2. Übungsblatt

3.0 VU Datenmodellierung

2. Mai 2013

Allgemeines

In diesem Übungsteil sollten Sie Aufgabenstellungen aus den Bereich SQL und Normalformentheorie bearbeiten.

Lösen Sie die Beispiele eigenständig, denn in der Praxis (und bei der Prüfung) sind Sie auch auf sich alleine gestellt. Wir weisen Sie darauf hin, dass abgeschriebene Lösungen mit 0 Punkten beurteilt werden.

Geben Sie ein einziges PDF Dokument ab. Erstellen Sie Ihr Abgabedokument computerunterstützt. Wir akzeptieren keine gescannten handschriftlichen PDF-Dateien.

24.05. 06:55 Uhr	Deadline für den Upload über den CourseManager. Es werden keine verspäteten Abgaben akzeptiert.
01.06. 12:00 Uhr	Feedback im CourseManager verfügbar
02.06. 23:59 Uhr	Deadline für die Reservierung eines Termins für das Abgabegespräch

Abgabegespräch

1. Sie müssen sich über den CourseManager zu einem Abgabegespräch (=Prüfungsgespräch) anmelden. Bitte machen Sie das rechtzeitig, je später Sie sich anmelden, umso eingeschränkter ist das Terminangebot. Bis zur oben genannten Deadline garantieren wir Ihnen einen Termin.

2. Sie müssen mindestens einen Punkt auf das Übungsblatt bekommen. Wenn Sie weniger als einen Punkt auf das Blatt bekommen, oder kein Blatt abgegeben haben, sind Sie nicht zum Abgabegespräch zugelassen.

3. Sie kommen mit Ihrem Studierendenausweis zu der von Ihnen reservierten Zeit vorbei, und absolvieren das Abgabegespräch. Stoffgebiet des Abgabegesprächs sind

die mit dem Übungsblatt abgedeckten Themengebiete. Wir setzen voraus, dass Sie sich mit Ihrer korrigierten Abgabe auseinandergesetzt haben.

4. Sie absolvieren Ihr Abgabegespräch gemeinsam mit anderen KollegInnen. Das Gespräch dauert ca. 60 Minuten.

5. Sie können auf die Abgabe maximal 15 Punkte erreichen. Diese setzen sich wie folgt zusammen:

   5 Punkte auf das Übungsblatt

   10 Punkte auf das Abgabegespräch

6. Die Assistenten tragen die Punkte des Abgabegesprächs in den CourseManager ein und Sie sehen dort, wieviele Punkte Sie bekommen haben.

7. Um die Lehrveranstaltung positiv abzuschließen, brauchen Sie mindestens 1 Punkt auf das Übungsblatt und 1 Punkt auf das Abgabegespräch.

8. Falls Sie nicht zu Ihrem Gesprächstermin erscheinen, bekommen Sie automatisch 0 Punkte und damit ein negatives Zeugnis.

SQL

Aufgabe 1 (eSQL) *[1.0 Punkte]*
Lösen Sie alle 10 (inkl. Unterpunkte) unter

`http://minteka.dbai.tuwien.ac.at/eSQL-tutorial/`

zur Verfügung gestellten SQL-Aufgaben des aktuellen Übungskurses. Loggen Sie sich dabei mit dem Usernamen und dem Passwort ein, das Sie bereits vom CourseManager kennen. Der Abschlusstest der Übung wird über dieselbe Plattform abgewickelt. Es empfiehlt sich daher zusätzlich auch mit Beispielen aus den vorigen Semestern zu üben.

Aufgabe 2 (Kaskadierendes Löschen) *[0.5 Punkte]*
Gegeben sei eine Datenbank mit den Relationen `person`, `fahrzeug` und `benutzt`, die wie folgt erstellt wurden:

```
CREATE TABLE person (
        pid INTEGER PRIMARY KEY,
        pname VARCHAR(80)
);

CREATE TABLE fahrzeug (
        fid INTEGER PRIMARY KEY,
        marke VARCHAR(80) NOT NULL,
        besitzer INTEGER REFERENCES person(pid) ON DELETE SET NULL
);
```

```sql
CREATE TABLE benutzt (
        bid INTEGER PRIMARY KEY,
        pid INTEGER REFERENCES person(pid) ON DELETE CASCADE,
        fid INTEGER REFERENCES fahrzeug(fid) ON DELETE CASCADE
);
```

Die Ausprägungen der Relationen seien:

$$\begin{aligned}\text{person} &= \{(1, \text{'Hans'}), (2, \text{'Anna'}), (3, \text{'Max'}), (4, \text{'Hans'})\}\\ \text{fahrzeug} &= \{(1, \text{'VW'}, 1), (2, \text{'Opel'}, 4), (3, \text{'BMW'}, 2), (4, \text{'Audi'}, 4)\}\\ \text{benutzt} &= \{(1, 1, 1), (2, 2, 1), (3, 4, 2), (4, 4, 3), (5, 3, 4), (6, 2, 2)\}\end{aligned}$$

Geben Sie die Ausprägungen an, die sich durch das Ausführen der Statements

(a) `DELETE FROM` person `WHERE` pname='Hans' und

(b) `DELETE FROM` fahrzeug `WHERE` besitzer=4 `OR` marke='BMW'

ergeben, jeweils auf die ursprüngliche Ausprägung angewendet.

