SS 2000 Schweppe, Hinze, Kanaeva

Probeklausur zu Einführung in Datenbanksysteme

keine Abgabe

1. Aufgabe *20 Punkte*

Betrachten Sie die folgende Situationsbeschreibung:

> „Ein Informatikinstitut beschließt, endlich Ordnung in seine Software zu bringen. Sie werden beauftragt, eine Datenbank zu entwerfen. Die Datenbank soll auch Anfragen nach inhaltlichen Kriterien zulassen, z.B. „welche Software zur Entwicklung von interaktiven Benutzerschnittstellen gibt es? “. Dazu wird eine zu registrierende Software mit beliebig vielen Indextermen (wie „Benutzerschnittstelle“, „Entwicklung“, „Animation“,...) indexiert. Manche Programme sind abhängig von anderen, z.B. setzt ein WWW-Klient wie Netscape Ghostview - ein Programm zur Anzeige von Postscriptdateien - voraus. Solche Abhängigkeiten sollen darstellbar sein. Daneben soll in der DB aufgezeichnet werden, welche Software bestellt, aber noch nicht geliefert wurde. “

Die Software-Datenbank soll zumindest die in der Aufgabe 2 aufgeführten Anfragen beantworten können. Machen Sie Ihre Annahmen explizit. Verzichten Sie auf Einzelheiten (die in der Praxis evtl. wichtig wären), wenn Sie nicht zur Beantwortung der Fragen der 2. Aufgabe gebraucht werden.

(a) Erstellen Sie ein konzeptionelles Modell der entsprechenden Datenbank (dabei kann man sich hinsichtlich der Attribute an den Anfragen der 2.Aufgabe orientieren). Verwenden Sie zur Modellierung die UML- oder die ER-Schema-Notation, geben Sie die Kardinalitäten für alle Relationen an.

(b) Markieren Sie die Schlüssel der auftretenden Entitäten.

(c) Leiten Sie ggf. daraus ein relationales Schema in der 3. Normalform ab.

2. Aufgabe *15 Punkte*

Formulieren Sie für die Datenbank aus der 1. Aufgabe folgende Anfragen in Tupelkalkül, relationalen Algebra und SQL (machen Sie ggf. geeignete Annahmen):

(a) Welche Software gibt es von Microsoft?

(b) Welche public-domain Programme zur Darstellung von 3D-Objekten gibt es?

(c) Wieviel wurde in diesem Jahr für Software ausgegeben?

(d) Wer hat Software bestellt, die Corba voraussetzt, und mit der Videokonferenzen durchgeführt werden können?

(e) Wieviele Textsysteme gibt es im Institut? (Verschiedene Versionen gelten nicht als verschiedene Systeme.)

(f) Welche Versionen von FrameMaker gibt es im Institut für PC /Windows-NT bzw. für Sun /Solaris

(g) Machen Sie eine Aufstellung der beschafften Software, aus der die Gesamtausgaben nach Gruppen (Systemsoftware, Compiler, Textverarbeitung, Sonstige) hervorgeht.

(h) Wie sieht das Verhältnis zwischen Anzahl public domain SW-Pakete / Anzahl kostenpflichtiger SW- Produkte aus?

(i) Welche SW wurde bestellt (Bestelldatum) aber noch nicht geliefert?

(j) Welche SW gibt es für alle Plattformen (d.h. im Institut verwendete Betriebssysteme)?

3. Aufgabe *6 Punkte*

Gegeben seien eine Relation R(A,B,C,D,E) und eine Menge der funktionalen Abhängigkeiten F = $\{\{A,B\} \rightarrow \{C,E\}, \{B\} \rightarrow \{D\}$ und $\{D\} \rightarrow \{E\}\}$. Welche der folgenden funktionalen Abhängigkeiten werden **nicht** von F impliziert?

(a) $\{A,D\} \rightarrow \{C,E\}$

(b) $\{B,C\} \rightarrow \{D\}$

(c) $\{A,B\} \rightarrow \{E\}$

(d) $\{B\} \rightarrow \{E\}$

4. Aufgabe *6 Punkte*

(a) Wenn ein Index den Zugriff beschleunigt, warum legt man dann nicht für alle Attribute einen Index an?

(b) Auf welche Attribute würden Sie in dem Schema in Aufgabe 1 einen Index legen?

5. Aufgabe *9 Punkte*

Geben Sie für die folgenden Schedules an, ob sie seriell, view-serialisierbar bzw. konfliktserialisierbar sind. Erstellen sie dazu zu jedem nicht-seriellen Schedule einen Präzedenzgraphen.

(a) $r_1(x), r_2(y), r_2(x), w_2(x), r_1(w), w_1(y), c_1, c_2$

(b) $r_2(x), w_3(x), w_1(y), r_2(y), r_1(y), w_3(y), c_1, c_2, c_3$

(c) $r_3(y), r_3(x), r_2(y), r_2(x), w_2(x), r_1(x), r_3(w), w_3(w), r_1(w), w_1(x), w_2(y), r_4(y), w_1(w), w_4(y), c_1, c_2, c_3, c_4$

6. Aufgabe *10 Punkte*

(a) Warum verhindert 2-phase-locking das Auftreten von *cascaded rollbacks* ?

(b) Betrachten Sie die folgende Sequenz:

S1: T1:R(X), T2:W(X), T2:W(Y), T3:W(Y), T1:W(Y), T1,T2,T3: Commit

Beschreiben Sie, wie die Sequenz unter den folgenden Voraussetzungen vom Scheduler behandelt wird: Timestamp der Transaktion Ti sei i, Anwendung des strikten 2PL mit Timestamps bei Verwendung der Wait/Die-Policy.