionale Abhängigkeiten

Beispiel: KAL

KUNDE(KName, KAdr, Kto)

KName $\rightarrow$ KAdr Kto

AUFTRAG(KName, Ware, Menge)

KName Ware $\rightarrow$ Menge

LIEFERANT(LName, LAdr, Ware, Preis)

LName $\rightarrow$ LAdr

LName Ware $\rightarrow$ Preis

Beispiel: STADT

STADT(SName,X,Y,EinwAnz)

SName $\rightarrow$ X, Y

X, Y $\rightarrow$ SName

SName $\rightarrow$ EinwAnz

Obacht: als Schlüssel kommen {SName} und {X,Y} in Frage; dies wird aber nicht durch STADT($\underline{\text{SName}}$,$\underline{\text{X}}$,$\underline{\text{Y}}$,EinwAnz) ausgedrückt

Beispiel: WOHNUNGSMARKT (WM)

WM	VName	Adresse	Wnr	Miete	MName	HTier
r	Ada	Idaweg3	1	800	Cher	Fisch
	Ada	Idaweg2	2	900	Dan	Hund
	Ada	Idaweg2	2	900	Dan	Maus
	Ada	Idaweg1	3	750	Eva	Vogel
	Bob	Idaweg4	1	300	Fred	Löwe
	Bob	Idaweg5	2	300	Gil	Hund
	Bob	Idaweg5	2	300	Gil	Katze
	Bob	Idaweg4	3	340	Hal	Katze

FA: Adresse, Wnr $\rightarrow$ MName, Miete, VName
MName $\rightarrow$ Adresse, Wnr, VName

VERMIETUNG	VName	Adresse	Wnr	Miete	MName
$r_1 = \pi_1(r)$	Ada	Idaweg3	1	800	Cher
	Ada	Idaweg2	2	900	Dan
	Ada	Idaweg1	3	750	Eva
	Bob	Idaweg4	1	300	Fred
	Bob	Idaweg5	2	300	Gil
	Bob	Idaweg4	3	340	Hal

TIERE	MName	HTier
$r_2 = \pi_2(r)$	Cher	Fisch
	Dan	Hund
	Dan	Maus
	Eva	Vogel
	Fred	Löwe
	Gil	Hund
	Gil	Katze
	Hal	Katze

$$r = r_1 * r_2$$

Funktionale Abhängigkeiten

Gegeben seien eine Attributmenge $A = \{A_1, ..., A_n\}$ und ein Relationenschema $R(A) = R(A_1, ..., A_n)$

Definition: Eine **funktionale Abhängigkeit (FA)** ist eine spezielle Integritätsbedingung, die syntaktisch durch ein Paar von Teilmengen $X, Y \subseteq A$ beschrieben wird

Notation: $X \to Y$ wobei X und Y ohne Mengenklammern und Kommata notiert werden; gesprochen 'X bestimmt Y'

Beispiel: 'LName Ware $\to$ LAdr Ware' statt '{LName, Ware} $\to$ {LAdr, Ware}'

'LName Ware $\to$ LAdr Ware' gleichbedeutend mit 'Ware LName $\to$ Ware LAdr'

Definition: Eine Relation r zum Schema $R(A)$ **erfüllt** die FA $X \to Y$ gdw. für je zwei Tupel $t_1, t_2 \in r$ gilt:
$\pi_X(t_1) = \pi_X(t_2) \Rightarrow \pi_Y(t_1) = \pi_Y(t_2)$

Äquivalente IB im Tupelkalkül:
$\forall t_1, t_2 : R \ (t_1.X = t_2.X \Rightarrow t_1.Y = t_2.Y)$
wobei X, Y Attributmengen und
$t_1.Z = t_2.Z :\Leftrightarrow t_1.B_1 = t_2.B_1 \wedge ... \wedge t_1.B_m = t_2.B_m$
für $Z = \{B_1, ..., B_m\}$

Beispiel: LIEFERANT-Relationen erüllen noch weitere FAen:

(1) LName $\rightarrow$ LName, LName Ware $\rightarrow$ Ware, ...