Lösung:

(a) person = $\{(2, \text{'Anna'}), (3, \text{'Max'})\}$
fahrzeug = $\{(1, \text{'VW'}, \text{NULL}), (2, \text{'Opel'}, \text{NULL}), (3, \text{'BMW'}, 2), (4, \text{'Audi'}, \text{NULL})\}$
benutzt = $\{(2, 2, 1), (5, 3, 4), (6, 2, 2)$

(b) person = $\{(1, \text{'Hans'}), (2, \text{'Anna'}), (3, \text{'Max'}), (4, \text{'Hans'})\}$
fahrzeug = $\{(1, \text{'VW'}, 1)\}$
benutzt = $\{(1, 1, 1), (2, 2, 1)\}$

Aufgabe 3 (Allquantifizierung) *[0.5 Punkte]*
In einem all-quantifizierten Ausdruck wird z.B.: nach jenen Kunden gesucht, die bereits *alle* angebotenen Pizzen bestellt haben. Diesen Sachverhalt kann man in klassischer Prädikatenlogik erster Stufe wie folgt ausdrücken: $\varphi(k) \equiv \mathit{Kunde}(k) \wedge \forall p\ (\mathit{Pizza}(p) \rightarrow \mathit{bestellt}(k, p))$ Erklären Sie die beiden Methoden, durch die ein Allquantor in SQL ausgedrückt werden kann, zunächst allgemein und dann auch an dem oben angeführten einfachen Beispiel. Geben Sie hierbei die entsprechenden Abfragen in SQL-Syntax an.

```
Kunde(kkey)
Pizza(pkey)
bestellt(Kunde.kkey, Pizza.pkey)
```

Lösung:
Prinzipiell kann ein All-Quantor in SQL mit Hilfe von `COUNT` oder durch `NOT EXISTS`, `NOT IN` Konstrukt ausgedrückt werden.

Bei dem Ansatz mit `COUNT` werden zunächst alle Tupel in der betrachteten Domäne gezählt. Dann werden die Tupel in der betrachteten Domäne gezählt, die überdies die gewünschte Bedingung erfüllen. Stimmen die beiden Ergenisse überein, erfüllen alle betrachteten Tupel die Bedingung. Eine Lösung wäre daher:

```sql
SELECT k.kkey
FROM Kunde k JOIN bestellt b ON b.kkey = k.kkey
GROUP BY k.kkey
HAVING COUNT(*) = (SELECT COUNT(*) FROM Pizza);
```

Der andere Ansatz nutzt die Äquivalenz $\forall x\ \varphi(x) \equiv \neg\exists x\ \neg\varphi(x)$. Wir wollen also all jene Kunden finden, für die es keine Pizza gibt, die sie nicht bestellt haben.

```sql
SELECT k.kkey
FROM kunde k
WHERE not exists (SELECT * FROM Pizza p
                  WHERE p.pkey NOT IN (SELECT b.pkey
                                       FROM bestellt b
                                       WHERE b.kkey=k.kkey))
```

Normalformentheorie

Aufgabe 4 (Armstrong Axiome) *[0.5 Punkte]*

(a) Zeigen Sie, dass folgendes Axiom aus den Armstrong Axiomen hergeleitet werden kann:

Falls $\alpha \to \beta\gamma$ und $\gamma \to \delta\eta$ gelten, dann gilt auch $\alpha \to \eta$.

(b) Gegeben ist ein Relationenschema $\mathcal{R} = ABCDEF$ und zwei Mengen F_1 und F_2 von funktionalen Abhängigkeiten.

$$F_1 = \{A \to BC, C \to D, AC \to C, AC \to D\}$$
$$F_2 = \{A \to BC, C \to D, A \to D\}$$

Sind F_1 und F_2 äquivalent? Begründen Sie Ihre Antwort formal und dokumentieren Sie den Lösungsweg.

