

Datenbanken:

Tutorium 8

Marvin Jahn

11.12.2019

Zerlegungsalgorithmen

- Gegeben: “schlechtes” relationales Schema

Zerlegungsalgorithmen

- Gegeben: “schlechtes” relationales Schema
- Ziel: Überführung (**Zerlegung**) dieses Schemas in eine möglichst hohe Normalform (höhere Normalform = “besseres” Schema)

Kanonische Überdeckung

Es sei \mathcal{F} eine Menge von FDs.

Kanonische Überdeckung

Es sei \mathcal{F} eine Menge von FDs.

Die **Hülle (closure)** von \mathcal{F} ist die Menge aller aus \mathcal{F} herleitbaren FDs (mit den *Armstrong Axiomen*). Sie wird mit \mathcal{F}^+ bezeichnet.

Kanonische Überdeckung

Es sei \mathcal{F} eine Menge von FDs.

Die **Hülle (closure)** von \mathcal{F} ist die Menge aller aus \mathcal{F} herleitbaren FDs (mit den *Armstrong Axiomen*). Sie wird mit \mathcal{F}^+ bezeichnet.

\mathcal{F}_c heißt **kanonische Überdeckung** von \mathcal{F} , falls die folgenden Kriterien gelten:

- $\mathcal{F}_c^+ = \mathcal{F}^+$

Kanonische Überdeckung

Es sei \mathcal{F} eine Menge von FDs.

Die **Hülle (closure)** von \mathcal{F} ist die Menge aller aus \mathcal{F} herleitbaren FDs (mit den *Armstrong Axiomen*). Sie wird mit \mathcal{F}^+ bezeichnet.

\mathcal{F}_c heißt **kanonische Überdeckung** von \mathcal{F} , falls die folgenden Kriterien gelten:

- $\mathcal{F}_c^+ = \mathcal{F}^+$
- In \mathcal{F}_c existieren keine FDs $\alpha \rightarrow \beta$, bei denen α oder β überflüssige Attribute enthalten, d.h.

$$\forall A \in \alpha : (F_c \setminus (\alpha \rightarrow \beta) \cup ((\alpha \setminus A) \rightarrow \beta))^+ \neq F_c^+,$$

$$\forall B \in \beta : (F_c \setminus (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta \setminus B)))^+ \neq F_c^+.$$

Kanonische Überdeckung

Es sei \mathcal{F} eine Menge von FDs.

Die **Hülle (closure)** von \mathcal{F} ist die Menge aller aus \mathcal{F} herleitbaren FDs (mit den *Armstrong Axiomen*). Sie wird mit \mathcal{F}^+ bezeichnet.

\mathcal{F}_c heißt **kanonische Überdeckung** von \mathcal{F} , falls die folgenden Kriterien gelten:

- $\mathcal{F}_c^+ = \mathcal{F}^+$
- In \mathcal{F}_c existieren keine FDs $\alpha \rightarrow \beta$, bei denen α oder β überflüssige Attribute enthalten, d.h.

$$\forall A \in \alpha : (F_c \setminus (\alpha \rightarrow \beta) \cup ((\alpha \setminus A) \rightarrow \beta))^+ \neq F_c^+,$$

$$\forall B \in \beta : (F_c \setminus (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta \setminus B)))^+ \neq F_c^+.$$

- Jede linke Seite einer FD in \mathcal{F}_c ist einzigartig. Dies kann durch sukzessive Anwendung der Vereinigungsregel erzielt werden:

$$\alpha \rightarrow \beta, \alpha \rightarrow \gamma \Rightarrow \alpha \rightarrow \beta\gamma.$$

Berechnung einer Kanonischen Überdeckung

Berechnung einer Kanonischen Überdeckung

Führe für jede FD $\alpha \rightarrow \beta \in \mathcal{F}$ die **Linksreduktion** durch:

- Überprüfe für alle $A \in \alpha$, ob β funktional abhängig von $\alpha - A$ bzgl. \mathcal{F} ist.
- Falls ja, ersetze $\alpha \rightarrow \beta$ durch $(\alpha - A) \rightarrow \beta$.

Berechnung einer Kanonischen Überdeckung

Führe für jede FD $\alpha \rightarrow \beta \in \mathcal{F}$ die **Linksreduktion** durch:

- Überprüfe für alle $A \in \alpha$, ob β funktional abhängig von $\alpha - A$ bzgl. \mathcal{F} ist.
- Falls ja, ersetze $\alpha \rightarrow \beta$ durch $(\alpha - A) \rightarrow \beta$.

Führe für jede (verbleibende) FD $\alpha \rightarrow \beta$ die **Rechtsreduktion** durch:

- Überprüfe für alle $B \in \beta$, ob β funktional abhängig von α bzgl. $\mathcal{F} - (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta - B))$
- Falls ja, ersetze $\alpha \rightarrow \beta$ in \mathcal{F} durch $\alpha \rightarrow (\beta - B)$.

