

Synthesealgorithmus [Relation A-H]

Überführen Sie das Relationenschema mit Hilfe des Synthesealgorithmus in die 3. Normalform!

$$R(A, B, C, D, E, F, G, H)$$

$$FA = \left\{ \begin{array}{l} \{F\} \rightarrow \{E\}, \\ \{A\} \rightarrow \{B, D\}, \\ \{A, E\} \rightarrow \{D\}, \\ \{A\} \rightarrow \{E, F\}, \\ \{A, G\} \rightarrow \{H\}, \end{array} \right\}$$

(a) Kanonische Überdeckung

(i) Linksreduktion

— Führe für jede funktionale Abhängigkeit $\alpha \rightarrow \beta \in F$ die Linksreduktion durch, überprüfe also für alle $A \in \alpha$, ob A überflüssig ist, d. h. ob $\beta \subseteq \text{AttrHülle}(F, \alpha - A)$.

Wir betrachten nur die zusammengesetzten Attribute:

$$\{A, E\} \rightarrow \{D\}$$

$$D \in \text{AttrHülle}(F, \{A, E \setminus E\}) = \{A, E, F, B, D\}$$

$$D \notin \text{AttrHülle}(F, \{A, E \setminus A\}) = \{E\}$$

$$\{A, G\} \rightarrow \{H\}$$

$$H \notin \text{AttrHülle}(F, \{A, G \setminus G\}) = \{A, E, F, B, D\}$$

$$H \notin \text{AttrHülle}(F, \{A, G \setminus A\}) = \{G\}$$

$$FA = \left\{ \begin{array}{l} \{F\} \rightarrow \{E\}, \\ \{A\} \rightarrow \{B, D\}, \\ \{A\} \rightarrow \{D\}, \\ \{A\} \rightarrow \{E, F\}, \\ \{A, G\} \rightarrow \{H\}, \end{array} \right\}$$

(ii) Rechtsreduktion

— Führe für jede (verbliebene) funktionale Abhängigkeit $\alpha \rightarrow \beta$ die Rechtsreduktion durch, überprüfe also für alle $B \in \beta$, ob $B \in \text{AttrHülle}(F - (\alpha \rightarrow \beta) \cup (\alpha \rightarrow (\beta - B)), \alpha)$ gilt. In diesem Fall ist B auf der rechten Seite überflüssig und kann eliminiert werden, d. h. $\alpha \rightarrow \beta$ wird durch $\alpha \rightarrow (\beta - B)$ ersetzt.

Nur die Attribute betrachten, die rechts doppelt vorkommen:

E

$$\text{AttrHülle}(F \setminus \{\{F\} \rightarrow \{E\}, \{F\}\}) = \{F\}$$

$$\text{AttrHülle}(F \setminus \{\{A\} \rightarrow \{E, F\} \cup \{A\} \rightarrow \{E\}, \{A\}\}) = \{A, B, D, F, E\}$$

D

$$\text{AttrHülle}(F \setminus \{\{A\} \rightarrow \{D\}, \{A\}\}) = \{A, B, D, F, E\}$$

$\{A\} \rightarrow \{D\}$ kann wegen der Armstrongschen Dekompositionsregel weggelassen werden. Wenn gilt $\{A\} \rightarrow \{B, D\}$, dann gilt auch $\{A\} \rightarrow \{B\}$ und $\{A\} \rightarrow \{D\}$

$$FA = \left\{ \begin{array}{l} \{ F \} \rightarrow \{ E \}, \\ \{ A \} \rightarrow \{ B, D \}, \\ \{ A \} \rightarrow \{ \emptyset \}, \\ \{ A \} \rightarrow \{ F \}, \\ \{ A, G \} \rightarrow \{ H \}, \end{array} \right\}$$

(iii) **Löschen leerer Klauseln**

— Entferne die funktionalen Abhängigkeiten der Form $\alpha \rightarrow \emptyset$, die im 2. Schritt möglicherweise entstanden sind. —

$$FA = \left\{ \begin{array}{l} \{ F \} \rightarrow \{ E \}, \\ \{ A \} \rightarrow \{ B, D \}, \\ \{ A \} \rightarrow \{ F \}, \\ \{ A, G \} \rightarrow \{ H \}, \end{array} \right\}$$

(iv) **Vereinigung**

— Fasse mittels der Vereinigungsregel funktionale Abhängigkeiten der Form $\alpha \rightarrow \beta_1, \dots, \alpha \rightarrow \beta_n$, so dass $\alpha \rightarrow \beta_1 \cup \dots \cup \beta_n$ verbleibt. —

$$FA = \left\{ \begin{array}{l} \{ F \} \rightarrow \{ E \}, \\ \{ A \} \rightarrow \{ B, D, F \}, \\ \{ A, G \} \rightarrow \{ H \}, \end{array} \right\}$$

(b) **Relationsschemata formen**

— Erzeuge für jede funktionale Abhängigkeit $\alpha \rightarrow \beta \in F_c$ ein Relationenschema $\mathcal{R}_\alpha := \alpha \cup \beta$. —

$$\begin{array}{l} R_1(\underline{E}, E) \\ R_2(\underline{A}, B, D, F) \\ R_3(\underline{A}, G, H) \end{array}$$

(c) **Schlüssel hinzufügen**

— Falls eines der in Schritt 2. erzeugten Schemata R_α einen Schlüsselkandidaten von \mathcal{R} bezüglich F_c enthält, sind wir fertig, sonst wähle einen Schlüsselkandidaten $K \subseteq \mathcal{R}$ aus und definiere folgendes zusätzliche Schema: $\mathcal{R}_K := K$ und $\mathcal{F}_K := \emptyset$ —

$$\begin{array}{l} R_1(\underline{E}, E) \\ R_2(\underline{A}, B, D, F) \\ R_3(\underline{A}, G, H) \\ R_4(\underline{A}, \underline{C}, G) \end{array}$$

(d) **Entfernung überflüssiger Teilschemata**

— Eliminiere diejenigen Schemata R_α , die in einem anderen Relationenschema $R_{\alpha'}$ enthalten sind, d. h. $R_\alpha \subseteq R_{\alpha'}$. —

\emptyset Nichts zu tun

Github: Module/10_DB/50_Relationale-Entwurfstheorie/30_Normalformen/10_Synthesealgorithmus/
Aufgabe_Relation-A-H.tex