Lösung:

(a) Eine mögliche Herleitung aus den drei Basis-Axiomen ist folgende:

$$\dfrac{\dfrac{\overset{[\text{Gegeben}]}{\alpha \to \beta\gamma} \qquad \dfrac{}{\beta\gamma \to \gamma}\,\text{Refl.}}{\alpha \to \gamma}\,\text{Trans.} \qquad \dfrac{\overset{[\text{Gegeben}]}{\gamma \to \delta\eta} \qquad \dfrac{}{\delta\eta \to \eta}\,\text{Refl.}}{\gamma \to \eta}\,\text{Trans.}}{\alpha \to \eta}\,\text{Trans.}$$

Verwendet man das zusätzliche Axiom der Dekomposition, dann ergibt sich als Herleitung:

$$\dfrac{\dfrac{\begin{array}{c}[\text{Gegeben}]\\ \alpha \to \beta\gamma\end{array}}{\alpha \to \gamma}\,\text{Dekomp.} \qquad \dfrac{\begin{array}{c}[\text{Gegeben}]\\ \gamma \to \delta\eta\end{array}}{\gamma \to \eta}\,\text{Dekomp.}}{\alpha \to \eta}\,\text{Trans.}$$

(b) Ja, da $F_1^+ = F_2^+$. Es reicht aus, folgende zwei Eigenschaften zu zeigen:

$F_1 \subseteq F_2^+$: Mithilfe der Armstrong-Axiome zeigen wir, dass $\mathrm{AC} \to \mathrm{C} \in F_2^+$ und dass $\mathrm{AC} \to \mathrm{D} \in F_2^+$. Die restlichen FDs folgen trivial, da $F_1 \setminus \{AC \to C, AC \to D\} \subseteq F_2 \subseteq F_2^+$.

$$\dfrac{}{\mathrm{AC} \to \mathrm{C}}\,\text{Refl.}$$

$$\dfrac{\dfrac{}{\mathrm{AC} \to \mathrm{A}}\,\text{Refl.} \qquad \begin{array}{c}[\text{Gegeben}]\\ \mathrm{A} \to \mathrm{D}\end{array}}{\mathrm{AC} \to \mathrm{D}}\,\text{Trans.}$$

$F_2 \subseteq F_1^+$: Analog zu ersten Fall. Es gilt $F_2 \setminus \{A \to D\} \subseteq F_1 \subseteq F_1^+$. Es bleibt somit zu zeigen: $\mathrm{A} \to \mathrm{D} \in F_1^+$.

$$\dfrac{\dfrac{\begin{array}{c}[\text{Gegeben}]\\ \mathrm{A} \to \mathrm{BC}\end{array}}{\mathrm{A} \to \mathrm{C}}\,\text{Dekomp.} \qquad \begin{array}{c}[\text{Gegeben}]\\ \mathrm{C} \to \mathrm{D}\end{array}}{\mathrm{A} \to \mathrm{D}}\,\text{Trans.}$$

Aufgabe 5 (Kanonische Überdeckung) *[0.5 Punkte]*
Bestimmen Sie eine kanonische Überdeckung folgender Mengen funktionaler Abhängigkeiten über dem Relationenschema $ABCDEF$ und dokumentieren Sie den Lösungsweg.

(a) $F^1 = \{A \to BC,\ B \to DA,\ FB \to CD,\ AD \to F,\ D \to BE\}$

(b) $F^2 = \{A \to E,\ C \to AB,\ ACD \to C,\ E \to BC,\ CD \to CEF\}$

Lösung:

(a) $F_C^1 = \{A \to B,\ B \to ACD,\ D \to BEF\}$
(oder auch $\{A \to BF,\ B \to ACD,\ D \to BE\}$, $\{A \to BC,\ B \to AD,\ D \to BEF\}$, $\{A \to BCF,\ B \to AD,\ D \to BE\}$)

(b) $F_C^2 = \{A \to E,\ C \to A,\ CD \to F,\ E \to BC\}$
(oder auch $\{A \to E,\ C \to AB,\ CD \to F,\ E \to C\}$)

Aufgabe 6 (Schlüsselbestimmung) *[0.5 Punkte]*

(a) Bestimmen Sie für folgendes Relationenschema samt funktionalen Abhängigkeiten alle Schlüssel und alle Superschlüssel.

$$\mathcal{R} = ABCDE$$
$$F = \{A \to B,\ B \to C,\ AD \to E,\ CD \to B,\ E \to D\}$$

Lösung:
Die Schlüssel sind AD und AE. Die Menge der Superschlüssel:

$$\{AD, AE, ABD, ABE, ACD, ACE, ADE, ABCD, ABDE, ABCE, ACDE, ABCDE\}$$

(b) Gegeben sei folgendes Relationenschema samt funktionalen Abhängigkeiten:

$$\mathcal{R} = ABCDEFG$$
$$F = \{CD \to ACD, A \to BC, B \to EF, E \to A\ , G \to ABG\}$$

Erklären Sie, warum A und CDG keine Schlüssel sind.

