

Datenbanken und Informationssysteme

Aufgabenblatt 4

Präsenzaufgabe 8: Schlagen Sie eine Vorgehensweise zur möglichst objektiven Bewertung des **Leistungsverhaltens von Datenbankmanagementsystemen** vor! Diskutieren Sie in der Übung die unterschiedlichen Vorschläge!

Übungsaufgabe 9: Lesen Sie im O'Neil 00 den Abschnitt zum **TPC-A Benchmark** im dortigen Kapitel „Update Transactions“! (Alternativ dazu können Sie auch die zugehörige Originalspezifikation des Benchmarks konsultieren, die Sie in unserer E-Learning-Plattform finden.)

Übungsaufgabe 10: Informieren Sie sich über **Batch Updates** in JDBC. Können diese bei der folgenden Praktikumsaufgabe gewinnbringend eingesetzt werden?

Praktikumsaufgabe 7: Zur Durchführung von Leistungsmessungen definieren wir eine einfache, skalierbare **Benchmark-Datenbank** aus dem Anwendungsbereich des Bankgewerbes. Die Datenbank besitze nur die folgenden vier Relationen:

- ACCOUNTS mit Informationen über Bankkonten
- BRANCHES mit Informationen über die Zweigstellen der Bank
- TELLERS mit Informationen über Geldautomaten
- HISTORY mit Protokolldaten zu Kontobewegungen.

Entsprechende SQL-Schemadefinitionen geben wir fest vor:

```
create table branches
(
    branchid    int not null,
    branchname  char(20) not null,
    balance     int not null,
    address     char(72) not null,
    primary key (branchid)
);

create table accounts
(
    accid       int not null,
    name        char(20) not null,
    balance     int not null,
    branchid    int not null,
    address     char(68) not null,
    primary key (accid),
    foreign key (branchid) references branches );
```

```

create table tellers
(
    tellerid    int not null,
    tellername  char(20) not null,
    balance     int not null,
    branchid    int not null,
    address     char(68) not null,
    primary key (tellerid),
    foreign key (branchid) references branches );

create table history
(
    accid       int not null,
    tellerid    int not null,
    delta       int not null,
    branchid    int not null,
    accbalance  int not null,
    cmmnt       char(30) not null,
    foreign key (accid) references accounts,
    foreign key (tellerid) references tellers,
    foreign key (branchid) references branches );

```

Sei $n \in \mathbf{N}$ ein vorgegebener Skalierungsfaktor. Eine initiale **n-tps-Datenbank** (diese zunächst merkwürdige Bezeichnung tps für "transactions per second" soll erst später erläutert werden!) enthalte dann die folgenden Tupel:

- n Tupel in der BRANCH-Relation mit fortlaufender BRANCHID (1 bis n), der BALANCE 0 und Strings der richtigen Länge für BRANCHNAME und ADDRESS,
- $n * 100000$ Tupel in der ACCOUNTS-Relation mit fortlaufender ACCID (1 bis $n * 100000$), dem Kontostand (BALANCE) 0, einer *zufälligen* BRANCHID (1 bis n) und wieder beliebigen Strings der richtigen Länge für NAME und ADDRESS,
- $n * 10$ Tupel in der TELLER-Relation mit fortlaufender TELLERID (1 bis $n * 10$), der BALANCE 0, einer *zufälligen* BRANCHID (1 bis n) und wieder beliebigen Strings der richtigen Länge für TELLERNAME und ADDRESS,
- 0 Tupel in der HISTORY-Relation.

- a) Entwickeln Sie ein Programm, das einen Aufrufparameter n erwartet und eine initiale n -tps-Datenbank auf dem gewählten Datenbankmanagementsystem erzeugt.
- b) Welche Mindestgrößen schätzen Sie für eine 1-tps-Datenbank bzw. allgemein für eine n -tps-Datenbank? Wie viel Plattenplatz wird auf dem Datenbank-Server *tatsächlich* für die erstellten Datenbanken benötigt?
- c) Versuchen Sie, die Laufzeit Ihres Programms zu beschleunigen! Dokumentieren Sie einzelne Verbesserungsideen und die jeweiligen Laufzeitveränderungen für eine *lokale* Ausführung Ihres Programms bei der Erzeugung einer 10-tps-Datenbank!
- d) Messen und protokollieren Sie die Laufzeit ihres optimierten Programms für $n=10$, $n=20$ und $n=50$ sowohl lokal auf dem Datenbank-Server als auch von einem "remote" Client! (Gemessen werden soll nur die Laufzeit zum Einfügen ohne evtl. notwendige vorherige DROP-TABLE- oder CREATE-TABLE-Anweisungen!)

Senden Sie eine **Dokumentation Ihrer Bearbeitung** zusammen mit dem gut kommentierten Programm als *eine* pdf-Datei innerhalb unserer E-Learning-Plattform ein!