Kapitel DB:VII (Fortsetzung)

VII. Entwurfstheorie relationaler Datenbanken

Dekompositionseigenschaften von Relationen

Herangehensweisen zum Datenbankentwurf

Top-Down:

1. Anforderungsanalyse
2. Erstellung eines konzeptuellen Schemas/Modells in einem „High-Level-Modell“, zum Beispiel im EER-Modell.
3. Abbildung des konzeptuellen Modells auf eine Menge von Relationen.
4. Verfeinerung des relationalen Modells.

Bottom-Up:

1. Festlegung einer Menge funktionaler Abhängigkeiten
2. Synthetisierung von Relationenschemata für die bestimmte formale Eigenschaften garantiert werden können.

Bemerkungen:

- Die Top-Down-Herangehensweise wird auch als “Design by Analysis” bezeichnet.
- Die Bottom-Up-Herangehensweise als “Design by Synthesis” bezeichnet.

Dekompositionseigenschaften von Relationen

Unterscheidung des formalen Instrumentariums

1. Eigenschaften einzelner Relationen – insbesondere:
   (a) Einhaltung einer bestimmten Normalform

2. Dekompositionseigenschaften – insbesondere:
   (a) Abhängigkeitserhaltung *(Dependency Preservation)*
   (b) verlustlose Zerlegung bzw. Verlustlosigkeit *(Lossless Join Property)*

Bemerkungen:

- Die Theorie der Normalformen ist im Zusammenhang mit beiden Herangehensweisen zum Datenbankentwurf (Top-Down, Bottom-Up) nützlich.
- Wiederholung: Die Herstellung einer (hohen) Normalform ist kein hinreichendes Kriterium für ein gutes Datenbankdesign.
- In [Heuer/Saake 2013] werden
  - die Dekompositionseigenschaften als Transformationseigenschaften,
  - die Abhängigkeitserhaltung als Abhängigkeitstreue und
  - die verlustlose Zerlegung als Verbundtreue bezeichnet.

Dekompositionseigenschaften von Relationen

Universalrelation und Dekomposition

Definition 9 (Universalrelation)

Die Universalrelation – genauer: das Universalrelationenschema – zu einer Datenbank entsteht durch Zusammenfassung aller in der Datenbank vorhandenen Attribute in einem einzigen relationalen Schema $\mathcal{R}$.

Definition 10 (Dekomposition eines Relationenschemas)

Sei $\mathcal{R}$ ein Relationenschema. Die Aufteilung von $\mathcal{R}$ in eine Menge $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ von Relationenschemata heißt Dekomposition oder Zerlegung von $\mathcal{R}$.

Attributerhaltung:

Die Eigenschaft der Attributerhaltung einer Dekomposition $\boldsymbol{\mathcal{R}}$ von $\mathcal{R}$ fordert, dass jedes Attribut von $\mathcal{R}$ in mindestens einem Relationenschema $\mathcal{R}_i$, $\mathcal{R}_i \in \boldsymbol{\mathcal{R}}$, auftaucht:

$$\bigcup_{i=1}^{m} \mathcal{R}_i = \mathcal{R}$$

Dekompositionseigenschaften von Relationen

(a) Abhängigkeitserhaltung

Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1, \ldots, \mathcal{R}_m$ sollte so sein, dass die Überprüfung jeder FD $(\alpha \to \beta) \in F$ *lokal* auf den $\mathcal{R}_i$ erfolgen kann. Man muss also keinen Join durchführen, um dann – auf der verbundenen Relation – die Abhängigkeiten prüfen zu können.

Das heißt, jede FD $(\alpha \to \beta) \in F$

- kommt entweder direkt in einem $\mathcal{R}_i$ vor oder
- kann mit Hilfe der Inferenzregeln abgeleitet werden.

Dekompositionseigenschaften von Relationen

(a) Abhängigkeitserhaltung

Definition 11 (Einschränkung von FDs, Abhängigkeitserhaltung)

Sei $\mathcal{R}$ ein Relationenschema mit zugehörigen funktionalen Abhängigkeiten F und sei $\mathcal{R}_i \subseteq \mathcal{R}$.

Die Einschränkung von F auf $\mathcal{R}_i$, in Zeichen: $F_{\mathcal{R}_i}$, ist die Menge von Abhängigkeiten $(\alpha \to \beta) \in F^+$ für die $(\alpha \cup \beta) \subseteq \mathcal{R}_i$ gilt.

Eine Dekomposition $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ von $\mathcal{R}$ ist abhängigkeitserhaltend hinsichtlich F, wenn gilt:

$$F \equiv F_{\mathcal{R}_1} \cup \ldots \cup F_{\mathcal{R}_m} \quad \text{bzw.} \quad F^+ = (F_{\mathcal{R}_1} \cup \ldots \cup F_{\mathcal{R}_m})^+$$

Dekompositionseigenschaften von Relationen

(a) Abhängigkeitserhaltung

Definition 11 (Einschränkung von FDs, Abhängigkeitserhaltung)

Sei $\mathcal{R}$ ein Relationenschema mit zugehörigen funktionalen Abhängigkeiten F und sei $\mathcal{R}_i \subseteq \mathcal{R}$.

Die Einschränkung von F auf $\mathcal{R}_i$, in Zeichen: $F_{\mathcal{R}_i}$, ist die Menge von Abhängigkeiten $(\alpha \rightarrow \beta) \in F^+$ für die $(\alpha \cup \beta) \subseteq \mathcal{R}_i$ gilt.

Eine Dekomposition $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ von $\mathcal{R}$ ist abhängigkeitserhaltend hinsichtlich F, wenn gilt:

$$F \equiv F_{\mathcal{R}_1} \cup \ldots \cup F_{\mathcal{R}_m} \quad \text{bzw.} \quad F^+ = (F_{\mathcal{R}_1} \cup \ldots \cup F_{\mathcal{R}_m})^+$$

Bemerkungen:

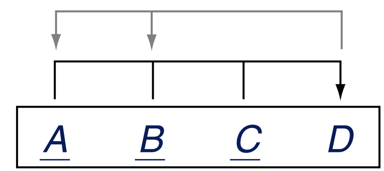
- Abhängigkeitserhaltung garantiert die effiziente Überprüfung von funktionalen Abhängigkeiten (nach dem Hinzufügen, Löschen oder Ändern von Tupeln), weil die Relationenschemata unabhängig voneinander analysiert werden können.
- Eine abhängigkeitserhaltende Dekomposition wird auch *hüllentreue* Dekomposition genannt.
- Abhängigkeitserhaltung erfordert, dass bei der Dekomposition eines Relationenschemas keine linke Seite einer funktionalen Abhängigkeit zerschnitten wird.
- Für ein Relationenschema $\mathcal{R}$ kann immer eine abhängigkeitserhaltende Dekomposition $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ gefunden werden, so dass jedes $\mathcal{R}_i$, $\mathcal{R}_i \in \boldsymbol{\mathcal{R}}$, in 3NF ist.
  Es kann nicht immer eine Dekomposition gefunden werden, die alle Abhängigkeiten erhält und bei der jedes $\mathcal{R}_i$ in BCNF ist.