(2) LName Ware $\rightarrow$ LAdr, ...

zu (1)

Definition: Eine FA $X \rightarrow Y$ heißt **trivial** gdw. $X \rightarrow Y$ von jeder Relation erfüllt wird

Lemma: FA $X \rightarrow Y$ trivial $\Leftrightarrow$ $X \supseteq Y$

zu (2)

Definition: Es seien F eine Menge von FAen zum Schema $R(A)$ und $X, Y \subseteq A$; 'F **impliziert** $X \rightarrow Y$' oder 'unter F ist $X \rightarrow Y$ **gültig**', notiert als $F \models X \rightarrow Y$, gdw. für alle Relationen r gilt: (r erfüllt alle FAen in F) $\Rightarrow$ (r erfüllt $X \rightarrow Y$)

Beispiel:

{ LName $\rightarrow$ LAdr } $\models$ LName Ware $\rightarrow$ LAdr

Definition: Abschluß von F:
$F^+ = \{X \rightarrow Y \mid F \models X \rightarrow Y\}$

Herleitungsregeln

Es gibt System von **Herleitungsregeln**, mit dem sich syntaktisch (ohne Untersuchung von Relationen) beweisen läßt, ob $F \models X \to Y$ gilt:

Notation: $F \vdash X \to Y$; ausgesprochen '$X \to Y$ herleitbar aus F' oder 'unter F ist $X \to Y$ herleitbar'

Armstrong–Axiome: $X, Y, Z, W \subseteq A$

I: Anfang: Für alle $X \to Y \in F : F \vdash X \to Y$

II: Reflexivität: Für alle $Y \subseteq X : F \vdash X \to Y$

III: Expansivität: Für alle X, Y, W, Z mit $W \supseteq Z : F \vdash X \to Y \Rightarrow F \vdash XW \to YZ$

$XW = X \cup W; \quad YZ = Y \cup Z$

IV: Transitivität: Für alle X, Y, Z:

$F \vdash X \to Y$ und $F \vdash Y \to Z \Rightarrow F \vdash X \to Z$

Definition: Hülle von F: $F^* := \{X \to Y \mid F \vdash X \to Y\}$

Satz: Für beliebige F und X, Y gilt:
$F \vdash X \to Y \Leftrightarrow F \models X \to Y$

D.h. obige Regeln bilden ein **konsistentes** ($\Rightarrow$) und **vollständiges** ($\Leftarrow$) Herleitungssystem

Kurz: Für beliebiges F gilt: $F^* = F^+$

Beispiel:

$F = \{ A\ W \rightarrow MN\ M\ VN\ (1),\ MN \rightarrow A\ W\ VN\ (2) \}$

nach I:

$F \vdash A\ W \rightarrow MN\ M\ VN$

$F \vdash MN \rightarrow A\ W\ VN$

nach II:

$F \vdash A\ W\ MN\ M\ VN \rightarrow MN\ VN$

...

nach III:

$F \vdash A\ W\ MN \rightarrow MN\ M\ VN$
aus (1) mit $Z=\emptyset$, $W=\{MN\}$

$F \vdash MN\ M\ VN \rightarrow A\ W\ MN\ M\ VN$
aus (2) mit $Z=W=\{MN, M, VN\}$

...

nach IV:

$(F \vdash MN \rightarrow A\ W)$ und $(F \vdash A\ W \rightarrow M)$ dann
$(F \vdash MN \rightarrow M)$

...

Herleitbarkeit ist entscheidbar

Verfahren: Gegeben F und $X \to Y$

Entscheide ob $F \vdash X \to Y$ gilt

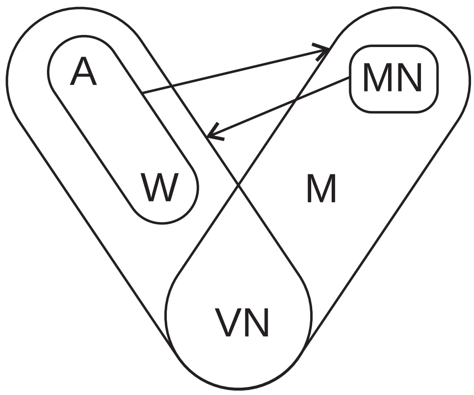
1. $Z := X$

1. repeat
   for each $U \subseteq Z, U \to V \in F$ do $Z := Z \cup V$
   until Z unverändert /* $Z = X^*$ */

3. Teste $Y \subseteq Z$ /* $X \to Y \in F^*$ gdw. $Y \subseteq X^*$ */

Bezeichnung: $X^* :=$ Hülle von X; maximale, von X aus erreichbare Attributmenge

Darstellung: Menge von FAen dargestellt als Graph mit Knoten=Attributmengen und Kanten=FAen



Beispiel:

{ MN }* = { A, W, MN, M, VN }
{ A, W }* = { A, W, MN, M, VN }
{ M }* = { M }
{ A }* = { A }

Minimale Überdeckungen

Zu einer Menge F von FAen gibt es eine minimale, nicht notwendigerweise eindeutige Überdeckung $G = MinCover(F)$

Definition:

G **Überdeckung** von F gdw. $G^* = F^*$

G **minimal** gdw. für alle $X \to Y \in G$ gilt:

(i) $|Y| = 1$
(Rechte Seiten einelementig)

