2. Übungsblatt

3.0 VU Datenmodellierung

13. November 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.

Deadlines

06.12.	**06:55 Uhr**	Deadline für den Upload über den CourseManager
14.12.	*12:00 Uhr*	Feedback im CourseManager verfügbar
15.12.	*23:59 Uhr*	Reservierung eines Termins für das Abgabegespräch

Abgabegespräch

1. Sie müssen sich über den CourseManager zu einem Abgabegespräch 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. Wenn Sie 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. Falls Sie nicht zu Ihrem Gesprächstermin erscheinen, bekommen Sie automatisch 0 Punkte auf das Gespräch.

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`, `Firma` und `arbeitet`, die wie folgt erstellt wurden:

```
CREATE TABLE Person (
        svnr VARCHAR(20) PRIMARY KEY,
        name VARCHAR(20) NOT NULL
);

CREATE TABLE Firma (
        id INTEGER PRIMARY KEY,
        name VARCHAR(20) NOT NULL,
        land VARCHAR(20) NOT NULL,
        leiter VARCHAR(20) REFERENCES Person(svnr) ON DELETE CASCADE
);

CREATE TABLE arbeitet (
        aid INTEGER PRIMARY KEY,
        svnr VARCHAR(20) REFERENCES Person(svnr) ON DELETE SET NULL,
        id INTEGER REFERENCES Firma(id) ON DELETE CASCADE
);
```

Die Ausprägungen der Relationen seien:

$$\begin{aligned}\text{Person} &= \{(\text{'1'}, \text{'Guenther'}), (\text{'2'}, \text{'Franz'}), (\text{'3'}, \text{'Maria'}), (\text{'4'}, \text{'Mario'})\}\\ \text{Firma} &= \{\,(1, \text{'Sony'}, \text{'Japan'}, \text{'1'}), (2, \text{'Apple'}, \text{'USA'}, \text{'3'}),\\ &\quad (3, \text{'Microsoft'}, \text{'USA'}, \text{'4'}), (4, \text{'OEBB'}, \text{'Oesterreich'}, \text{'2'})\}\\ \text{arbeitet} &= \{\,(1, \text{'1'}, 2), (2, \text{'1'}, 3), (3, \text{'1'}, 4), (4, \text{'2'}, 3), (5, \text{'2'}, 4), (6, \text{'3'}, 3),\\ &\quad (7, \text{'3'}, 4), (8, \text{'4'}, 2)\}\end{aligned}$$

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

(a) `DELETE FROM` Person `WHERE` name = 'Guenther' und

(b) `DELETE FROM` Firma `WHERE` land = 'USA'

ergeben, jeweils auf die ursprüngliche Ausprägung angewendet.

Lösung:

(a) Person = {('2', 'Franz'), ('3', 'Maria'), ('4', 'Mario')}
Firma = {(2, 'Apple', 'USA', '3'), (3, 'Microsoft', 'USA', '4'),
(4, 'OEBB', 'Oesterreich', '2')}
arbeitet = {(1, *null*, 2), (2, *null*, 3), (3, *null*, 4),
(4, '2', 3), (5, '2', 4), (6, '3', 3), (7, '3', 4), (8, '4', 2)}

(b) Person = {('1', 'Guenther'), ('2', 'Franz'), ('3', 'Maria'), ('4', 'Mario')}
Firma = {(1, 'Sony', 'Japan', '1'), (4, 'OEBB', 'Oesterreich', '2')}
arbeitet = {(3, '1', 4), (5, '2', 4), (7, '3', 4)}

Aufgabe 3 (Allquantifizierung) *[0.5 Punkte]*
In einem all-quantifizierten Ausdruck wird z.B.: nach jenen Köchen gesucht, die *alle* angebotenen Speisen zubereiten können. Diesen Sachverhalt kann man in klassischer Prädikatenlogik erster Stufe wie folgt ausdrücken: $\varphi(k) \equiv \mathit{Koch}(k) \wedge \forall s\ (\mathit{Speise}(s) \rightarrow \mathit{zubereiten}(k, s))$. 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.

```
Koch(kkey)
Speise(skey)
zubereiten(Koch.kkey, Speise.skey)
```

