



Übung zur Vorlesung *Grundlagen: Datenbanken* im WS19/20

Christoph Anneser, Moritz Sichert, Lukas Vogel (gdb@in.tum.de)

<https://db.in.tum.de/teaching/ws1920/grundlagen/>

Blatt Nr. 08

Tool zum Üben von funktionalen Abhängigkeiten: <https://normalizer.db.in.tum.de/>.

Hausaufgabe 1

Bewerten Sie die folgende Aussage: Die Modellierung in einem ER-Diagramm führt immer zu Relationen, die mindestens in 4. Normalform sind.

- Unter welchen Voraussetzungen können aus einem ER-Diagramm FDs (oder MVDs) entstehen, die die 4. Normalform (oder eine der niedrigeren) verletzen? Geben Sie ein beispielhaftes ER-Diagramm an, in dem ein Zusammenhang nicht korrekt modelliert wurde.
- Wie hängt die Modellierung von Entitytypen und Relationships zusammen mit der Bestimmung von FDs und MVDs?
- Welche Auswirkung hat die Verfeinerung von Relationen aus dem ER-Modell auf die Normalformen?

Hausaufgabe 2

Betrachten Sie ein abstraktes Relationenschema $R : \{[A, B, C, D, E, F, G]\}$ mit den FDs:

$$\begin{aligned}A &\rightarrow BC \\DE &\rightarrow B \\F &\rightarrow A \\E &\rightarrow BF \\A &\rightarrow DE \\C &\rightarrow A\end{aligned}$$

Überführen Sie die Relation verlustfrei und abhängigkeitsbewahrend in die dritte Normalform.

Hausaufgabe 3

Bestimmen Sie alle Kandidatenschlüssel der Relation R . Wenden Sie den Dekompositionsalgorithmus an, um die Relation R in die BCNF zu zerlegen und unterstreichen Sie die Schlüssel der Teilrelationen des Endergebnisses.

$$R : \{[A, B, C, D, E, F]\}$$

FDs:

- $B \rightarrow DA$
- $DEF \rightarrow B$
- $C \rightarrow EA$

Prüfen Sie als erstes, ob FD 1) für die Zerlegung geeignet ist und - falls dies der Fall ist - verwenden Sie diese im ersten Zerlegungsschritt. Für diese Aufgabe ist zu bedenken, dass die oben angegebenen FDs eine Charakterisierung der insgesamt geltenden FDs sind. Die Menge der geltenden FDs ist größer. Wieso? Wie muss dies beim Dekompositionsalgorithmus genutzt werden?

Hausaufgabe 4

Gegeben sei die durch folgende SQL-Statements definierte Ausprägung einer Relation.

```
create table kinder_fahrraeder (
  person varchar(100) not null,
  kind_name varchar(100) not null,
  kind_alter integer not null,
  fahrrad_typ varchar(100) not null,
  fahrrad_farbe varchar(100) not null
);
insert into kinder_fahrraeder values
('Thomas', 'Markus', 10, 'Trekking-Fahrrad', 'schwarz'),
('Thomas', 'Markus', 10, 'Mountainbike', 'rot'),
('Thomas', 'Johanna', 5, 'Trekking-Fahrrad', 'schwarz'),
('Thomas', 'Johanna', 5, 'Mountainbike', 'rot');
```

Es gelten die beiden komplementären MVD

1. $person \twoheadrightarrow \{kind_name, kind_alter\}$ und
2. $person \twoheadrightarrow \{fahrrad_typ, fahrrad_farbe\}$

sowie die FD

3. $kind_name \rightarrow kind_alter$.

- a) Laura, das dritte Kind von Thomas, wird geboren. Fügen Sie Laura per SQL-Insert-Statement hinzu und beachten Sie dabei die MVDs. Formulieren Sie Ihr Statement so, dass es auch ohne Kenntnis der Fahrräder von Thomas funktioniert (d.h. nicht `insert ... 'Mountainbike', 'rot'`);).
- b) Allgemein gesprochen: In eine Relation $R : \{[A, B, C]\}$ mit den MVDs $A \twoheadrightarrow B$ und $A \twoheadrightarrow C$ soll für ein bestimmtes a in Spalte A ein neuer Wert b in Spalte B eingefügt werden. Wie viele Tupel müssen hinzugefügt werden, damit die MVDs weiterhin gelten?
- c) Was passiert, wenn Thomas seine beiden Fahrräder verkauft?
- d) Überführen Sie die Relation `kinder_fahrraeder` mit dem Dekompositionsalgorithmus in die 4. NF.
- e) Schreiben Sie ein SQL-Statement um zu prüfen ob die MVDs der Relation `kinder_fahrraeder` erfüllt sind.