Dekompositionseigenschaften von Relationen

(a) Abhängigkeitserhaltung: Beispiel

PLZverzeichnis			
Ort	BLand	Strasse	PLZ
Frankfurt	Hessen	Goethestrasse	60313
Frankfurt	Hessen	Schillerstrasse	60437
Frankfurt	Brandenburg	Goethestrasse	15234

$\{PLZ\} \rightarrow \{Ort, BLand\}$
$\{Ort, BLand, Strasse\} \rightarrow \{PLZ\}$



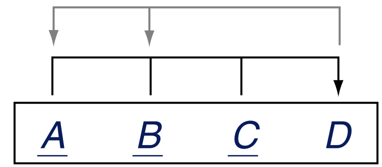
Dekompositionseigenschaften von Relationen

(a) Abhängigkeitserhaltung: Beispiel

PLZverzeichnis			
Ort	BLand	Strasse	PLZ
Frankfurt	Hessen	Goethestrasse	60313
Frankfurt	Hessen	Schillerstrasse	60437
Frankfurt	Brandenburg	Goethestrasse	15234

$\{\text{PLZ}\} \rightarrow \{\text{Ort}, \text{BLand}\}$

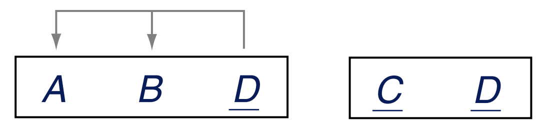
$\{\text{Ort}, \text{BLand}, \text{Strasse}\} \rightarrow \{\text{PLZ}\}$



Durch folgende Zerlegung geht die FD $\{\text{Ort}, \text{BLand}, \text{Strasse}\} \rightarrow \{\text{PLZ}\}$ verloren:

Orte		
Ort	BLand	PLZ
Frankfurt	Hessen	60313
Frankfurt	Hessen	60437
Frankfurt	Brandenburg	15234

Strassen	
Strasse	PLZ
Goethestrasse	60313
Schillerstrasse	60473
Goethestrasse	15234



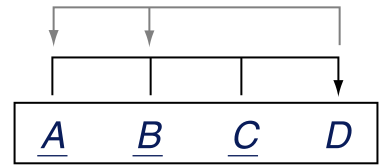
Dekompositionseigenschaften von Relationen

(a) Abhängigkeitserhaltung: Beispiel

PLZverzeichnis			
Ort	BLand	Strasse	PLZ
Frankfurt	Hessen	Goethestrasse	60313
Frankfurt	Hessen	Schillerstrasse	60437
Frankfurt	Brandenburg	Goethestrasse	15234

$\{\text{PLZ}\} \rightarrow \{\text{Ort}, \text{BLand}\}$

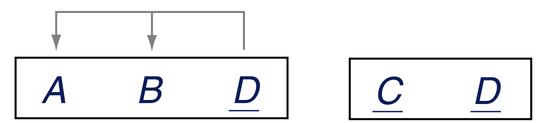
$\{\text{Ort}, \text{BLand}, \text{Strasse}\} \rightarrow \{\text{PLZ}\}$



Durch folgende Zerlegung geht die FD $\{\text{Ort}, \text{BLand}, \text{Strasse}\} \rightarrow \{\text{PLZ}\}$ verloren:

Orte		
Ort	BLand	PLZ
Frankfurt	Hessen	60313
Frankfurt	Hessen	60437
Frankfurt	Brandenburg	15234

Strassen	
Strasse	PLZ
Goethestrasse	60313
Schillerstrasse	60473
Goethestrasse	15234



Lokal konsistentes Update der Relationen:

Orte		
Ort	BLand	PLZ
Frankfurt	Hessen	60313
Frankfurt	Hessen	60437
Frankfurt	Brandenburg	15234
Frankfurt	Brandenburg	15235

Strassen	
Strasse	PLZ
Goethestrasse	60313
Schillerstrasse	60473
Goethestrasse	15234
Goethestrasse	15235

Bemerkungen:

- Das Schema PLZverzeichnis ist in 3NF, aber nicht in BCNF, denn es existiert die Abhängigkeit $\{PLZ\} \to \{Ort, BLand\}$ und $\{PLZ\}$ ist nicht Superschlüssel.
- Durch Elimination der abhängigen Attributmenge $\{Ort, BLand\}$ und Erstellung eines weiteren Schemas bestehend aus $\{Ort, BLand, PLZ\}$ wird die BCNF hergestellt.
- Die Zerlegung ist nicht abhängigkeitserhaltend. Die Verletzung der FD $\{Ort, BLand, Strasse\} \to \{PLZ\}$ ist erst nach einem Join (Join-Attribut ist „PLZ“) erkennbar. Obwohl $\{Ort, BLand, Strasse\}$ Schlüssel sein sollte, existieren nach einem Join u.a. folgende Tupel:

  (Frankfurt, Brandenburg, Goethestrasse, 15234)

  $\neq$

  (Frankfurt, Brandenburg, Goethestrasse, 15235)

- Das Relationenschema „Strassen“ enthält nur noch triviale Abhängigkeiten; der Schlüssel von „Strassen“ besteht deshalb aus der Menge aller Attribute des Schemas.
- Vorgriff: Weil $\{PLZ\}$ einen Schlüssel in einer der beiden neuen Relationen darstellt, ist die Zerlegung verlustlos.

Dekompositionseigenschaften von Relationen

(b) Verlustlose Zerlegung bzw. Verbundtreue

Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1, \ldots, \mathcal{R}_m$ sollte so sein, dass die in einer Ausprägung r des Relationenschemas $\mathcal{R}$ enthaltene Information aus den Ausprägungen $r_1, \ldots, r_m$ der Relationenschemata $\mathcal{R}_1, \ldots, \mathcal{R}_m$ konstruierbar ist.

Definition 12 (verlustlose Zerlegung)

Sei $\mathcal{R}$ ein Relationenschema mit zugehörigen funktionalen Abhängigkeiten F. Eine Zerlegung $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ von $\mathcal{R}$ ist verlustlos bzw. verbundtreu hinsichtlich F, wenn für jede Ausprägung r des Relationenschemas $\mathcal{R}$, die F erfüllt, gilt:

$$r = \pi_{\mathcal{R}_1}(r) \bowtie \ldots \bowtie \pi_{\mathcal{R}_m}(r)$$

Bemerkungen:

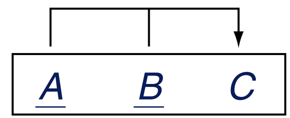
- Eine Zerlegung ist dann nicht verlustlos, wenn nach dem Join zusätzliche *(spurious)* Tupel entstanden sind. Zusätzliche Tupel bedeuten einen Informationsverlust, weil durch sie eindeutige Zuordnungen verloren gegangen sind.