(ii) $(Z \subseteq X \land G \vdash Z \to Y) \Rightarrow Z = X$
(Linke Seiten nicht redundant)

(iii) $G - \{X \to Y\} \not\vdash X \to Y$
(G nicht redundant)

Verfahren: Konstruktion einer minimalen Überdeckung

0. $G := F$

1. for each $X \to Y \in G, Y = B_1 \cdots B_m$ do
   $G := G - \{X \to Y\} \cup \{X \to B_1, \cdots, X \to B_m\}$

2. repeat
   for each $X \to Y \in G, Z \subset X, G \vdash Z \to Y$ do
   $G := G - \{X \to Y\} \cup \{Z \to Y\}$
   until G unverändert

3. for each $X \to Y \in G$ do
   if $G - \{X \to Y\} \vdash X \to Y$ then
   $G := G - \{X \to Y\}$

Beispiele zu minimalen Überdeckungen

nach 0.	A W → MN M VN MN → A W VN	
nach 1.	A W → MN A W → M A W → VN MN → A MN → W MN → VN	ableitbar
nach 2.	– unverändert –	
nach 3.	A W → MN A W → M MN → A MN → W MN → VN	

nach 0.	A → B B → C C → A
	B → A C → B A → C
nach 1. und 2.	– unverändert –
nach 3.	A → B B → C C → A
oder	
nach 3.	B → A C → B A → C

...

Schlüssel und Determinanten

Definition: $X \subseteq A$ heißt **Schlüsselkandidat** oder auch kurz **Schlüssel** von $R(A)$ bzgl. F gdw.

(i) $F \models X \rightarrow A$ und

(ii) $F \models Y \rightarrow A \;\wedge\; Y \subseteq X \quad \Rightarrow \; Y = X$

d.h. X ist minimale Attributmenge mit $X \rightarrow A$

Definition: Seien $X_1, ..., X_k \subseteq A$ alle Schlüsselkandidaten von $R(A)$; $A_i \in A$ heißt **Nicht-Schlüssel-Attribut** (NSA), gdw.
$A_i \notin X_1 \cup ... \cup X_k$

Definition: Seien $D \subseteq A, Y \in A, Y \notin D$; D heißt **Determinante** von Y in $R(A)$ bzgl. F gdw.

(i) $F \models D \rightarrow Y$ und

(ii) $F \models C \rightarrow Y \;\wedge\; C \subseteq D \;\Rightarrow\; C = D$

(i): D determiniert Y; (ii): D determiniert Y minimal

Normalformen

Gegeben seien Relationenschema $R(A)$ und FAen F

Definition: $R(A)$ ist in **n-ter Normalform** (1NF, 2NF, 3NF) bzgl. F gdw ...

1NF: Alle Attributwerte sind atomar, d.h. nicht selbst Mengen von Werten

2NF: Kein NSA Z hängt partiell von einem Schlüssel X ab, d.h. $\neg\exists Y(Y \subset X \wedge Y \rightarrow Z)$ mit $Y \rightarrow Z$ nicht trivial

3NF: Kein NSA Z hängt transitiv von einem Schlüssel X ab, d.h. $\neg\exists Y(X \rightarrow Y \wedge Y \rightarrow Z)$ mit $Y \rightarrow Z$ nicht trivial und $X \not\leftarrow Y$

Satz: $R(A)$ ist in 3NF bzgl. F gdw. alle Determinanten von allen Nicht-Schlüssel-Attributen Schlüssel sind

Definition: $R(A)$ ist in **Boyce-Codd-Normalform** (BCNF) bzgl. F gdw. alle Determinanten von allen Attributen Schlüssel sind

Überblick: 1NF $\Leftarrow$ 2NF $\Leftarrow$ 3NF $\Leftarrow$ BCNF

Beispiele: Normalformen, Schlüssel, Determinanten

- LIEFERANT(LName, LAdr, Ware, Preis)

  LName $\rightarrow$ LAdr

  LAdr Ware $\rightarrow$ Preis

  Schlüsselkandidaten: - { LName, Ware }

  Werte atomar, deswegen **in 1NF**; aber es gibt partielle Abhängigkeit 'LName $\rightarrow$ LAdr', deswegen nicht in 2NF

- LIEFORT(LName, LAdr, LOrt)

  LName $\rightarrow$ LAdr

  LAdr $\rightarrow$ LOrt

  Schlüsselkandidaten: - { LName }

  keine partielle Abhängigkeit, deswegen **in 2NF**; aber es gibt transitive Abhängigkeit, deswegen nicht in 3NF

- SSP(Stadt, StrasseNr, Plz)

