

Übungsblatt Optimizer/Join

Dr. Karsten Tolle

Aufgabe 1:

Es seien die beiden folgenden Relationen gegeben:

- *KUNDENDATEN (NAME VORNAME ID)*
- *KONTODATEN (ID KONTO SALDO DISPO)*

Die Relationen weisen folgende Eigenschaften auf:

- *Es gibt 20000 Kundendatensätze.*
- *Es gibt 40000 Konten.*
- *25 Kundensätze passen in einen Datenblock.*
- *30 Kontensätze passen in einen Datenblock.*
- *Die Daten liegen dicht gepackt auf der Platte, jeweils sortiert nach dem Schlüssel.*
- *Im Hauptspeicher stehen 101 Blöcke zum Laden der Hauptdateidaten zur Verfügung.*

Bestimmen Sie die Anzahl der Blockzugriffe für den natürlichen Verbund zwischen KUNDENDATEN und KONTEN (`SELECT * FROM Kundendaten ku, Kontodaten ko WHERE ku.ID = ko.ID`) für die folgenden Verfahren. Ergebnisse werden einfach ausgegeben und müssen im Hauptspeicher nicht berücksichtigt werden, evtl. Zwischenergebnisse müssen jedoch auf die Platte geschrieben werden.

- Nested-Loop Join mit Nested Blocks*
- Nested-Loop Join mit Rocking*
- Sort-Merge Join (beachte Externes Sortieren! Verfahren selbst festlegen.)*
- Simple-Hash Join (hier können/müssen Annahmen getroffen werden, diese bitte mit nennen)*

Aufgabe 2:

Folgende Situation:

Die Tabelle 'myTable' mit den Attributen a, b und c hat 1.000.000 Einträge und einen Index auf Attribut 'a' (nicht Schlüssel-Attribut). Die Anfrage

```
SELECT a,b,c FROM myTable WHERE a > 19850
```

benötigt 10 Minuten um ausgeführt zu werden, was ungewöhnlich lange ist. Woran kann dies liegen?

Aufgabe 3:

Was ist der *Selectivity Factor*? Wie wird dieser berechnet, welche Werte kann er annehmen und wie wirkt er sich auf die Optimierung aus?