- Oft ist es ausreichend, sich bei der Dekomposition von $\mathcal{R}$ auf den binären Fall zu beschränken. Es gilt nämlich: Ist $\{\mathcal{R}_1, \ldots, \mathcal{R}_m\}$ eine verlustlose Zerlegung von $\mathcal{R}$ hinsichtlich F und ist $\{\mathcal{R}_{i_1}, \ldots, \mathcal{R}_{i_k}\}$ eine verlustlose Zerlegung von $\mathcal{R}_i$ hinsichtlich $F_{\mathcal{R}_i}$, so ist auch $\{\mathcal{R}_1, \ldots, \mathcal{R}_{i-1}, \mathcal{R}_{i_1}, \ldots, \mathcal{R}_{i_k}, \mathcal{R}_{i+1}, \ldots, \mathcal{R}_m\}$ eine verlustlose Zerlegung von $\mathcal{R}$ hinsichtlich F.

Dekompositionseigenschaften von Relationen

(b) Verlustlose Zerlegung: Beispiel

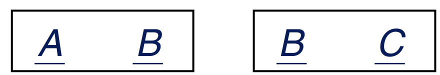
Biertrinker		
Kneipe	Gast	Bier
Kowalski	Kemper	Pils
Kowalski	Eickler	Hefeweizen
Innsteg	Kemper	Hefeweizen



Zerlegung in die Relationenschemata „Besucht“ und „Trinkt“ :

Besucht	
Kneipe	Gast
Kowalski	Kemper
Kowalski	Eickler
Innsteg	Kemper

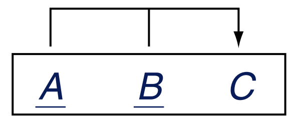
Trinkt	
Gast	Bier
Kemper	Pils
Eickler	Hefeweizen
Kemper	Hefeweizen



Dekompositionseigenschaften von Relationen

(b) Verlustlose Zerlegung: Beispiel

Biertrinker		
Kneipe	Gast	Bier
Kowalski	Kemper	Pils
Kowalski	Eickler	Hefeweizen
Innsteg	Kemper	Hefeweizen



Zerlegung in die Relationenschemata „Besucht“ und „Trinkt“ :

Besucht	
Kneipe	Gast
Kowalski	Kemper
Kowalski	Eickler
Innsteg	Kemper

Trinkt	
Gast	Bier
Kemper	Pils
Eickler	Hefeweizen
Kemper	Hefeweizen

$\underline{A}$ $\underline{B}$ $\underline{B}$ $\underline{C}$

Die Bildung des natürlichen Verbundes zeigt, dass die Assoziation von Biersorten und Gästen relativ zur Kneipe verloren gegangen ist:

Besucht	
Kneipe	Gast
Kowalski	Kemper
Kowalski	Eickler
Innsteg	Kemper

$\bowtie$

Trinkt	
Gast	Bier
Kemper	Pils
Eickler	Hefeweizen
Kemper	Hefeweizen

$\rightsquigarrow$

Besucht $\bowtie$ Trinkt			
Kneipe	Gast	Bier	
Kowalski	Kemper	Pils	
Kowalski	Kemper	Hefeweizen	$\star$
Kowalski	Eickler	Hefeweizen	
Innsteg	Kemper	Pils	$\star$
Innsteg	Kemper	Hefeweizen	

Bemerkungen:

- Im Beispiel gilt nur die FD $\{\text{Kneipe}, \text{Gast}\} \rightarrow \{\text{Bier}\}$.
- Die Existenz von einer der folgenden FDs würde Verlustlosigkeit garantieren:
  $\{\text{Gast}\} \rightarrow \{\text{Bier}\}$, $\{\text{Gast}\} \rightarrow \{\text{Kneipe}\}$

Dekompositionseigenschaften von Relationen

(b) Verlustlose Zerlegung (Fortsetzung)

Formulierung 1:

Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1$ und $\mathcal{R}_2$ ist verlustlos, wenn mindestens eine der folgenden Bedingungen gilt:

- $((\mathcal{R}_1 \cap \mathcal{R}_2) \rightarrow \mathcal{R}_1) \in F^+$
- $((\mathcal{R}_1 \cap \mathcal{R}_2) \rightarrow \mathcal{R}_2) \in F^+$

Dekompositionseigenschaften von Relationen

(b) Verlustlose Zerlegung (Fortsetzung)

Formulierung 1:

Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1$ und $\mathcal{R}_2$ ist verlustlos, wenn mindestens eine der folgenden Bedingungen gilt:

- $((\mathcal{R}_1 \cap \mathcal{R}_2) \to \mathcal{R}_1) \in F^+$
- $((\mathcal{R}_1 \cap \mathcal{R}_2) \to \mathcal{R}_2) \in F^+$

Formulierung 2:

Sei $\mathcal{R} = \alpha \cup \beta \cup \gamma$, $\mathcal{R}_1 = \alpha \cup \beta$, und $\mathcal{R}_2 = \alpha \cup \gamma$ mit paarweisen disjunkten Attributmengen α, β und γ. Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1$ und $\mathcal{R}_2$ ist verlustlos, wenn mindestens eine der folgenden Bedingungen gilt:

- $\beta \subseteq \textit{AttributeClosure}(F, \alpha)$
- $\gamma \subseteq \textit{AttributeClosure}(F, \alpha)$

Bemerkungen:

- Die Attributmenge im Schnitt der beiden Relationenschemata, $\mathcal{R}_1 \cap \mathcal{R}_2$, bestimmt mindestens *eines* der beiden Relationenschemata funktional, ist also (Super-)Schlüssel für eines der beiden Relationenschemata.

  Das macht auch den Zusammenhang zur Verlustlosigkeit klar: Zu jeder Schlüsselausprägung gibt es höchstens *ein* Tupel, und somit besteht keine Möglichkeit, bei einem Join zusätzliche (= falsche) Tupel dazu zu kombinieren.

Relationale Dekomposition

Algorithm: RelDecomposition
Input: $\mathcal{R}$. Universalrelation.
F. Menge funktionaler Abhängigkeiten für $\mathcal{R}$.
Output: $\boldsymbol{\mathcal{R}}$. Verlustlose Zerlegung von $\mathcal{R}$ mit Schemata in BCNF.