Lösung:
A is kein Schlüssel, da $\mathsf{AttrHülle}(F, \{A\}) = \{A, B, C, E, F\} \neq \mathcal{R}$. Da nun aber DG der einzige Schlüssel ist, ist CDG nicht minimal und deshalb auch kein Schlüssel.

Aufgabe 7 (Synthesealgorithmus) *[0.5 Punkte]*
Gegeben sei folgendes Relationenschema samt funktionalen Abhängigkeiten:

$$\mathcal{R} = ABCDEF$$
$$F = \{A \to BC,\ C \to DA,\ E \to ABC,\ F \to CD,\ CD \to BEF\}$$

Gesucht ist eine verlustlose und abhängigkeitserhaltende Zerlegung in dritter Normalform. Wenden Sie hierzu den Synthesealgorithmus an und dokumentieren Sie das Ergebnis der einzelnen Schritte. Bestimmen Sie alle Schlüssel von $\mathcal{R}$ und allen Relationen der Zerlegung.

Lösung:

1. Bestimmung der kanonischen Überdeckung:

$$F_c = \{A \to C,\ E \to A,\ F \to CD,\ C \to BEF\}$$

2. Erstelle Relationenschemata für jedes Element von F_c:

Relationenschema	Geltende FDs
$\mathcal{R}_1 = AC$	$F_1 = \{A \to C\}$
$\mathcal{R}_2 = EA$	$F_2 = \{E \to A\}$
$\mathcal{R}_3 = FCD$	$F_3 = \{F \to CD\}$
$\mathcal{R}_4 = CBEF$	$F_4 = \{C \to BEF\}$

3. Bestimmung aller Kandidatenschlüssel von $\mathcal{R}$ bzgl. F_c: A, C, E, F.

   Der Schlüssel A ist in einem der erzeugten Teilschemata enthalten ($\mathcal{R}_1$).

4. Kein Relationenschema ist zu eliminieren.

Ergebnis (Schlüssel sind unterstrichen):

Relationenschema	Geltende FDs
$\mathcal{R}_1 = \underline{A}C$	$F_1 = \{A \to C\}$
$\mathcal{R}_2 = \underline{E}A$	$F_2 = \{E \to A\}$
$\mathcal{R}_3 = \underline{F}CD$	$F_3 = \{F \to CD\}$
$\mathcal{R}_4 = \underline{C}BEF$	$F_4 = \{C \to BEF\}$

Aufgabe 8 (Normalformen) *[0.5 Punkte]*
Gegeben sei folgendes Relationenschema samt funktionalen Abhängigkeiten:

$$\mathcal{R} = ABCD$$
$$F = \{AB \to C,\ A \to D,\ CD \to A\}$$

Geben Sie an, ob $\mathcal{R}$

(a) in dritter Normalform ist,

(b) in Boyce-Codd-Normalform ist,

und begründen Sie Ihre Antworten.

Lösung:
AB und BCD sind die Schlüssel von $\mathcal{R}$. Jede FD erfüllt eine der drei Bedingungen für die dritte Normalform. Auf Grund der beiden FDs $\{A \to D\}$ und $\{CD \to A\}$ ist F nicht in BCNF. Sie sind nämlich weder trivial noch steht links ein Superschlüssel.

Aufgabe 9 (Dekompositionsalgorithmus) *[0.5 Punkte]*
Gegeben sei folgendes Relationenschema samt funktionalen Abhängigkeiten:

$$R = UVWXY$$
$$F = \{UY \to X,\ UX \to WV\}$$

Gesucht ist eine verlustlose Zerlegung in Boyce-Codd-Normalform. Wenden Sie hierzu den Dekompositionsalgorithmus an und dokumentieren Sie das Ergebnis der einzelnen Schritte. Bestimmen Sie alle Schlüssel von $\mathcal{R}$ und allen Relationen der Zerlegung. Ist die Zerlegung abhängigkeitserhaltend? Wenn die Zerlegung nicht abhängigkeitserhaltend ist, geben Sie an, welche Abhängigkeiten verloren gegangen sind.

Lösung:
Der einzige Schlüssel von $\mathcal{R}$ ist UY. Die FD $UX \to WV$ verletzt die BCNF, da sie nicht trivial und UX kein Superschlüssel ist. Zerlege $\mathcal{R}$ daher in:

$\mathcal{R}_1 = UXWV$	$F_2 = \{UX \to WV\}$	Schlüssel: UX
$\mathcal{R}_2 = UXY$	$F_1 = \{UY \to X\}$	Schlüssel: UY

$\mathcal{R}_1$ und $\mathcal{R}_2$ sind beide in BCNF und die Zerlegung ist abhängigkeitserhaltend, da bereits $F = F_1 \cup F_2$ gilt.