

Datenbanksysteme



herausgegeben von der Fachschaft Mathematik/Informatik, umbrochen von Sabine

Aufgabe 1 – ER-Modellierung und Multiple Choice (15 Punkte)

Folgende erweiterten ER-Diagramme (EER-Diagramme) stellen eine Modellierung eines Krankenhaus-Szenarios dar. Die Kardinalitäten sind als Standardkardinalitäten angegeben.

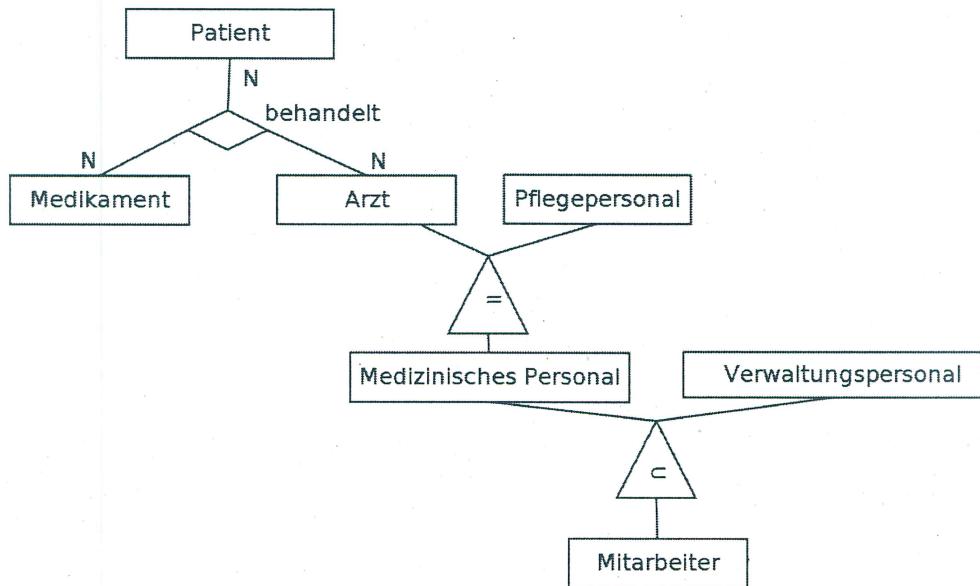


Abbildung 1: ER-Diagramm 1

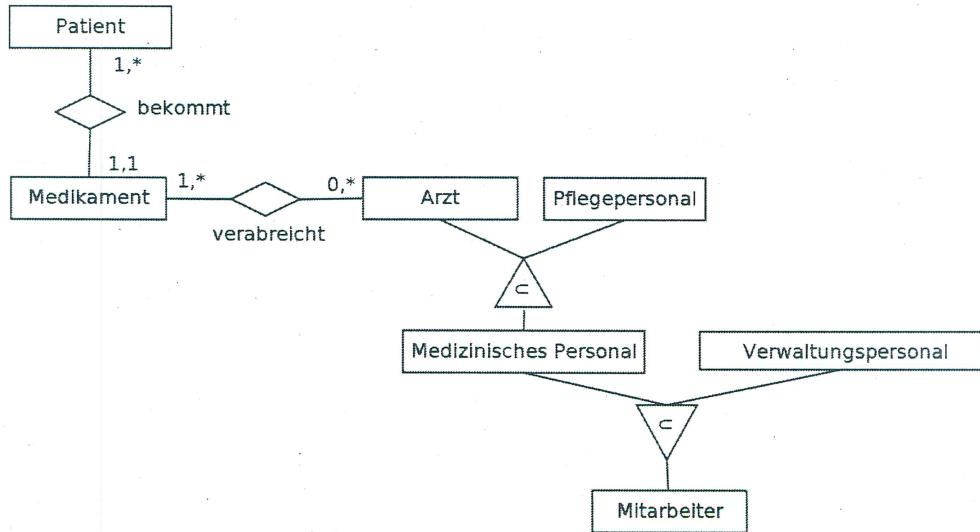


Abbildung 2: ER-Diagramm 2

- a) Nennen Sie die Unterschiede, die die beiden Modellierungen implizieren, und erläutern Sie diese kurz, indem Sie sie an Hand der Modelle veranschaulichen. Wir unsererseits sehen insgesamt sechs nichttriviale Unterschiede.

(10 Punkte)

- b) Geben Sie für folgende 10 Aussagen hier auf dem Aufgabenblatt an, ob die Aussage korrekt oder falsch ist.