Berechnung einer Kanonischen Überdeckung

Führe für jede FD $\alpha \rightarrow \beta \in \mathcal{F}$ die **Linksreduktion** durch:

- Überprüfe für alle $A \in \alpha$, ob β funktional abhängig von $\alpha - A$ bzgl. \mathcal{F} ist.
- Falls ja, ersetze $\alpha \rightarrow \beta$ durch $(\alpha - A) \rightarrow \beta$.

Führe für jede (verbleibende) FD $\alpha \rightarrow \beta$ die **Rechtsreduktion** durch:

- Überprüfe für alle $B \in \beta$, ob β funktional abhängig von α bzgl. $\mathcal{F} - (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta - B))$
- Falls ja, ersetze $\alpha \rightarrow \beta$ in \mathcal{F} durch $\alpha \rightarrow (\beta - B)$.

Entferne alle FDs der Form $\alpha \rightarrow \emptyset$ von \mathcal{F} .

Berechnung einer Kanonischen Überdeckung

Führe für jede FD $\alpha \rightarrow \beta \in \mathcal{F}$ die **Linksreduktion** durch:

- Überprüfe für alle $A \in \alpha$, ob β funktional abhängig von $\alpha - A$ bzgl. \mathcal{F} ist.
- Falls ja, ersetze $\alpha \rightarrow \beta$ durch $(\alpha - A) \rightarrow \beta$.

Führe für jede (verbleibende) FD $\alpha \rightarrow \beta$ die **Rechtsreduktion** durch:

- Überprüfe für alle $B \in \beta$, ob β funktional abhängig von α bzgl. $\mathcal{F} - (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta - B))$
- Falls ja, ersetze $\alpha \rightarrow \beta$ in \mathcal{F} durch $\alpha \rightarrow (\beta - B)$.

Entferne alle FDs der Form $\alpha \rightarrow \emptyset$ von \mathcal{F} .

Fasse mittels der Vereinigungsregel FDs der Form $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$ zu $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ zusammen.

Wdh.: Dritte Normalform (3NF)

Ein Relationenschema R ist in **3NF**, wenn für jede FD $\alpha \rightarrow \beta$ mind. eine der folgenden Eigenschaften gilt:

- $\alpha \rightarrow \beta$ ist *trivial*, d.h. $\beta \subset \alpha$.
- α ist ein Superschlüssel.
- Jedes Attribut von β ist in einem Schlüssel enthalten.

3NF-Synthesealgorithmus

- überführt Schema R mit FDs \mathcal{F} in 3NF
- berechnete Zerlegung ist *verlustlos* und *abhängigkeitsbewahrend*
- gegeben: kanonische Überdeckung \mathcal{F}_c

3NF-Synthesealgorithmus

- überführt Schema R mit FDs \mathcal{F} in 3NF
- berechnete Zerlegung ist *verlustlos* und *abhängigkeitsbewahrend*
- gegeben: kanonische Überdeckung \mathcal{F}_c

Vorgehen:

- Für jede FD $\alpha \rightarrow \beta$ in \mathcal{F}_c erstelle Unterschema $R_\alpha := \alpha \cup \beta$ mit FDs $\mathcal{F}_\alpha := \{\alpha' \rightarrow \beta' \in \mathcal{F}_c : \alpha' \cup \beta' \subset R_\alpha\}$

3NF-Synthesealgorithmus

- überführt Schema R mit FDs \mathcal{F} in 3NF
- berechnete Zerlegung ist *verlustlos* und *abhängigkeitsbewahrend*
- gegeben: kanonische Überdeckung \mathcal{F}_c

Vorgehen:

- Für jede FD $\alpha \rightarrow \beta$ in \mathcal{F}_c erstelle Unterschema $R_\alpha := \alpha \cup \beta$ mit FDs $\mathcal{F}_\alpha := \{\alpha' \rightarrow \beta' \in \mathcal{F}_c : \alpha' \cup \beta' \subset R_\alpha\}$
- Falls keines der im ersten Schritt erstellten Schemata einen Kandidatenschlüssel von R bzgl. \mathcal{F}_c enthält, wähle einen Kandidatenschlüssel $\kappa \subset R$ und füge das Schema $R_\kappa = \kappa$, $\mathcal{F}_\kappa = \emptyset$ hinzu.