  Stadt StrasseNr $\rightarrow$ Plz

  Plz $\rightarrow$ Stadt

  Schlüsselkandidaten: - { Stadt, StrasseNr }
- { Plz, StrasseNr }

  keine NSAe, deswegen **in 3NF**; aber Plz Determinante von Stadt, deswegen nicht in BCNF

- STADT(SName,X,Y,EinwAnz)

  SName $\rightarrow$ X, Y

  X, Y $\rightarrow$ SName

  SName $\rightarrow$ EinwAnz

  Schlüsselkandidaten: - { SName }
  - { X, Y }

  Determinanten von Attributen sind Schlüsselkandidaten, deswegen **in BCNF**

Attribut		Determinanten
SName	$\leftarrow$	{ X, Y }
X	$\leftarrow$	{ SName }
Y	$\leftarrow$	{ SName }
EinwAnz	$\leftarrow$	{ SName }, { X, Y }

Überblick Normalformen

Kriterium	NF	Beispiel
Attributewerte atomar	1NF	LIEFERANT
keine partiellen Abängigkeiten von NSAen	2NF	LIEFORT
keine transitiven Abängigkeiten von NSAen = Determinanten von NSAen sind Schlüssel	3NF	SSP
Determinanten von Attributen sind Schlüssel	BCNF	STADT

Definition: Zerlegung

$$
\begin{array}{ccc}
R(A) & \mapsto & R_1(B_1), R_2(B_2) \\
F & \mapsto & F_1, F_2
\end{array}
$$

mit $A = B_1 \cup B_2$
mit $F_i = \{X \to Y \mid F \models X \to Y$ und $X, Y \subseteq B_i\}$
(oder Erzeugendenmenge)

Beispiel

SSP	Stadt	StrasseNr	Plz
	HB	Idaweg12	28203
	HH	Idaweg12	22767

Projektion π $\swarrow$ $\searrow$

SS	Stadt	StrasseNr
	HB	Idaweg12
	HH	Idaweg12

SP	StrasseNr	Plz
	Idaweg12	28203
	Idaweg12	22767

Verbund $*$ $\searrow$ $\swarrow$

SSP	Stadt	StrasseNr	Plz
	HB	Idaweg12	28203
	HB	Idaweg12	22767
	HH	Idaweg12	28203
	HH	Idaweg12	22767

Äquivalenzkriterien

Verlustlosigkeit

Definition: Die Zerlegung
$(R(A), F) \Rightarrow (R_1(B_1), F_1; R_2(B_2), F_2)$ ist **verlustlos**
gdw. für alle Relationen r zu $R(A)$, die F erfüllen,
gilt: $r = \pi_{B_1}(r) * \pi_{B_2}(r)$

Bemerkung: Generell gilt $r \subseteq \pi_{B_1}(r) * \pi_{B_2}(r)$ und ...

Beispiel: Zerlegung von SSP

... $r_1 \supseteq \pi_{B_1}(r_1 * r_2)$ und $r_2 \supseteq \pi_{B_2}(r_1 * r_2)$

Beispiel: $R_1(A, B)$ mit $r_1 = \{(a, b), (a, b_1)\}$ und
$R_2(B, C)$ mit $r_2 = \{(b, c), (b_2, c)\}$
$\pi_{A,B}(r_1 * r_2) = \{(a, b)\}$ und $\pi_{B,C}(r_1 * r_2) = \{(b, c)\}$

Satz: Die obige Zerlegung verlustlos gdw.
(1) $F \models B_1 \cap B_2 \rightarrow B_1 - B_2$ oder
(2) $F \models B_1 \cap B_2 \rightarrow B_2 - B_1$

Erhaltung von funktionalen Abhängigkeiten

Definition: Die obige Zerlegung **erhält die FAen** F
gdw. gilt: $F_1 \cup F_2 \models F$
d.h. für alle Relationen r zu $R(A)$ gilt:
$\pi_{B_i}(r)$ erfüllt F_i $(i = 1, 2) \Rightarrow r$ erfüllt F

Beispiele: verlustlose und FA-erhaltende Zerlegungen

- SSP(Stadt, StrasseNr, Plz),
  $\{$ Stadt StrasseNr $\rightarrow$ Plz, Plz $\rightarrow$ Stadt $\}$

  SS(Stadt,StrasseNr), $\emptyset$
  SP(StrasseNr,Plz), $\emptyset$

  nicht verlustlos: weder StrasseNr $\rightarrow$ Stadt
  noch StrasseNr $\rightarrow$ Plz
  $B_1 \cap B_2 = \{$ StrasseNr $\}$
  $B_1 - B_2 = \{$ Stadt $\}$
  $B_2 - B_1 = \{$ Plz $\}$

  nicht FA-erhaltend

- LIEFERANT(LName,LAdr,Ware,Preis)
  $\{$ LName $\rightarrow$ LAdr, LName Ware $\rightarrow$ Preis $\}$