Hinweis: Für jede korrekte Antwort erhalten Sie $\frac{1}{2}$ Punkt, für jede falsche Antwort wird Ihnen $\frac{1}{2}$ Punkt abgezogen. Die Aufgabe wird mit mindestens null Punkten bewertet.

	RICHTIG	FALSCH
Bei der Transformation der drei-stelligen Beziehung 'behandelt' im ER-Diagramm 1 entsteht eine Relation, die einen zusammengesetzten Schlüssel aus den Schlüsseln von Patient, Medikament und Arzt enthält.	<input type="checkbox"/>	<input type="checkbox"/>
Die Modellierung im ER-Diagramm 2 erlaubt nur Medikamente, die von mindestens einem Arzt verabreicht wurden.	<input type="checkbox"/>	<input type="checkbox"/>
In der ersten Phase des Zweiphasen-Sperrprotokolls (2PL) darf keine Sperre frei gegeben werden.	<input type="checkbox"/>	<input type="checkbox"/>
Das Zweiphasen-Sperrprotokoll (2PL) vermeidet das Auftreten von Deadlocks.	<input type="checkbox"/>	<input type="checkbox"/>
Ist H (konflikt-)serialisierbar, dann gilt, dass H äquivalent zu einer seriellen History H_S ist.	<input type="checkbox"/>	<input type="checkbox"/>
Eine History kann Lost Updates enthalten.	<input type="checkbox"/>	<input type="checkbox"/>
Die Zuordnung Dichte-erreichbarer Punkte mit DBSCAN ist nichtdeterministisch.	<input type="checkbox"/>	<input type="checkbox"/>
Fremdschlüssel gehören zu den Modellierungskonzepten von ER-Modellen.	<input type="checkbox"/>	<input type="checkbox"/>
Ein Ziel für die Abbildung vom ER-Modell ins Relationenmodell ist die Kapazitätsverminderung.	<input type="checkbox"/>	<input type="checkbox"/>
Gibt es nur einelementige Schlüsselkandidaten, so befindet sich eine Relation mit atomaren Attributen mindestens in 2. Normalform (2NF).	<input type="checkbox"/>	<input type="checkbox"/>

(5 Punkte)

Aufgabe 2 – Funktionale Abhängigkeiten (15 Punkte)

- a) Gegeben sei die Relation $R(A, B, C, D, E, F)$ mit der folgenden Menge an funktionalen Abhängigkeiten:

$$\begin{aligned} F = \{ & A \rightarrow E \\ & E \rightarrow F \\ & A C \rightarrow D, \\ & D \rightarrow B E, \\ & E F \rightarrow A \} \end{aligned}$$

Bestimmen Sie hierfür folgende Mengen:

- Die Menge aller möglichen Schlüssel S
- F_A^+
- $\{A\}_F^*$

(3 Punkte)

b) Gegeben sei die Relation $R(A, B, C, D)$. Geben Sie jeweils eine Menge an funktionalen Abhängigkeiten (F_1, F_2, F_3) an, so dass für R mit der jeweiligen Menge an funktionalen Abhängigkeiten gilt:

- $F_1 : R$ ist in 1NF, aber nicht in 2NF.
- $F_2 : R$ ist in 2NF, aber nicht in 3NF.
- $F_3 : R$ ist in 3NF.

(3 Punkte)

c) Gegeben sei die Relation $R(A, B, C, D, E, F)$ mit der folgenden Menge an funktionalen Abhängigkeiten:

$$\begin{aligned} F = \{ & A \rightarrow B E \\ & F \rightarrow C D, \\ & F \rightarrow B E, \\ & A E \rightarrow B D, \\ & C D \rightarrow B E F, \\ & C F \rightarrow B \} \end{aligned}$$

- Wann ist eine funktionale Abhängigkeit f redundant und wann ist ein Attribut überflüssig im Kontext des Synthesealgorithmus?
- Geben Sie alle überflüssigen Attribute in F an.
- Geben Sie alle redundanten funktionalen Abhängigkeiten in F an.

(4 Punkte)

d) Zeigen oder widerlegen Sie folgende Aussagen über die Dekomposition des Relationenschemas R in X_1 und X_2 und einer Menge an funktionalen Abhängigkeiten F :

$X_1 \cap X_2 \rightarrow X_1 \in F_R^+ \Rightarrow$ Die Dekomposition von R in X_1 und X_2 ist **verbundtreu**.

$X_1 \cap X_2 \rightarrow X_1 \in F_R^+ \Rightarrow$ Die Dekomposition von R in X_1 und X_2 ist **abhängigkeitstreu**.