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 Ergebnisse überein, erfüllen alle betrachteten Tupel die Bedingung. Eine Lösung wäre daher:

```
SELECT k.kkey
FROM Koch k JOIN zubereiten z ON z.kkey = k.kkey
GROUP BY k.kkey
HAVING COUNT(*) = (SELECT COUNT(*) FROM Speise);
```

Der andere Ansatz nutzt die Äquivalenz $\forall x\ \varphi(x) \equiv \neg\exists x\ \neg\varphi(x)$. Wir wollen also all jene Köche finden, für die es keine Speise gibt, die sie nicht zubereiten können.

```
SELECT k.kkey
FROM Koch k
WHERE NOT EXISTS (SELECT * FROM Speise s
                  WHERE s.skey NOT IN (SELECT z.skey
                                       FROM zubereiten z
                                       WHERE z.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 $\alpha \to \alpha\beta$ und $\beta \to \delta$ gelten, dann gilt auch $\alpha\gamma \to \delta$.

Sie können hierbei auch von der Vereinigungs-, der Dekompositions- und der Pseudotransitivitätsregel Gebrauch machen.

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

$$F_1 = \{A \to B, B \to C, C \to A, E \to D\}$$
$$F_2 = \{B \to A, A \to CB, C \to B, E \to D\}$$

Sind F_1 und F_2 äquivalent? Begründen Sie Ihre Antwort formal und dokumentieren Sie den Lösungsweg. *Hinweis:* Zwei Mengen von funktionalen Abhängigkeiten sind äquivalent, wenn sie dieselbe Hülle besitzen.

Lösung:

(a) Eine mögliche Herleitung aus den drei Armstrong-Axiomen unter der Verwendung der Dekompositionsregel ist folgende:

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

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

$F_1 \subseteq F_2^+$: Mithilfe der Armstrong-Axiome zusammen mit der Vereinigungs- und der Dekompositionsregel zeigen wir, dass $\mathrm{A} \to \mathrm{B} \in F_2^+$, $\mathrm{B} \to \mathrm{C} \in F_2^+$, und $\mathrm{C} \to \mathrm{A} \in F_2^+$.

Die verbleibende FD ($E \to D$) folgt trivial, da $F_1 \setminus \{A \to B, B \to C, C \to A\} \subseteq F_2 \subseteq F_2^+$.

$$\dfrac{\overset{[\text{Gegeben}]}{\mathrm{A} \to \mathrm{CB}}}{\mathrm{A} \to \mathrm{B}}\,\text{Dekomp.}$$

$$\dfrac{\overset{[\text{Gegeben}]}{\mathrm{B} \to \mathrm{A}} \qquad \dfrac{\overset{[\text{Gegeben}]}{\mathrm{A} \to \mathrm{CB}}}{\mathrm{A} \to \mathrm{C}}\,\text{Dekomp.}}{\mathrm{B} \to \mathrm{C}}\,\text{Trans.}$$

$$\dfrac{\overset{[\text{Gegeben}]}{\mathrm{C} \to \mathrm{B}} \qquad \overset{[\text{Gegeben}]}{\mathrm{B} \to \mathrm{A}}}{\mathrm{C} \to \mathrm{A}}\,\text{Trans.}$$

$F_2 \subseteq F_1^+$: Analog zum ersten Fall. Es gilt $F_2 \setminus \{B \to A, A \to CB, C \to B\} \subseteq F_1 \subseteq F_1^+$. Es bleibt somit zu zeigen: $\mathrm{B} \to \mathrm{A} \in F_1^+$, $\mathrm{A} \to \mathrm{CB} \in F_1^+$, und $\mathrm{C} \to \mathrm{B} \in F_1^+$.

$$\dfrac{\overset{[\text{Gegeben}]}{\mathrm{B} \to \mathrm{C}} \qquad \overset{[\text{Gegeben}]}{\mathrm{C} \to \mathrm{A}}}{\mathrm{B} \to \mathrm{A}}\,\text{Trans.}$$

$$\dfrac{\dfrac{\overset{[\text{Gegeben}]}{\mathrm{A} \to \mathrm{B}} \qquad \overset{[\text{Gegeben}]}{\mathrm{B} \to \mathrm{C}}}{\mathrm{A} \to \mathrm{C}}\,\text{Trans.} \qquad \overset{[\text{Gegeben}]}{\mathrm{A} \to \mathrm{B}}}{\mathrm{A} \to \mathrm{CB}}\,\text{Vereinig.}$$

$$\dfrac{\dfrac{[\text{Gegeben}]}{\text{C} \rightarrow \text{A}} \qquad \dfrac{[\text{Gegeben}]}{\text{A} \rightarrow \text{B}}}{\text{C} \rightarrow \text{B}}\ \text{Trans.}$$