1. $\boldsymbol{\mathcal{R}} = \{\mathcal{R}\}$
2. **WHILE** $\exists \mathcal{R}' : (\mathcal{R}' \in \boldsymbol{\mathcal{R}} \;\wedge\; \mathcal{R}'$ not in BCNF$)$ **DO**
3. Find FD $(\alpha \to \beta) \in F_{\mathcal{R}'}$ that violates BCNF
4. Decompose $\mathcal{R}'$ into $\mathcal{R}'_1 = \mathcal{R}' - \beta$ and $\mathcal{R}'_2 = \alpha \cup \beta$
5. $\boldsymbol{\mathcal{R}} = (\boldsymbol{\mathcal{R}} - \{\mathcal{R}'\}) \;\cup\; \{\mathcal{R}'_1, \mathcal{R}'_2\}$
6. **ENDDO**
7. **RETURN**$(\boldsymbol{\mathcal{R}})$

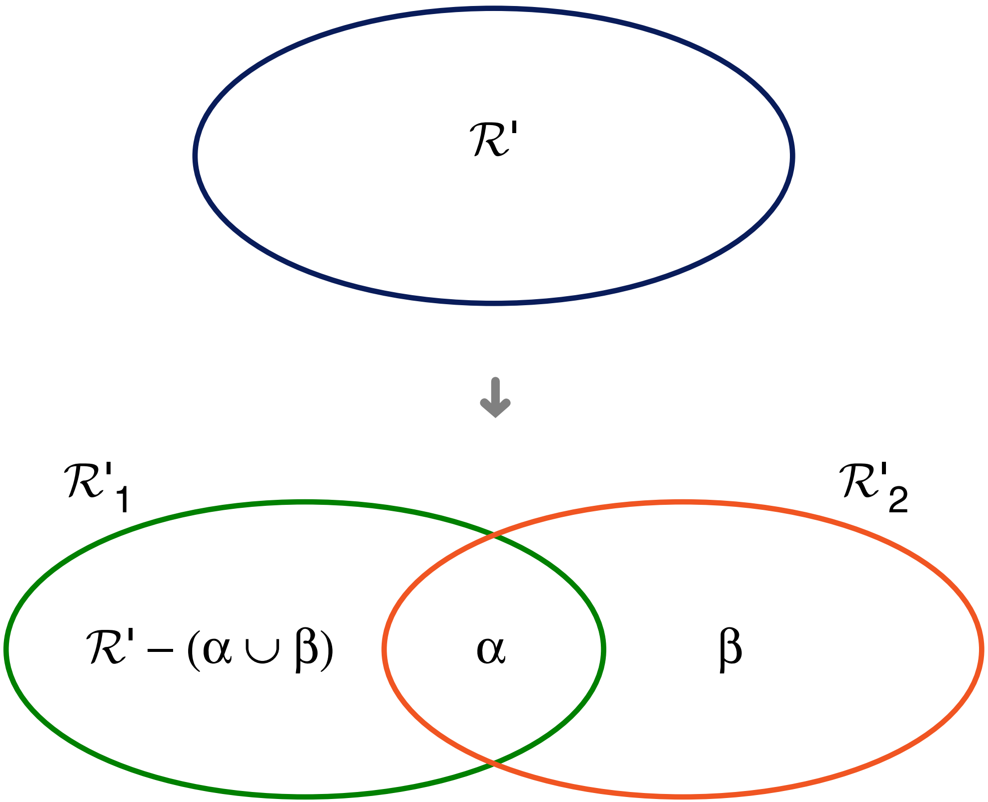
Vergleiche Schritt 5 mit der Illustration der Boyce-Codd-Normalform.

Bemerkungen:

- Schritt 3. Wie man eine BCNF-verletzende FD in der While-Schleife findet: Ist ein Relationenschema $\mathcal{R}'$ nicht in BCNF, so existiert in $\mathcal{R}'$ eine FD $\alpha \to \beta$ mit $\alpha \cap \beta = \emptyset$ und $\alpha \not\to \mathcal{R}'$.
- Schritt 4. Die Dekomposition garantiert Verlustlosigkeit: $\mathcal{R}_1' \cap \mathcal{R}_2' = \alpha$ mit $\alpha \to \mathcal{R}_2'$
- Eine durch den Algorithmus RelDecomposition erzeugte Dekomposition ist nicht notwendigerweise abhängigkeitserhaltend.

Relationale Dekomposition

Illustration der Zerlegung eines Relationenschemas $\mathcal{R}'$ in die Schemata $\mathcal{R}'_1$ und $\mathcal{R}'_2$ entlang der funktionalen Abhängigkeit $\alpha \to \beta$:



[Kemper/Eickler 2011]

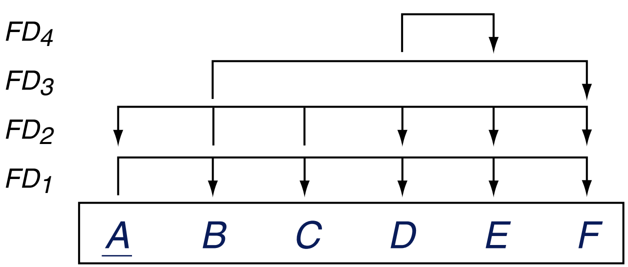
Relationale Dekomposition

Beispiel:

- $\mathcal{R}_{\text{Grundstuecke}} = \{\underline{\text{SteuerNr}}, \text{Landkreis}, \text{GrundstNr}, \text{GrundstGroesse}, \text{Preis}, \text{Steuersatz}\}$

- FD_1: $\{\text{SteuerNr}\} \rightarrow \{\text{Landkreis}, \text{GrundstNr}, \text{GrundstGroesse}, \text{Preis}, \text{Steuersatz}\}$
- FD_2: $\{\text{Landkreis}, \text{GrundstNr}\} \rightarrow \{\text{SteuerNr}, \text{GrundstGroesse}, \text{Preis}, \text{Steuersatz}\}$
- FD_3: $\{\text{Landkreis}\} \rightarrow \{\text{Steuersatz}\}$
- FD_4: $\{\text{GrundstGroesse}\} \rightarrow \{\text{Preis}\}$

Grundstuecke					
<u>SteuerNr</u>	Landkreis	GrundstNr	GrundstGroesse	Preis	Steuersatz



$\leadsto$ $\mathcal{TAFEL}$

Relationale Synthese

Algorithm: RelSynthesis

Input: $\mathcal{R}$. Universalrelation.
F. Menge funktionaler Abhängigkeiten für $\mathcal{R}$.

Output: $\boldsymbol{\mathcal{R}}$. Verlustlose und *abhängigkeitserhaltende* Zerlegung von $\mathcal{R}$ mit Schemata in 3NF.