(5 Punkte)

Aufgabe 3 – SQL (15 Punkte)

Das Fitnesszentrum "Karfitsport" gestaltet seine Datenhaltung mithilfe eines SQL-basierten Datenbanksystems. Dort seien Kunden, Kurse und Abonnements durch folgende Relationen modelliert:

```

CREATE TABLE Kunde
(
    id INTEGER NOT NULL,
    name VARCHAR2(20) NOT NULL,
    alter INTEGER NOT NULL,
    geschlecht CHAR NOT NULL,
    PRIMARY KEY (id)
);

CREATE TABLE Fitnesskurse
(
    id INTEGER NOT NULL,
    name VARCHAR2(20) NOT NULL,
    preis NUMBER NOT NULL,
    dauer NUMBER NOT NULL,
    CONSTRAINT kurse_pk PRIMARY KEY (id)
);

CREATE TABLE Abo
(
    k_id INTEGER NOT NULL,
    f_id INTEGER NOT NULL,
    FOREIGN KEY (k_id) REFERENCES Kunde(id),
    FOREIGN KEY (f_id) REFERENCES Fitnesskurse(id)
);

```

Die Inhalte der Relationen sind wie folgt:

Kunde			
id	name	alter	geschlecht
1	'Arnold Pancino'	25	'M'
2	'Kat Woman'	22	'F'
3	'Jack Chan'	55	'M'
4	'Bat Mädelson'	50	'F'

Fitnesskurse			
id	name	preis	dauer
1	'Tanz-Aerobic'	50	10
2	'Aquafitness'	100	15
3	'Muskelaufbau'	70	20
4	'Yoga'	50	15

Abo	
k_id	f_id
1	1
1	2
1	2
2	2
2	3
3	3

Hinweise:

- alter bezeichnet das Alter eines Kunden (ganzzahlig, von 18 bis 100 einschließlich).
- geschlecht ist das Geschlecht eines Kunden. 'M' steht für Mann, 'F' für Frau.
- preis ist der Preis eines Fitnesskurses, den das Fitnesszentrum anbietet, in Euro (ganzzahlig).
- dauer ist die gesamte Dauer eines Fitnesskurses in Stunden (ganzzahlig).

Alle Aufgaben lassen sich mit SQL-Konstrukten lösen, die in der Vorlesung vorgestellt wurden. Die Lösungen müssen dem SQL-Standard folgen und unabhängig vom Datenbankinhalt sein.

- a) Der Geschäftsführer möchte wissen, wie oft die Kurse des Fitnesszentrums abonniert werden. Erstellen Sie eine Sicht namens 'Teilnehmerzahl', die den Namen jedes Kurses und die Anzahl seiner Teilnehmer ausgibt. Vergeben Sie aussagekräftige Attributnamen.

Hinweis: Bei Kursen ohne Abonnements soll die gelieferte Anzahl der Teilnehmer gleich 0 sein.
(2 Punkte)

- b) Ein Nutzer hat sich beschwert, dass er für einen Kurs den doppelten Preis bezahlen musste. Nach Überprüfung der Datenbank, hat der Geschäftsführer festgestellt, dass beim Erstellen des Abonnements dieses Kunden ein Fehler passiert ist und das Abonnement doppelt eingetragen wurde. Infolgedessen hat er erkannt, dass das derzeitige Schema die folgende Bedingung nicht erfüllt: Jeder Eintrag in der Tabelle 'Abo' soll eindeutig sein. Formulieren Sie die richtige Folge von SQL-Anweisungen, die zuerst die aktuelle Datenbank entsprechend bereinigt und anschließend bewirkt, dass Fehler dieser Art nicht erneut geschehen können.

Hinweis: Sie dürfen im Rahmen dieser Teilaufgabe explizite ID-Werte verwenden.
(2 Punkte)

- c) Das Fitnesszentrum hat entschieden, einen neuen Kurs anzubieten, und möchte hierfür eine Werbekampagne starten. Da der neue Kurs recht innovativ, aufwändig und teuer ist, sollen die Zielkunden dieser Kampagne nur Kunden des Fitnesszentrums sein, die nicht älter als 25 Jahre sind, und die nicht mehr als 150 Euro insgesamt für die Kurse zahlen, die sie bereits abonniert haben. Formulieren Sie eine SQL-Anfrage, die die Namen dieser Kunden ausgibt.