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

(a) $F^1 = \{A \to BC,\ B \to CD,\ CD \to CE,\ B \to EF\}$

(b) $F^2 = \{A \to BC,\ BC \to BDE,\ D \to F,\ E \to EG,\ B \to C,\ C \to DE\}$

Lösung:

(a) $F_C^1 = \{A \to B,\ B \to CDF,\ CD \to E\}$

(b) $F_C^2 = \{A \to B,\ B \to C,\ C \to DE,\ D \to F,\ E \to G\}$

Aufgabe 6 (Schlüsselbestimmung) *[0.5 Punkte]*

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

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

Lösung:
Der einzige Schlüssel ist A. Die Menge der Superschlüssel:

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

(b) Gegeben sei folgendes Relationenschema samt funktionalen Abhängigkeiten:

$$\mathcal{R} = ABCDEFGH$$
$$F = \{A \to BF, B \to ACD, D \to BE, G \to H\}$$

Erklären Sie, warum B und ABG keine Schlüssel sind. Berechnen Sie weiters alle Schlüssel.

Lösung:
B is kein Schlüssel, da $\mathsf{AttrHülle}(F, \{B\}) = \{A, B, C, D, E, F\} \neq \mathcal{R}$. Da nun aber BG ein Schlüssel ist, ist ABG nicht minimal und deshalb kein Schlüssel. Die Menge aller Schlüssel ist $\{AG, BG, DG\}$.

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

$$\mathcal{R} = ABCDEF$$
$$F = \{A \to BC,\ B \to CD,\ D \to E,\ A \to EF,\ BF \to A\}$$

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:

$$F_c = \{A \to BF,\ B \to CD,\ BF \to A,\ D \to E\}$$

2. Erstelle Relationenschemata für jedes Element von F_c:

Relationenschema	Geltende FDs
$\mathcal{R}_1 = ABF$	$F_1 = \{A \to BF, BF \to A\}$
$\mathcal{R}_2 = BCD$	$F_2 = \{B \to CD\}$
$\mathcal{R}_3 = ABF$	$F_3 = \{BF \to A,\ A \to BF\}$
$\mathcal{R}_4 = DE$	$F_4 = \{D \to E\}$

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

   Der Schlüssel A ist in zumindest einem der erzeugten Teilschemata enthalten ($\mathcal{R}_1$). Es muss kein weiteres Schema erzeugt werden.

4. Das Schema $\mathcal{R}_3$ kann eliminiert werden, da $\mathcal{R}_3 \subseteq \mathcal{R}_1$ gilt.

Ergebnis (Schlüssel sind unterstrichen):

Relationenschema	Geltende FDs
$\mathcal{R}_1 = \underline{BF}\underline{A}$	$F_1 = \{A \to BF, BF \to A\}$
$\mathcal{R}_2 = \underline{B}CD$	$F_2 = \{B \to CD\}$
$\mathcal{R}_4 = \underline{D}E$	$F_4 = \{D \to E\}$

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

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

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

(a) in dritter Normalform ist,

(b) in Boyce-Codd-Normalform ist,

und begründen Sie Ihre Antworten.

Lösung:
AB, AD und AE sind die einzigen Schlüssel von $\mathcal{R}$. Die FD $B \to C$ ist nicht trivial, B ist kein Superschlüssel von $\mathcal{R}$, und C ist in keinem Schlüssel von $\mathcal{R}$ enthalten. Daher ist $\mathcal{R}$ weder in 3NF noch in BCNF.

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

$$R = ABCD$$
$$F = \{A \to BC,\ C \to D,\ B \to A\}$$

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:
Die Schlüssel von $\mathcal{R}$ sind A und B. Die FD $C \to D$ verletzt die BCNF, da sie nicht trivial und C kein Superschlüssel ist. Zerlege $\mathcal{R}$ daher in:

$\mathcal{R}_1 = CD$	$F_2 = \{C \to D\}$	Schlüssel: C
$\mathcal{R}_2 = ABC$	$F_1 = \{A \to BC,\ B \to A\}$	Schlüssel: A und B

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