  LIEFERANT'(LName,LAdr), $\{$ LName $\rightarrow$ LAdr $\}$
  ANGEBOT(LName,Ware,Preis),
  $\{$ LName Ware $\rightarrow$ Preis $\}$

  verlustlos: LName $\rightarrow$ LAdr
  $B_1 \cap B_2 = \{$ LName $\}$
  $B_1 - B_2 = \{$ LAdr $\}$
  $B_2 - B_1 = \{$ Ware, Preis $\}$

  FA-erhaltend

- STUD(Matnr, Loginid, Name)
  $\{$ Matnr $\rightarrow$ Name, Loginid $\rightarrow$ Name $\}$

  ML(Matnr, Loginid), $\emptyset$
  LN(Loginid, Name), $\{$ Loginid $\rightarrow$ Name $\}$

  verlustlos: Loginid $\rightarrow$ Name
  $B_1 \cap B_2 = \{$ Loginid $\}$
  $B_1 - B_2 = \{$ Matnr $\}$
  $B_2 - B_1 = \{$ Name $\}$

  nicht FA-erhaltend

- STUD(Loginid, Name, Abschluss, MinSemAnz)
  $\{$ Loginid $\rightarrow$ Name, Abschluss $\rightarrow$ MinSemAnz $\}$

  LNM(Loginid, Name, MinSemAnz),
  $\{$ Loginid $\rightarrow$ Name $\}$
  AM(Abschluss, MinSemAnz),
  $\{$ Abschluss $\rightarrow$ MinSemAnz $\}$

  nicht verlustlos:
  weder MinSemAnz $\rightarrow$ Loginid Name noch
  MinSemAnz $\rightarrow$ Abschluss
  $B_1 \cap B_2 = \{$ MinSemAnz $\}$
  $B_1 - B_2 = \{$ Loginid, Name $\}$
  $B_2 - B_1 = \{$ Abschluss $\}$

  FA-erhaltend

Normalisierung in 3NF

Satz: Seien $R(A)$ ein Relationenschema und F eine Menge von FAen über A; dann gibt es immer eine Zerlegung $(R_i(B_i), F_i)_{i=1}^m$ von $(R(A), F)$, so daß gilt:

(1) Die Zerlegung ist verlustlos

(2) Die FAen werden erhalten

(3) Jedes Relationenschema $R_i(B_i)$ ist in 3NF bzgl. F_i

Synthese-Verfahren für 3NF-Schemata

0. Berechne eine minimale Überdeckung G von F
   keine FA in G folgt aus anderen FAen in G;
   in jeder FA $X \to Y$ enthält X keine überflüssigen Attribute und Y ist einelementige Menge;
   für alle FAen $X \to Y$ gilt: $F \models X \to Y \Leftrightarrow G \models X \to Y$, d.h. $F^+ = G^+$

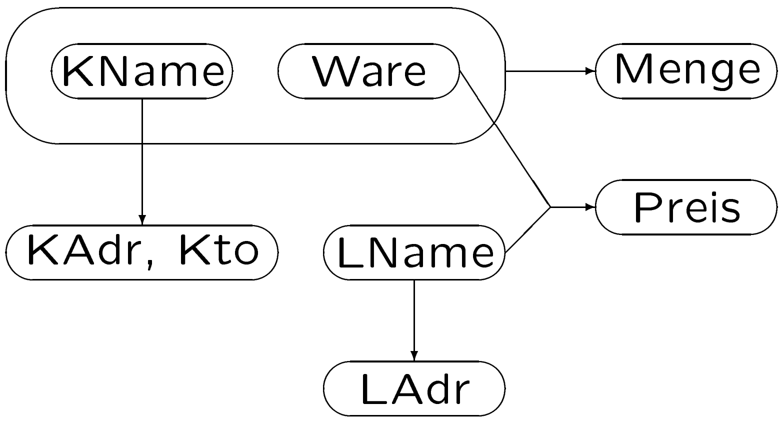
1. Falls $X \to Y \in G$ mit $X \cup Y = A$, so ist $R(A)$ in 3NF bzgl. F

2. Sonst gruppiere die FAen aus G nach gleicher linker Seite und definiere zu jeder Gruppe $F_i := \{D \to C_1, D \to C_2, ..., D \to C_r\}$ ein Relationen-Schema $R_i(B_i)$ mit $B_i = D \cup C_1 \cup ... \cup C_r$; $R_i(B_i)$ ist in 3NF bzgl. F_i