3NF-Synthesealgorithmus

- überführt Schema R mit FDs \mathcal{F} in 3NF
- berechnete Zerlegung ist *verlustlos* und *abhängigkeitsbewahrend*
- gegeben: kanonische Überdeckung \mathcal{F}_c

Vorgehen:

- Für jede FD $\alpha \rightarrow \beta$ in \mathcal{F}_c erstelle Unterschema $R_\alpha := \alpha \cup \beta$ mit FDs $\mathcal{F}_\alpha := \{\alpha' \rightarrow \beta' \in \mathcal{F}_c : \alpha' \cup \beta' \subset R_\alpha\}$
- Falls keines der im ersten Schritt erstellten Schemata einen Kandidatenschlüssel von R bzgl. \mathcal{F}_c enthält, wähle einen Kandidatenschlüssel $\kappa \subset R$ und füge das Schema $R_\kappa = \kappa$, $\mathcal{F}_\kappa = \emptyset$ hinzu.
- Entferne die Schemata, die bereits in einem anderen Relationenschema enthalten sind.

Wdh.: Boyce-Codd Normalform (BCNF)

Ein Relationenschema R ist in **BCNF**, wenn für jede FD $\alpha \rightarrow \beta$ mind. eine der folgenden Eigenschaften gilt:

- $\alpha \rightarrow \beta$ ist *trivial*, d.h. $\beta \subset \alpha$.
- α ist ein Superschlüssel.

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.
- \mathcal{F}_i bezeichne die Menge der nicht-trivialen FDs $\alpha \rightarrow \beta \in \mathcal{F}^+$ mit $\alpha \cup \beta \subset R_i$, d.h. \mathcal{F}_i enthält die zu R_i "gehörenden" FDs.

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.
- \mathcal{F}_i bezeichne die Menge der nicht-trivialen FDs $\alpha \rightarrow \beta \in \mathcal{F}^+$ mit $\alpha \cup \beta \subset R_i$, d.h. \mathcal{F}_i enthält die zu R_i "gehörenden" FDs.
- Solange es noch $R_i \in Z$ gibt, die nicht in BCNF sind (bzgl. \mathcal{F}_i):

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.
- \mathcal{F}_i bezeichne die Menge der nicht-trivialen FDs $\alpha \rightarrow \beta \in \mathcal{F}^+$ mit $\alpha \cup \beta \subset R_i$, d.h. \mathcal{F}_i enthält die zu R_i "gehörenden" FDs.
- Solange es noch $R_i \in Z$ gibt, die nicht in BCNF sind (bzgl. \mathcal{F}_i):
 - ▶ Es gibt also eine nicht-triviale FD $(\alpha \rightarrow \beta) \in \mathcal{F}_i$, die die BCNF verletzt, d.h. α ist kein Superschlüssel von R_i .

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.
- \mathcal{F}_i bezeichne die Menge der nicht-trivialen FDs $\alpha \rightarrow \beta \in \mathcal{F}^+$ mit $\alpha \cup \beta \subset R_i$, d.h. \mathcal{F}_i enthält die zu R_i "gehörenden" FDs.
- Solange es noch $R_i \in Z$ gibt, die nicht in BCNF sind (bzgl. \mathcal{F}_i):
 - ▶ Es gibt also eine nicht-triviale FD $(\alpha \rightarrow \beta) \in \mathcal{F}_i$, die die BCNF verletzt, d.h. α ist kein Superschlüssel von R_i .
 - ▶ Wähle eine solche FD mit $\alpha \cap \beta = \emptyset$ und β möglichst groß. Letzteres führt dazu, dass der Algorithmus schneller terminiert.

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.
- \mathcal{F}_i bezeichne die Menge der nicht-trivialen FDs $\alpha \rightarrow \beta \in \mathcal{F}^+$ mit $\alpha \cup \beta \subset R_i$, d.h. \mathcal{F}_i enthält die zu R_i "gehörenden" FDs.
- Solange es noch $R_i \in Z$ gibt, die nicht in BCNF sind (bzgl. \mathcal{F}_i):
 - ▶ Es gibt also eine nicht-triviale FD $(\alpha \rightarrow \beta) \in \mathcal{F}_i$, die die BCNF verletzt, d.h. α ist kein Superschlüssel von R_i .
 - ▶ Wähle eine solche FD mit $\alpha \cap \beta = \emptyset$ und β möglichst groß. Letzteres führt dazu, dass der Algorithmus schneller terminiert.
 - ▶ Definiere $R_{i_1} := \alpha \cup \beta$, $R_{i_2} := R_i - \beta$.

