UNIVERSITÄT DES SAARLANDES
Fachbereich 14– Informatik
Prof. Dr.–Ing. G. Weikum
Dipl.–Inform. A. Kraiß

Datenbanksysteme WS 1998/99

Teilklausur #2 — 22.01.1999

Aufgabe 1: Join-Index (2+8+4=14 Punkte)

Gegeben seien die Relationen *VERKAUF*(*PNr, KNr, Anzahl*) und *KUNDEN*(*KNr, Adresse*). Die unterstrichenen Attribute bilden den Primärschlüssel. Die Relation *VERKAUF* umfasse *10 Mio.* Tupel, wobei jedes Tupel eine konstante Größe von *100 Bytes* habe. Die Relation *KUNDEN* umfasse *1000* Tupel, wobei jedes Tupel eine konstante Größe von *20 Bytes* habe. Die nutzbare Nettogröße einer Seite sei *2000 Bytes*. Zur Speicherung eines Wertes des Attributes *KNr* benötige man *4 Bytes*. Jeder Kunde soll alle Produkte gekauft haben. Zur effizienten Ausführung eines Natural Joins zwischen den beiden Relationen existiere ein Join-Index in Form eines B*-Baumes. Zur Speicherung eines TIDs benötige man *4 Bytes*, und zur Speicherung eines Sohnzeigers benötige man ebenfalls *4 Bytes*.

(a) Bestimmen Sie die Höhe *h* des Join-Index. Ein Index, der nur aus der Wurzel besteht, habe eine Höhe von *1*. Ignorieren Sie den Platzbedarf zum Verketten der Blattseiten.

(b) Nehmen Sie im folgenden an, daß neben den beiden Relationen lediglich der Join-Index aus (a), jedoch mit einer Höhe von *h=4*, existiere. Bestimmen Sie die Anzahl der Seitenzugriffe für die folgenden Join-Ausführungsstrategien zur Ausführung des Natural Join zwischen den Relationen *VERKAUF* und *KUNDEN*. Nehmen Sie an, daß den Joins ein Puffer zum Speichern von nur *2 Datenseiten* und *1 Indexseite* zur Verfügung stehe, und daß jedes Verfolgen eines Sohnzeigers bzw. eines TIDs zu einem neuen Seitenzugriff führt.

(i) Nested-Block-Join (ohne Index).

(ii) Nested-Loop-Join mit *VERKAUF* in der äußeren Schleife und Indexzugriff in der inneren Schleife. Gehen Sie hierbei von einer für diesen Join günstigen Speicherungsreihenfolge der TIDs aus.

(iii) Nested-Loop-Join mit *KUNDEN* in der äußeren Schleife und Indexzugriff in der inneren Schleife.

(iv) Join durch Scan des Index.

(c) Sei *h* die Höhe des Join-Index mit $h>2$, und sei *p* die Anzahl der Blattseiten pro Schlüsselwert mit $p>1$. Die Anzahl der Tupel in der Relation *KUNDEN* sei durch t_K gegeben, und die Anzahl der Tupel in der Relation *VERKAUF* sei durch t_V gegeben. Weiterhin sei die Anzahl der Seiten für die Relation *KUNDEN* durch p_K gegeben. Es gelte: $h \leq t_K$.

Zeigen Sie: Die Anzahl der Blockzugriffe bei Ausführung eines Nested-Loop-Joins mit *KUNDEN* in der äußeren Schleife und dem Zugriff auf den Join-Index in der inneren Schleife kann niemals niedriger sein als bei Ausführung des Joins durch Scan des Join-Index.

Lösung zu a)

Anzahl Schlüssel im Index: $|KUNDEN| = 1000$

Für Fanout *f* gilt: $2000\ Bytes \geq |TID| + (f-1) * (|TID| + |\ Kunden\#|)$

$$\rightarrow 2000 \geq 4 + (f-1) * 8 \rightarrow f = \left\lfloor \frac{2000-4}{8} + 1 \right\rfloor = 250$$

Höhe h: $h = \lceil log_{250} 1000 \rceil + 1 = 3$

Lösung zu b)

Relation *VERKAUF*: *10 Mio. Tupel* à *100 Bytes* $\rightarrow \frac{10\ Mio. * 100\ Bytes}{2000\ Bytes} = 500.000\ Datenseiten$

Relation *KUNDEN*: *1000 Tupel* à *20 Bytes* $\rightarrow \frac{1000 * 20\ Bytes}{2000\ Bytes} = 10\ Datenseiten$

Für jeden Kunden existieren *10.000* Tupel in *VERKAUF*.