3. Falls noch kein Schlüssel von $R(A)$ in einem der $R_i(B_i)$ enthalten ist, füge einen Schlüssel X von $R(A)$ als weiteres Schema $R'(X)$ hinzu

Beispiel: Synthese-Verfahren

0. Minimale Überdeckung



KName → KAdr, KName → Kto
KName Ware → Menge
LName → LAdr
LName Ware → Preis

1. Schritt nicht anwendbar

2. KUNDE(KName, KAdr, Kto)
   AUFTRAG(KName, Ware, Menge)
   LIEFERANT'(LName, LAdr)
   ANGEBOT(LName, Ware, Preis)

3. Schlüssel: KName, Ware, LName
   BEZUG (KName, Ware, LName)

Normalisierung in BCNF: Zerlegungs-Verfahren

Eingabe: $S = \langle (R(A), F) \rangle$

Ausgabe: $T = \langle (R_i(B_i), F_i) \mid i = 1, \cdots m \rangle$ in BCNF (3NF), verlustlos; jedoch nicht immer FA–erhaltend (0)

1. $T := \{S\}$

2. solange T nicht in BCNF [3NF]: wähle Teilschema $(Q(C), G) \in T$, das nicht in BCNF [3NF] ist, und eine Determinante X eines Attributes [NSA] Y, die kein Schlüssel ist (1)

   d.h. insbesondere $X \to Y \in G^*$ und $X \cap Y = \emptyset$

   $$T := T - \{\ (Q(C), G)\ \} \cup \{\ (Q_1(X \cup Y), G(X \cup Y))\ \} \cup \{\ (Q_2(C - Y), G(C - Y))\ \} \qquad (2/3)$$

Anmerkungen:

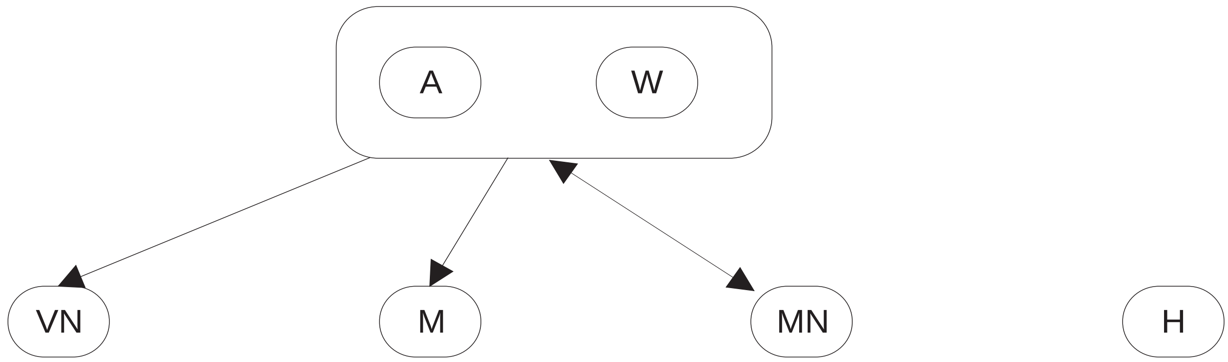
(0) Für nur 3NF gibt es auch eine FA–erhaltende Variante des Verfahrens

(1) Determinanten sind aus einer minimalen Überdeckung von G ablesbar

(2) Anwendung des Zerlegungssatzes

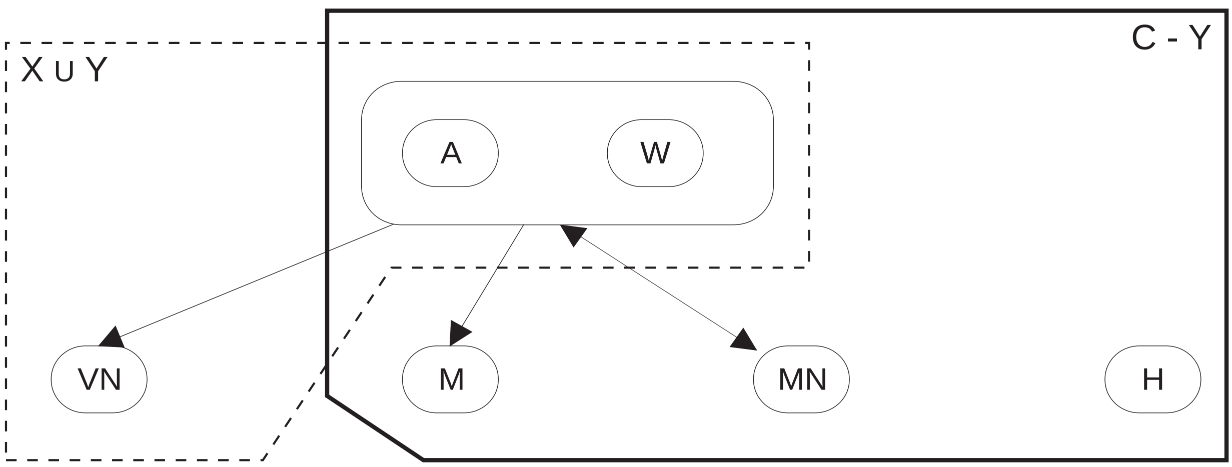
(3) Mehr FAen werden erhalten, wenn eine minimale Überdeckung der Projektion von G^* auf $X \cup Y$ gebildet wird; also statt $G(X \cup Y)$ besser $MinCover(\pi_{X \cup Y}(G^*))$; für $C - Y$ analog