$\boldsymbol{\mathcal{R}} = \emptyset$

Determine a canonical cover F_c of F

RETURN$(\boldsymbol{\mathcal{R}})$

Relationale Synthese

Algorithm: RelSynthesis
Input: $\mathcal{R}$. Universalrelation.
F. Menge funktionaler Abhängigkeiten für $\mathcal{R}$.
Output: $\boldsymbol{\mathcal{R}}$. Verlustlose und *abhängigkeitserhaltende* Zerlegung von $\mathcal{R}$ mit Schemata in 3NF.

$\boldsymbol{\mathcal{R}} = \emptyset$
Determine a canonical cover F_c of F
FOREACH $(\alpha \to \beta) \in F_c$ **DO**
  Synthesize $\mathcal{R}_\alpha = \alpha \cup \beta$, $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{R}} \cup \{\mathcal{R}_\alpha\}$
  $F_\alpha = \{(\gamma \to \delta) \in F_c \mid (\gamma \cup \delta) \subseteq \mathcal{R}_\alpha\}$
ENDDO
RETURN$(\boldsymbol{\mathcal{R}})$

Relationale Synthese

Algorithm: RelSynthesis

Input: $\mathcal{R}$. Universalrelation.
F. Menge funktionaler Abhängigkeiten für $\mathcal{R}$.

Output: $\boldsymbol{\mathcal{R}}$. Verlustlose und *abhängigkeitserhaltende* Zerlegung von $\mathcal{R}$ mit Schemata in 3NF.

$\boldsymbol{\mathcal{R}} = \emptyset$
Determine a canonical cover F_c of F
FOREACH $(\alpha \to \beta) \in F_c$ **DO**
  Synthesize $\mathcal{R}_\alpha = \alpha \cup \beta$, $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{R}} \cup \{\mathcal{R}_\alpha\}$
  $F_\alpha = \{(\gamma \to \delta) \in F_c \mid (\gamma \cup \delta) \subseteq \mathcal{R}_\alpha\}$
ENDDO
IF $\nexists \alpha : ((\alpha \to \beta) \in F_c$ with $\alpha \to \mathcal{R})$ **THEN**
  Determine a candidate key $\kappa \subseteq \mathcal{R}$
  $\mathcal{R}_\kappa = \kappa$, $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{R}} \cup \{\mathcal{R}_\kappa\}$, $F_\kappa = \emptyset$
ENDIF

RETURN$(\boldsymbol{\mathcal{R}})$

Relationale Synthese

Algorithm: RelSynthesis

Input: $\mathcal{R}$. Universalrelation.
F. Menge funktionaler Abhängigkeiten für $\mathcal{R}$.

Output: $\boldsymbol{\mathcal{R}}$. Verlustlose und *abhängigkeitserhaltende* Zerlegung von $\mathcal{R}$ mit Schemata in 3NF.

1. $\boldsymbol{\mathcal{R}} = \emptyset$
2. Determine a canonical cover F_c of F
3. **FOREACH** $(\alpha \rightarrow \beta) \in F_c$ **DO**
4.   Synthesize $\mathcal{R}_\alpha = \alpha \cup \beta$, $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{R}} \cup \{\mathcal{R}_\alpha\}$
5. $F_\alpha = \{(\gamma \rightarrow \delta) \in F_c \mid (\gamma \cup \delta) \subseteq \mathcal{R}_\alpha\}$
6. **ENDDO**
7. **IF** $\nexists \alpha : ((\alpha \rightarrow \beta) \in F_c$ with $\alpha \rightarrow \mathcal{R})$ **THEN**
8.   Determine a candidate key $\kappa \subseteq \mathcal{R}$
9. $\mathcal{R}_\kappa = \kappa$, $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{R}} \cup \{\mathcal{R}_\kappa\}$, $F_\kappa = \emptyset$
10. **ENDIF**
11. **FOREACH** $\mathcal{R}_\alpha, \mathcal{R}_{\alpha'} \in \boldsymbol{\mathcal{R}}$ **DO** **IF** $\mathcal{R}_\alpha \subseteq \mathcal{R}_{\alpha'}$ **THEN** $\boldsymbol{\mathcal{R}} = \boldsymbol{\mathcal{R}} - \{\mathcal{R}_\alpha\}$
12. **RETURN**$(\boldsymbol{\mathcal{R}})$

Bemerkungen:

- Schritt 2. Bestimmung einer kanonischen Überdeckung für die Menge der funktionalen Abhängigkeiten.
- Schritt 3-6. Synthetisierung der Relationenschemata; diese befinden sich – per Konstruktion auf Basis der kanonischen Überdeckung – in 3NF.
- Schritt 7-10. Überprüfung, ob eines der generierten Relationenschemata einen Schlüssel für die Universalrelation enthält. Falls nicht, ist ein solcher Schlüssel zu bestimmen und ein aus den Schlüsselattributen bestehendes Relationenschema hinzuzunehmen.
- Schritt 11. Eliminierung von subsummierten Relationenschemata.

Bemerkungen: (Fortsetzung)

- Es stellt sich u.a. die Frage, wie wertvoll die Eigenschaft der Verlustlosigkeit ist [Heuer/Saake 2013] :
  - Muss sich die Universalrelation rekonstruieren lassen bzw. hat sie eine aus der Anwendung stammende zwingende Semantik?
  - Ist der natürliche Verbund als Rekonstruktionsoperator das Maß aller Dinge?

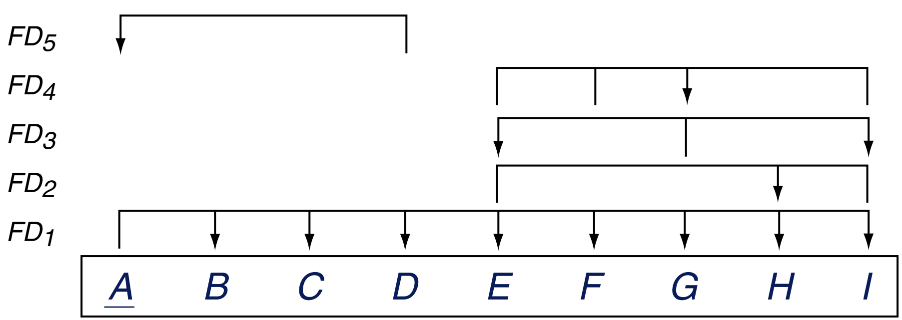
Relationale Synthese

Beispiel:

- $\mathcal{R}_{\text{MitarbeiterAdr}} = \{\underline{\text{PersNr}}, \text{Name}, \text{Gehaltsstufe}, \text{Raum}, \text{Ort}, \text{Strasse}, \text{PLZ}, \text{Vorwahl}, \text{BLand}\}$