Dekompositions-Algorithmus

- überführt Schema R mit FDs \mathcal{F} *verlustlos* in BCNF
- es gibt Schemata, die nicht *abhängigkeitsbewahrend* in BCNF zerlegt werden können

Vorgehen:

- Starte $Z := \{R\}$.
- \mathcal{F}_i bezeichne die Menge der nicht-trivialen FDs $\alpha \rightarrow \beta \in \mathcal{F}^+$ mit $\alpha \cup \beta \subset R_i$, d.h. \mathcal{F}_i enthält die zu R_i "gehörenden" FDs.
- Solange es noch $R_i \in Z$ gibt, die nicht in BCNF sind (bzgl. \mathcal{F}_i):
 - ▶ Es gibt also eine nicht-triviale FD $(\alpha \rightarrow \beta) \in \mathcal{F}_i$, die die BCNF verletzt, d.h. α ist kein Superschlüssel von R_i .
 - ▶ Wähle eine solche FD mit $\alpha \cap \beta = \emptyset$ und β möglichst groß. Letzteres führt dazu, dass der Algorithmus schneller terminiert.
 - ▶ Definiere $R_{i_1} := \alpha \cup \beta$, $R_{i_2} := R_i - \beta$.
 - ▶ Entferne R_i aus Z und füge R_{i_1} und R_{i_2} zu Z hinzu.

Dekompositions-Algorithmus

Beispiel: Gegeben sei die Relation (in 3NF, nicht in BCNF)

$R := \text{Staedte} : \{[Ort, Bundeland, Ministerpraesident, Einwohner]\}$

mit den FDs

$$\{Ort, Bundeland\} \rightarrow \{Einwohner\} \quad (1)$$

$$\{Bundeland\} \rightarrow \{Ministerpraesident\} \quad (2)$$

$$\{Ministerpraesident\} \rightarrow \{Bundeland\}. \quad (3)$$

Dekompositions-Algorithmus

Beispiel: Gegeben sei die Relation (in 3NF, nicht in BCNF)

$R := \text{Staedte} : \{[Ort, Bundeland, Ministerpraesident, Einwohner]\}$

mit den FDs

$$\{Ort, Bundeland\} \rightarrow \{Einwohner\} \quad (1)$$

$$\{Bundeland\} \rightarrow \{Ministerpraesident\} \quad (2)$$

$$\{Ministerpraesident\} \rightarrow \{Bundeland\}. \quad (3)$$

Dekompositions-Algorithmus:

- Starte $Z = \{R\}$.

Dekompositions-Algorithmus

Beispiel: Gegeben sei die Relation (in 3NF, nicht in BCNF)

$R := \text{Staedte} : \{[Ort, Bundeland, Ministerpraesident, Einwohner]\}$

mit den FDs

$$\{Ort, Bundeland\} \rightarrow \{Einwohner\} \quad (1)$$

$$\{Bundeland\} \rightarrow \{Ministerpraesident\} \quad (2)$$

$$\{Ministerpraesident\} \rightarrow \{Bundeland\}. \quad (3)$$

Dekompositions-Algorithmus:

- Starte $Z = \{R\}$.
- Die FD $\{Bundeland\} \rightarrow \{Ministerpraesident\}$ verletzt die BCNF.

Dekompositions-Algorithmus

Beispiel: Gegeben sei die Relation (in 3NF, nicht in BCNF)

$R := \text{Staedte} : \{[Ort, Bundeland, Ministerpraesident, Einwohner]\}$

mit den FDs

$$\{Ort, Bundeland\} \rightarrow \{Einwohner\} \quad (1)$$

$$\{Bundeland\} \rightarrow \{Ministerpraesident\} \quad (2)$$

$$\{Ministerpraesident\} \rightarrow \{Bundeland\}. \quad (3)$$

Dekompositions-Algorithmus:

- Starte $Z = \{R\}$.
- Die FD $\{Bundeland\} \rightarrow \{Ministerpraesident\}$ verletzt die BCNF.
- Also zerlege:

$$R_1 := \{[Bundeland, Ministerpraesident]\}, (2) \in \mathcal{F}_1, (3) \in \mathcal{F}_1,$$

$$R_2 := \{[Ort, Bundeland, Einwohner]\}, (1) \in \mathcal{F}_2.$$

Dekompositions-Algorithmus

Beispiel: Gegeben sei die Relation (in 3NF, nicht in BCNF)

$R := \text{Staedte} : \{[Ort, Bundeland, Ministerpraesident, Einwohner]\}$

mit den FDs

$$\{Ort, Bundeland\} \rightarrow \{Einwohner\} \quad (1)$$

$$\{Bundeland\} \rightarrow \{Ministerpraesident\} \quad (2)$$

$$\{Ministerpraesident\} \rightarrow \{Bundeland\}. \quad (3)$$

Dekompositions-Algorithmus:

- Starte $Z = \{R\}$.
- Die FD $\{Bundeland\} \rightarrow \{Ministerpraesident\}$ verletzt die BCNF.
- Also zerlege:

$$R_1 := \{[Bundeland, Ministerpraesident]\}, (2) \in \mathcal{F}_1, (3) \in \mathcal{F}_1,$$

$$R_2 := \{[Ort, Bundeland, Einwohner]\}, (1) \in \mathcal{F}_2.$$

- R_1 ist bzgl. \mathcal{F}_1 und R_2 bzgl. \mathcal{F}_2 in BCNF, also haben wir eine Zerlegung gefunden.