Beispiel: Zerlegungs-Verfahren



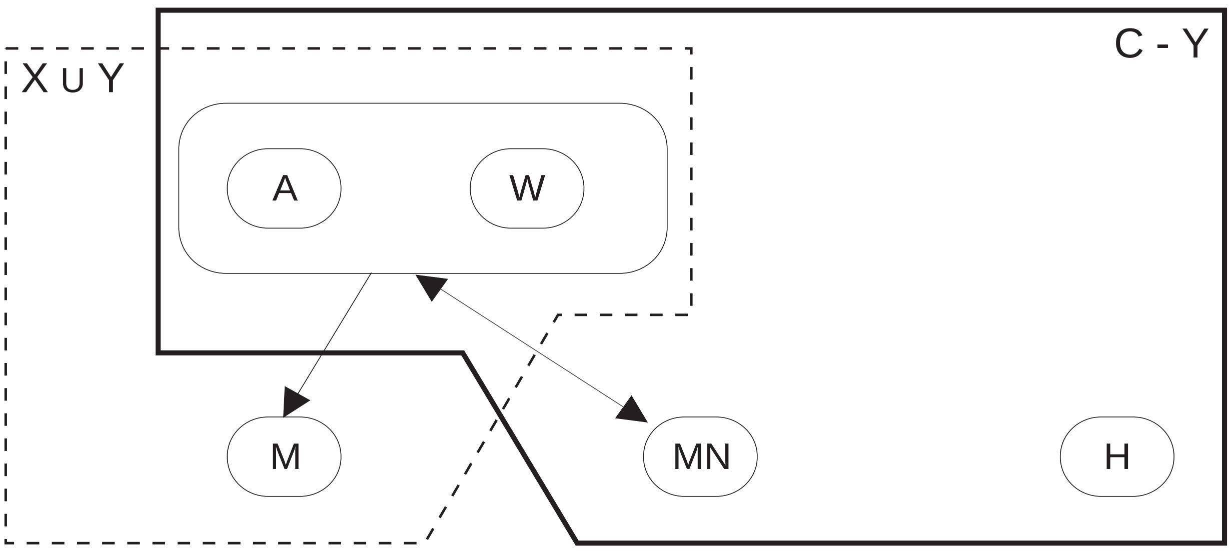
Schlüsselkandidaten: { A, W, H }, { MN, H }

mittels 'A W $\rightarrow$ VN' zerlegen



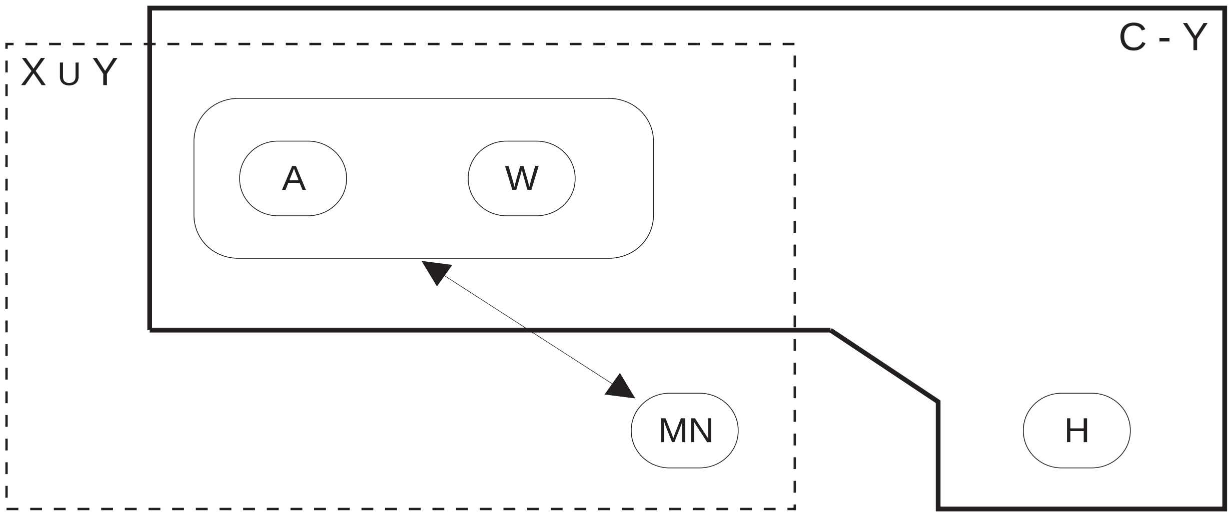
entstehendes Relationenschema:
VERMIETER(<u>A</u>, <u>W</u>, VN)