- FD_1: $\{\text{PersNr}\} \to \{\text{Name}, \text{Gehaltsstufe}, \text{Raum}, \text{Ort}, \text{Strasse}, \text{PLZ}, \text{Vorwahl}, \text{BLand}\}$
- FD_2: $\{\text{Ort}, \text{BLand}\} \to \{\text{Vorwahl}\}$
- FD_3: $\{\text{PLZ}\} \to \{\text{Ort}, \text{BLand}\}$
- FD_4: $\{\text{Ort}, \text{Strasse}, \text{BLand}\} \to \{\text{PLZ}\}$
- FD_5: $\{\text{Raum}\} \to \{\text{PersNr}\}$

MitarbeiterAdr								
<u>PersNr</u>	Name	Gehaltsstufe	Raum	Ort	Strasse	PLZ	Vorwahl	BLand



$\rightsquigarrow$ $\mathcal{TAFEL}$

Mehrwertige Abhängigkeiten

Definition 13 (mehrwertig abhängig)

Sei $\mathcal{R}$ ein relationales Schema und gelte $\alpha, \beta, \gamma \subseteq \mathcal{R}$ mit $\alpha \cup \beta \cup \gamma = \mathcal{R}$. Dann ist β mehrwertig abhängig von α, in Zeichen: $\alpha \twoheadrightarrow \beta$, wenn in jeder gültigen Ausprägung von $\mathcal{R}$ gilt: Für jedes Paar von Tupeln, t_1, t_2, mit $t_1(\alpha) = t_2(\alpha)$ existieren zwei weitere Tupel t_3 und t_4 mit folgenden Eigenschaften:

$$t_1(\alpha) = t_2(\alpha) = t_3(\alpha) = t_4(\alpha)$$

$$t_1(\beta) = t_3(\beta)$$

$$t_2(\beta) = t_4(\beta)$$

$$t_1(\gamma) = t_4(\gamma)$$

$$t_2(\gamma) = t_3(\gamma)$$

Mehrwertige Abhängigkeiten

Definition 13 (mehrwertig abhängig)

Sei $\mathcal{R}$ ein relationales Schema und gelte $\alpha, \beta, \gamma \subseteq \mathcal{R}$ mit $\alpha \cup \beta \cup \gamma = \mathcal{R}$. Dann ist β mehrwertig abhängig von α, in Zeichen: $\alpha \twoheadrightarrow \beta$, wenn in jeder gültigen Ausprägung von $\mathcal{R}$ gilt: Für jedes Paar von Tupeln, t_1, t_2, mit $t_1(\alpha) = t_2(\alpha)$ existieren zwei weitere Tupel t_3 und t_4 mit folgenden Eigenschaften:

$$t_1(\alpha) = t_2(\alpha) = t_3(\alpha) = t_4(\alpha)$$
$$t_1(\beta) = t_3(\beta)$$
$$t_2(\beta) = t_4(\beta)$$
$$t_1(\gamma) = t_4(\gamma)$$
$$t_2(\gamma) = t_3(\gamma)$$

Alternative Darstellung (gleiche Farbe entspricht gleichem Wert) [Beispiel] :

$$\begin{array}{lll} t_1(\alpha) & t_1(\beta) & t_1(\gamma) \\ t_3(\alpha) & t_3(\beta) & t_3(\gamma) \\ \\ t_4(\alpha) & t_4(\beta) & t_4(\gamma) \\ t_2(\alpha) & t_2(\beta) & t_2(\gamma) \end{array}$$

Bemerkungen:

- Ist β mehrwertig abhängig von α, so kann man in der zugehörigen Relation r bei je zwei Tupeln, die den gleichen α-Wert haben, die β-Werte vertauschen und es gilt, dass die resultierenden Tupel auch in r enthalten sind.

- Mehrwertige Abhängigkeiten *(Multivalued Dependency, MVD)* stellen eine Verallgemeinerung funktionaler Abhängigkeiten (FDs) dar. Die linke Seite einer MVD bestimmt für ihre rechte Seite eine *Menge* von Werten (Stichwort: mehrwertig). Es gilt:

  $$\alpha \rightarrow \beta \quad \Rightarrow \quad \alpha \twoheadrightarrow \beta$$

  Eine FD ist eine MVD, bei der höchstens eine Ausprägung von β mit je einer Ausprägung von α verknüpft ist.

- Eine MVD kann immer dann entstehen, wenn zwei unabhängige 1: N-Beziehungen in *einer* Relation kombiniert werden. Die Beziehungen können als orthogonal zueinander (= unabhängig voneinander) verstanden werden.

- Aus der Symmetrie der Definition folgt auch die *Komplementregel*:

  $$\alpha \twoheadrightarrow \beta \quad \Rightarrow \quad \alpha \twoheadrightarrow \gamma$$

Mehrwertige Abhängigkeiten

Beispiel:

Faehigkeiten		
PersNr	Sprache	ProgSprache
3002	griechisch	C
3002	lateinisch	Pascal
3002	griechisch	Pascal
3002	lateinisch	C
3005	deutsch	Java

- In dieser Relation gelten die MVDs $\{\text{PersNr}\} \twoheadrightarrow \{\text{Sprache}\}$ und $\{\text{PersNr}\} \twoheadrightarrow \{\text{ProgSprache}\}$.
- Die Attributmenge $\{\text{PersNr}, \text{Sprache}, \text{ProgSprache}\}$ bildet den Schlüssel; die Relation ist also in BCNF.

Mehrwertige Abhängigkeiten

Sei $\mathcal{R}$ ein relationales Schema und gelte $\alpha, \beta, \gamma \subseteq \mathcal{R}$ mit $\alpha \cup \beta \cup \gamma = \mathcal{R}$ und sei r eine Aupägung von $\mathcal{R}$.

Semantisch drückt eine mehrwertige Abhängigkeit die Unabhängigkeit der Attributmengen β und γ voneinander in der Relation r aus: pro α-Wert bildet das kartesische Produkt der β- und γ-Werte die Menge der $\beta\gamma$-Werte für α:

$$\alpha \twoheadrightarrow \beta$$

$$\Leftrightarrow$$

$$\forall a \in \{t(\alpha) \mid t \in r\} : \quad \pi_\beta \qquad \times \pi_\gamma \qquad = \pi_{\beta\gamma}$$

Mehrwertige Abhängigkeiten

Sei $\mathcal{R}$ ein relationales Schema und gelte $\alpha, \beta, \gamma \subseteq \mathcal{R}$ mit $\alpha \cup \beta \cup \gamma = \mathcal{R}$ und sei r eine Auprägung von $\mathcal{R}$.