(3,5 Punkte)

- d) Der Geschäftsführer hat eine Beschwerde wegen der zu großen Anzahl an Teilnehmern pro Kurs bekommen. Er möchte deswegen eine Höchstbelegung einführen und diese beim Einfügen von Abonnements automatisch überwachen und gewährleisten. In diesem Sinne erstellt er eine PL/SQL-Prozedur namens 'NeuesAbonnement', mit der überprüft werden soll, ob die Belegung eines Kurses durch Einfügen eines weiteren Abonnements nicht überschritten wird. Dazu soll zunächst in der Variablen 'belegung' berechnet werden, wieviele Teilnehmer den Kurs mit der ID 'fkurs.id' abonniert haben. Falls die Belegung kleiner als die Höchstbelegung 'max_belegung' ist, soll ein Tupel für das neue Abonnement des Kunden mit der ID 'kunden.id' für den Kurs mit der ID 'fkurs.id' eingefügt werden. Andernfalls wird eine entsprechende Meldung ausgegeben und das Abonnement nicht eingefügt.

Hinweis: Bei Kursen ohne Abonnements soll als Anzahl der Teilnehmer 0 angegeben werden. Verwenden Sie den auf dem Lösungsblatt vorgegebenen Rahmen der PL/SQL-Prozedur.

(2,5 Punkte)

- e) Um mit den Kunden einfacher kommunizieren zu können, möchte der Geschäftsführer ihre E-Mail-Adressen erfassen können. Dies ist allerdings im aktuellen Schema nicht vorgesehen. Erweitern Sie das Schema so, dass es möglich ist, die E-Mail-Adressen der Kunden bei der Registrierung in der Tabelle 'Kunde' zu erfassen. Der Kunde soll nicht gezwungen werden, eine E-Mail-Adresse anzugeben und diese soll nicht länger als 25 Buchstaben sein.
- (1 Punkt)
- f) Die Fitnesstrainerin des Yogakurses findet es besser, die Teilnehmer als Paare zu organisieren. Ein solches Paar sollte dazu noch die folgenden Bedingungen erfüllen: (1) Die Teilnehmer eines Paares sollen das gleiche Geschlecht haben, (2) Die Teilnehmer eines Paares sollen ungefähr das gleiche Alter haben, genauer: der Unterschied soll nicht größer als 5 Jahre sein. Formulieren Sie eine SQL-Anweisung, die alle möglichen Paare von Namen verschiedener Kunden liefert, die an dem Kurs 'Yoga' teilnehmen und die oben erwähnten Bedingungen erfüllen. Vermeiden Sie Duplikate.
- (4 Punkte)

Aufgabe 4 – Histories (15 Punkte)

Gegeben seien die folgenden Transaktionen:

$$T_1 = r_1[x] \quad r_1[y] \quad w_1[x] \quad w_1[z] \quad r_1[x] \quad c_1$$

$$T_2 = w_2[x] \quad w_2[y] \quad r_2[z] \quad w_2[z] \quad r_2[x] \quad c_2$$

- a) Finden Sie alle Histories, die sich aus den Transaktionen T_1 und T_2 bilden lassen und folgende Eigenschaften besitzen:
- serialisierbar *und*
 - nicht rücksetzbar (nicht in RC)

Hilfestellung: Überlegen Sie sich zur Lösung dieses Problems, wie eine solche History möglichst einfach erzeugt werden kann. Betrachten Sie dazu beide möglichen Fälle einer serialisierbaren History, also sowohl $T_1 < T_2$ als auch $T_2 < T_1$. Geben Sie für beide Fälle jeweils eine History an, die die geforderten Eigenschaften erfüllt. Wenn Sie eine Lösung gefunden haben, versuchen Sie weitere Lösungen aus der gefundenen durch Modifikationen abzuleiten. Wie viele Histories gibt es insgesamt, die die Eigenschaften erfüllen? Begründen Sie Ihre Antwort (ein explizites Notieren aller Histories ist nicht zwingend verlangt).

(7 Punkte – erläutern Sie Ihren Lösungsweg ausführlich um Teipunkte zu erhalten)

- b) Finden Sie eine History, die sich aus den Transaktionen T_1 und T_2 bilden lässt und folgende Eigenschaften besitzt:
- nicht serialisierbar *und*
 - strikt rücksetzbar (in STRICT)

(7 Punkte – erläutern Sie Ihren Lösungsweg ausführlich um Teipunkte zu erhalten)

- c) Lassen sich aus der in Aufgabenteil b) gefundenen History weitere Lösungen ableiten, die die in Aufgabe b) geforderten Eigenschaften erfüllen? Wie viele solcher Histories lassen sich daraus bilden?
- (1 Punkt)