Rest C-Y (fett umrandet) mittels 'A W $\rightarrow$ M' zerlegen



entstehendes Relationenschema: MIETEN(<u>A</u>, <u>W</u>, M)

Rest C-Y (fett umrandet) mittels 'A W $\rightarrow$ MN' zerlegen



entstehendes Relationenschema:
MIETER(<u>A</u>, <u>W</u>, MN) oder MIETER(A, W, <u>MN</u>)

entstehendes Relationenschema: TIERE(<u>A</u>, <u>W</u>, <u>H</u>)

nicht minimale Anzahl von Relationenschemata:
VERMIETER und MIETEN zusammenfassbar

Beispiel: Zerlegungs-Verfahren

- SSP(Stadt, StrasseNr, Plz)

  Stadt StrasseNr $\rightarrow$ Plz

  Plz $\rightarrow$ Stadt

  Schlüsselkandidaten: - { Stadt, StrasseNr }
- { Plz, StrasseNr }

- einzig mögliche Wahl für X und Y: Plz $\rightarrow$ Stadt

- entstehende Relationenschemata:

  PS(<u>Plz</u>, Stadt)

  SP(<u>StrasseNr</u>, <u>Plz</u>)

- in BNCF, aber FA 'Stadt StrasseNr $\rightarrow$ Plz' verlorengegangen

- **BCNF und FA-Erhaltung sind unvereinbar**

Motivation 4NF

SFSBL	Schulfach	Schulbuch	Lehrer
	Mathe	BooleFunctions	Meier
	Mathe	RusselLogic	Meier
	Mathe	BooleFunctions	Schmid
	Mathe	RusselLogic	Schmid
	Philo	RusselLogic	Meier

Im obigen Zustand ist keine der 9 möglichen nicht-trivialen FAen gültig:
z.B. gilt nicht: Schulfach $\rightarrow$ Schulbuch wegen (Mathe,BooleFunctions,...) und (Mathe,RusselLogic,...)
z.B. gilt nicht: Schulfach Schulbuch $\rightarrow$ Lehrer wegen (Mathe,BooleFunctions,Meier) und (Mathe,BooleFunctions,Schmid)

Für obiges Relationschema also Menge der FAen leer; Schlüssel besteht damit aus Gesamtattributmenge; damit in BCNF; dennoch Redundanz

SFSB	Schulfach	Schulbuch
	Mathe	BooleFunctions
	Mathe	RusselLogic
	Philo	RusselLogic

SFL	Schulfach	Lehrer
	Mathe	Meier
	Mathe	Schmid
	Philo	Meier

Es gelten mehrwertige Abhängigkeiten:
Schulfach $\twoheadrightarrow$ Schulbuch, Schulfach $\twoheadrightarrow$ Lehrer

Normalisierung in 4NF

4NF: BCNF und es gibt keine mehrwertige Abhängigkeit (MA) $X \twoheadrightarrow Y$

Idee der MAen: Jeder X-Wert bestimmt eine Menge von Y-Werten unabhängig vom Kontext, d.h. unabhängig von den weiteren Attributen

Beispiel: Schulfach $\twoheadrightarrow$ Schulbuch unabhängig vom Attribut Lehrer, Schulfach $\twoheadrightarrow$ Lehrer unabhängig vom Attribut Schulbuch

Definition: Es seien X, Y disjunkt und es gelte $X \cup Y \cup Z = A$ in $R(A)$ mit $Z \neq \emptyset$ und Z disjunkt zu $X \cup Y$; eine MA $X \twoheadrightarrow Y$ ist gültig gdw. für jede Relation r gilt: $r = \pi_{XY}(r) * \pi_{XZ}(r)$

Bermerkung: Zur 4NF gibt es Zerlegungs-Verfahren, die Verlustlosigkeit und (für konfliktfreie MAen) auch Erhaltung der Abhängigkeiten garantieren

... und ein Artikel von Kent heisst:
“A Simple Guide to **Five** Normal Forms ... ”