Semantisch drückt eine mehrwertige Abhängigkeit die Unabhängigkeit der Attributmengen β und γ voneinander in der Relation r aus: pro α-Wert bildet das kartesische Produkt der β- und γ-Werte die Menge der $\beta\gamma$-Werte für α:

$$\alpha \twoheadrightarrow \beta$$

$$\Leftrightarrow$$

$$\forall a \in \{t(\alpha) \mid t \in r\} : \quad \pi_{\beta}(\sigma_{\alpha=a}(r)) \times \pi_{\gamma}(\sigma_{\alpha=a}(r)) \; = \; \pi_{\beta\gamma}(\sigma_{\alpha=a}(r))$$

Mehrwertige Abhängigkeiten

Sei $\mathcal{R}$ ein relationales Schema und gelte $\alpha, \beta, \gamma \subseteq \mathcal{R}$ mit $\alpha \cup \beta \cup \gamma = \mathcal{R}$ und sei r eine Auprägung von $\mathcal{R}$.

Semantisch drückt eine mehrwertige Abhängigkeit die Unabhängigkeit der Attributmengen β und γ voneinander in der Relation r aus: pro α-Wert bildet das kartesische Produkt der β- und γ-Werte die Menge der $\beta\gamma$-Werte für α:

$$\alpha \twoheadrightarrow \beta$$

$$\Leftrightarrow$$

$$\forall a \in \{t(\alpha) \mid t \in r\}: \quad \pi_\beta(\sigma_{\alpha=a}(r)) \times \pi_\gamma(\sigma_{\alpha=a}(r)) \; = \; \pi_{\beta\gamma}(\sigma_{\alpha=a}(r))$$

Alternative Formulierung für den Spezialfall $|\alpha| = |\beta| = |\gamma| = 1$:

Wenn $\{b_1, \ldots, b_k\}$ und $\{c_1, \ldots, c_l\}$ die β- bzw. γ-Werte für einen bestimmten α-Wert a in einer Relation r sind, so muss r auch die folgenden $k \cdot l$ Tripel enthalten:

$$\{a\} \times \{b_1, \ldots, b_k\} \times \{c_1, \ldots, c_l\}$$

Bemerkungen:

- Immer wenn zwei Tupel mit gleichem α-Wert und verschiedenem β-Wert existieren, so müssen auch Tupel existieren, die für diesen α-Wert und jeden β-Wert alle Kombinationen von γ-Werten beinhalten.

- Man nennt mehrwertige Abhängigkeiten auch „tupelgenerierende“ Abhängigkeiten: eine Relationenausprägung kann bei Verletzung einer MVD durch das Einfügen zusätzlicher Tupel in einen Zustand überführt werden, der die MVD erfüllt.

Definition 14 (triviale MVD)

Sei $\mathcal{R}$ ein relationales Schema, $\alpha \cup \beta \subseteq \mathcal{R}$. Eine MVD $\alpha \twoheadrightarrow \beta$ ist trivial hinsichtlich $\mathcal{R}$, falls jede mögliche Ausprägung r von $\mathcal{R}$ diese MVD erfüllt.

Definition 14 (triviale MVD)

Sei $\mathcal{R}$ ein relationales Schema, $\alpha \cup \beta \subseteq \mathcal{R}$. Eine MVD $\alpha \twoheadrightarrow \beta$ ist trivial hinsichtlich $\mathcal{R}$, falls jede mögliche Ausprägung r von $\mathcal{R}$ diese MVD erfüllt.

Ein Relationenschema $\mathcal{R}$ mit zugehöriger Menge F von funktionalen und mehrwertigen Abhängigkeiten ist in vierter Normalform (4NF), wenn für jede MVD $(\alpha \twoheadrightarrow \beta) \in F^+$ mindestens eine der folgenden Bedingungen erfüllt ist:

- die MVD ist trivial
- α ist Superschlüssel von $\mathcal{R}$

Mehrwertige Abhängigkeiten

Vierte Normalform

Definition 14 (triviale MVD)

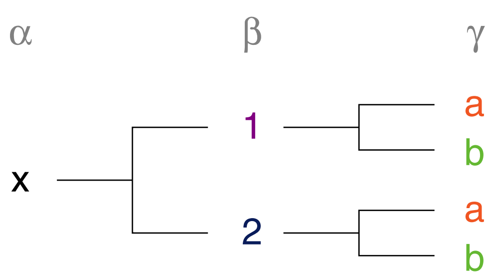
Sei $\mathcal{R}$ ein relationales Schema, $\alpha \cup \beta \subseteq \mathcal{R}$. Eine MVD $\alpha \twoheadrightarrow \beta$ ist trivial hinsichtlich $\mathcal{R}$, falls jede mögliche Ausprägung r von $\mathcal{R}$ diese MVD erfüllt.

Ein Relationenschema $\mathcal{R}$ mit zugehöriger Menge F von funktionalen und mehrwertigen Abhängigkeiten ist in vierter Normalform (4NF), wenn für jede MVD $(\alpha \twoheadrightarrow \beta) \in F^+$ mindestens eine der folgenden Bedingungen erfüllt ist:

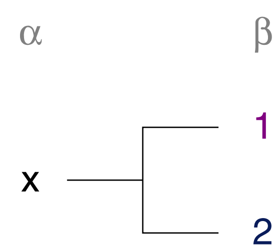
- die MVD ist trivial
- α ist Superschlüssel von $\mathcal{R}$

Beispiel [Definition] :

nichttriviale MVD $\alpha \twoheadrightarrow \beta$



triviale MVD $\alpha \twoheadrightarrow \beta$



Bemerkungen:

- Man kann zeigen, dass $\alpha \twoheadrightarrow \beta$ genau dann trivial ist, wenn $\beta \subseteq \alpha$ oder wenn $\beta = \mathcal{R} - \alpha$ gilt.
- Bei Relationen in der vierten Normalform wird die durch mehrwertige Abhängigkeiten verursachte Redundanz ausgeschlossen: Relationen in 4NF enthalten keine zwei voneinander unabhängigen, mehrwertigen Fakten.
- Die vierte Normalform erreicht man durch Elimination der rechten Seite einer der beiden mehrwertigen Abhängigkeiten. Der eliminierte Teil bildet zusammen mit der linken Seite der MVD eine neue Relation.
- Die vierte Normalform ist eine Verschärfung der Boyce-Codd-Normalform.