Größe eines Index-Eintrags: |Kunden#|+|TID|+10.000*|TID|= *40.008 Bytes.*

Anzahl Seiten pro Blatt: $\lceil \frac{40.008}{2000} \rceil = 21$ Seiten

Anzahl Blattseiten: *1000*21= 21.000 Seiten.*

- **Nested-Block-Join**: $10 + 10 * 500.000 = 5,00001 * 10^6$
- Nested-Loop mit *VERKAUF* in äußerer Schleife und Indexzugriff:
  Annahme: KUNDEN-TID ist in erster Blattseite gespeichert:
  $500.000\ Seiten + 10\ Mio.\ Tupel * (h + 1\ Seite) = 50,5 * 10^6$
- Nested-Loop mit *KUNDEN* in äußerer Schleife und Indexzugriff:
  $10\ Seiten + 1000\ Tupel * (h + 20\ Blattseiten + 10.000\ Tupel) = 10,02401 * 10^6$
- Scan des Join–Index
  (h–1)+21.000 Blattseiten + 1000 Datenseiten + 10 Mio. Datenseiten = $10,022003 * 10^6$

Lösung zu c):

Seitenzugriffe beim Nested-Loop-Join mit Indexzugriff: $p_K + t_K * \left((h-1) + p + \frac{t_V}{t_K} \right)$

Seitenzugriffe beim Index–Join: $(h-1) + t_k * p + t_k + t_V$

Annahme: Die Anzahl der Seitenzugriffe beim Nested-Loop-Join mit Indexzugriff ist niedriger als beim Index-Join, d.h. es gelte:

$$p_K + t_K * \left((h-1) + p_l + \frac{t_V}{t_K} \right) < (h-1) + t_k * p_l + t_k + t_V$$

Daraus folgt:

$$p_K + t_K * (h-1) < (h-1) + t_k$$

$$p_K < (2-h) * t_K + h - 1$$

Wegen $h \leq t_K$ gilt außerdem:

$$p_K < (3-h) * t_K - 1$$

Aus $h>2$ folgt damit: $p_K < 0$ (Widerspruch)

Aufgabe 2: Algebraische Transformation (8 Punkte)

Gegeben seien die Relationen $R_1(\underline{A},B,C)$, $R_2(\underline{A},B,C)$ und $S(\underline{D},F)$ sowie die View $R(\underline{A},B,C)=R_1 \cup R_2$. Die unterstrichenen Attribute bilden den Primärschlüssel. Es gelten die folgenden Integritätbedingungen:

I1: $R(\underline{A},B,C)= R_1 \cup R_2$

I2: $\sigma_{[A>10]}(R_1) = R_1$

I3: $\sigma_{[A \leq 10]}(R_2) = R_2$

Die Relationen R_1 und R_2 bilden eine sogenannte "horizontale" Partitionierung der View R, die z.B. einer Einteilung in "billige" und "teure" Produkte entsprechen könnte.

Das Attribut F sei Fremdschlüssel in der Relation S auf die Relation R_2, d.h. es gelte:

I4: $\pi_{[A]}(R_2) \supseteq \pi_{[F]}(S)$

Gegeben sei die folgende Query:

$$\pi_{[F]}(\sigma_{[A \leq 10]} (R \,|x|_{[A=F]}\, S))$$

Transformieren Sie die Query so, daß unnötige Operationen eliminiert und essentielle Operationen auf möglichst kleinen Operanden ausgeführt werden. Berücksichtigen Sie neben gegebenen und ableitbaren Integritätsbedingungen die folgenden allgemeinen Regeln:

R1: $\pi_{[R.A]} (R \,|x|_{[R.A=S.B]}\, S) = \pi_{[R.A]} (R)$

wenn $\pi_{[R.A]} (R) \subseteq \pi_{[S.B]} (S)$

R2: $\sigma_{[P1]} (R \,|x|_{[P2]}\, S) = (\sigma_{[P1]} (R)) \,|x|_{[P2]}\, S$, wenn P1 nur Attribute aus R enthält

$\sigma_{[P1]} (R \,|x|_{[P2]}\, S) = R \,|x|_{[P2]}\, (\sigma_{[P1]} (S))$, wenn P1 nur Attribute aus S enthält

R3: $\sigma_{[P]} (R_1 \cup R_2) = \sigma_{[P]} (R_1) \cup \sigma_{[P]} (R_2)$

R4: $(R_1 \cup R_2) \,|x|_{[P]}\, S = (R_1 \,|x|_{[P]}\, S) \cup (R_2 \,|x|_{[P]}\, S)$

R5: $R \,|x|_{[P]}\, S = S \,|x|_{[P]}\, R$

Wichtig: Geben Sie bei jedem Transformationsschritt an, welche der Regeln Sie angewendet haben.

Lösung:

	$\pi_{[F]}(\sigma_{[A \leq 10]}(R \,|x|_{[A=F]}\, S))$
[I1]	$\pi_{[F]}(\sigma_{[A \leq 10]}((R_1 \cup R_2) \,|x|_{[A=F]}\, S))$
[R4]	$\pi_{[F]}(\sigma_{[A \leq 10]}((R_1 \,|x|_{[A=F]}\, S) \cup (R_2 \,|x|_{[A=F]}\, S)))$
[R3]	$\pi_{[F]}(\sigma_{[A \leq 10]}(R_1 \,|x|_{[A=F]}\, S) \cup \sigma_{[A \leq 10]}(R_2 \,|x|_{[A=F]}\, S))$
[R2]	$\pi_{[F]}((\sigma_{[A \leq 10]}(R_1) \,|x|_{[A=F]}\, S) \cup (\sigma_{[A \leq 10]}(R_2) \,|x|_{[A=F]}\, S))$
[I2,I3]	$\pi_{[F]}((\emptyset \,|x|_{[A=F]}\, S) \cup (R_2 \,|x|_{[A=F]}\, S)) = \pi_{[E]}(R_2 \,|x|_{[A=F]}\, S)$
[R1,I4]	$\pi_{[F]}(S)$