Mehrwertige Abhängigkeiten

Verlustlose Zerlegung

Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen und mehrwertigen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1$ und $\mathcal{R}_2$ ist genau dann verlustlos, wenn mindestens eine der folgenden Bedingungen gilt:

- $((\mathcal{R}_1 \cap \mathcal{R}_2) \twoheadrightarrow \mathcal{R}_1) \in F^+$
- $((\mathcal{R}_1 \cap \mathcal{R}_2) \twoheadrightarrow \mathcal{R}_2) \in F^+$

Mehrwertige Abhängigkeiten

Verlustlose Zerlegung

Eine Zerlegung von $\mathcal{R}$ mit zugehörigen funktionalen und mehrwertigen Abhängigkeiten F in die Relationenschemata $\mathcal{R}_1$ und $\mathcal{R}_2$ ist genau dann verlustlos, wenn mindestens eine der folgenden Bedingungen gilt:

- $((\mathcal{R}_1 \cap \mathcal{R}_2) \twoheadrightarrow \mathcal{R}_1) \in F^+$
- $((\mathcal{R}_1 \cap \mathcal{R}_2) \twoheadrightarrow \mathcal{R}_2) \in F^+$

Beispiel (Fortsetzung):

Faehigkeiten		
PersNr	Sprache	ProgSprache
3002	griechisch	C
3002	lateinisch	Pascal
3002	griechisch	Pascal
3002	lateinisch	C
3005	deutsch	Java

$\leadsto$

Sprachen	
PersNr	Sprache
3002	griechisch
3002	lateinisch
3005	deutsch

ProgSprachen	
PersNr	ProgSprache
3002	C
3002	Pascal
3005	Java

Die Zerlegung in die Relationenschemata „Sprachen“ und „ProgSprachen“ ist verlustlos:

Faehigkeiten $= \pi_{\text{PersNr,Sprache}}(\text{Faehigkeiten}) \bowtie \pi_{\text{PersNr,ProgSprache}}(\text{Faehigkeiten})$

Mehrwertige Abhängigkeiten

Relationale Dekomposition

Algorithm: RelDecompositionMVD

Input: $\mathcal{R}$. Universalrelation.

F. Menge funktionaler Abhängigkeiten für $\mathcal{R}$.

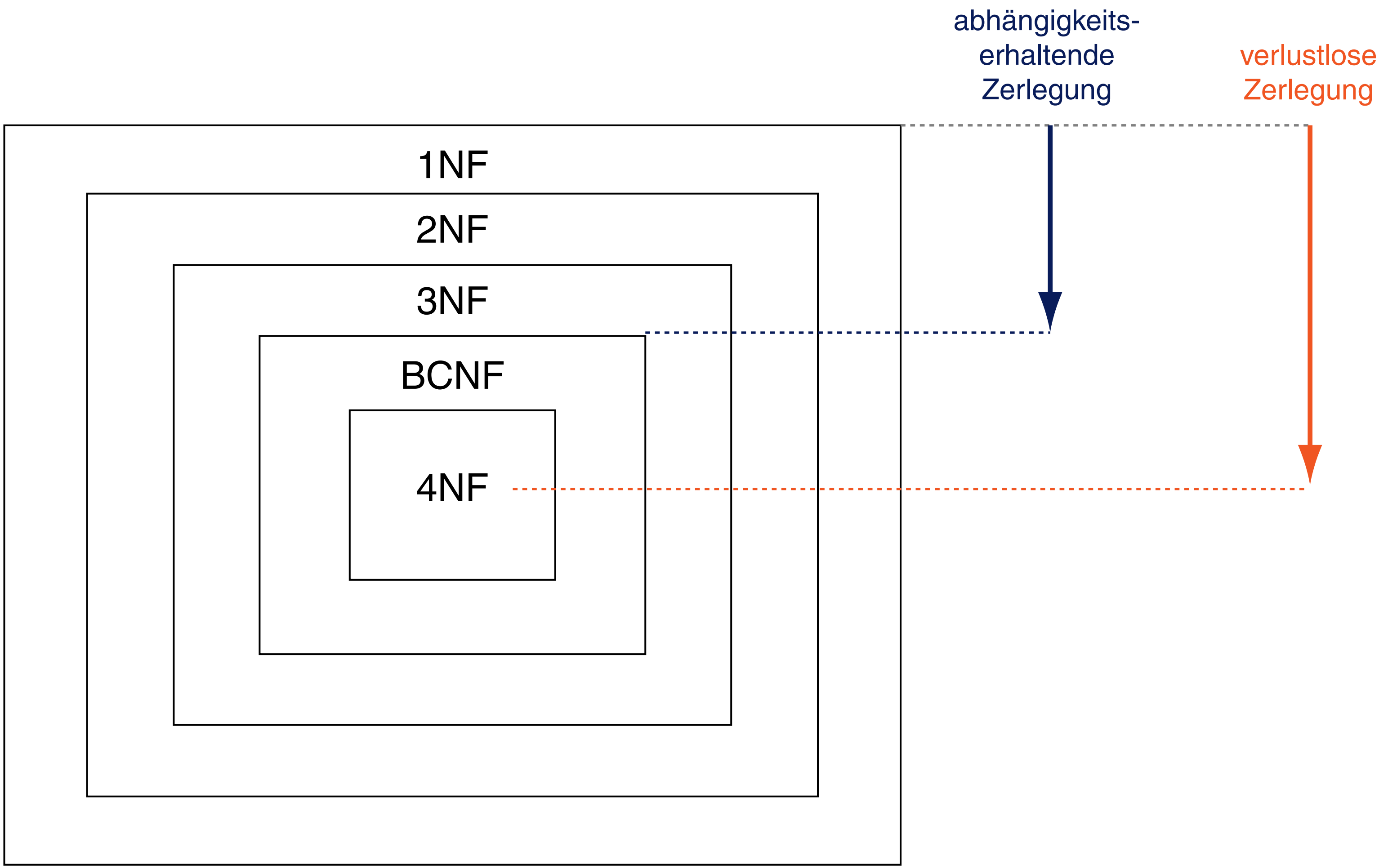
Output: $\boldsymbol{\mathcal{R}}$. Verlustlose Zerlegung von $\mathcal{R}$ mit Schemata in 4NF.

1. $\boldsymbol{\mathcal{R}} = \{\mathcal{R}\}$
2. **WHILE** $\exists \mathcal{R}' : (\mathcal{R}' \in \boldsymbol{\mathcal{R}} \;\wedge\; \mathcal{R}'$ not in 4NF$)$ **DO**
3. Find nontrivial MVD $(\alpha \twoheadrightarrow \beta) \in F_{\boldsymbol{\mathcal{R}}'}$ that violates 4NF
4. Decompose $\mathcal{R}'$ into $\mathcal{R}'_1 = \mathcal{R}' - \beta$ and $\mathcal{R}'_2 = \alpha \cup \beta$
5. $\boldsymbol{\mathcal{R}} = (\boldsymbol{\mathcal{R}} - \{\mathcal{R}'\}) \;\cup\; \{\mathcal{R}'_1, \mathcal{R}'_2\}$
6. **ENDDO**
7. **RETURN**$(\boldsymbol{\mathcal{R}})$

Bemerkungen:

- Der Algorithmus RelDecompostionMVD erzeugt nicht notwendigerweise eine Zerlegung, die abhängigkeitserhaltend ist. Dies folgt aus der Tatsache, das eine Relation in 4NF gleichzeitig auch immer in BCNF ist.

Beziehungen der Normalformen



[Kemper/Eickler 2011]