**Praktikumsaufgabe 7**

**Aufbau Benchmark-Datenbank mit  
Leistungsmessungen**

Fach: Datenbanken und Informationssysteme

Semester: Wintersemester 15/16

Von: André Schlüß, Johannes Nowack, Timo Knufmann

**Teilaufgabe b): Schätzung der Größe für n-tps Datenbank**

Um eine ungefähre Schätzung der Größe der Datenbank vornehmen zu können, wurden als erstes die Größen der einzelnen Datentypen bestimmt. Diese wurden mittels der Dokumentation des Datenbanksystems ermittelt.

Folgende Datentypen und –größen sind von Relevanz:

* Integer: 4 Byte
* Char-String: 2 Byte pro Character. In den verwendeten Strings wurde UTF-8 als Kodierung gewählt. UTF-8 kann 1 – 4 Bytes groß sein. Es wird davon ausgegangen, dass nur Buchstaben, Zahlen und einfache Sonderzeichen verwendet werden. Diese sind in der „Basic Multilingual Plane“ von UTF-8 vorhanden, welche 2 Byte groß ist.

Diese Größen wurden mit der entsprechenden Anzahl in den jeweiligen Tabellen verrechnet, sodass sich die Größe pro Datensatz in einer Tabelle ergab. Anschließend wurde die Anzahl der Datensätze für n in den jeweiligen Tabellen mit einbezogen. Daraus ergibt sich folgende Tabelle:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Ein Datensatz: | Anzahl pro n |  | Größe für n: |
| Tabelle branches: |  | 100 | 1 |  | 100 |
| Tabelle accounts: |  | 100 | 100000 |  | 10000000 |
| Tabelle tellers: |  | 100 | 10 |  | 1000 |
| Tabelle history: |  | 50 | 0 |  | 0 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| n= | 1 |  |  | Gesamt in Kilobytes: | 9766,69922 |

Wie unten rechts zu lesen, liegt die geschätzte Größe bei ca. **9.800 Kilobytes** für n = 1.

**Vergleich mit tatsächlicher Größe:**

Die tatsächliche Größe der Datenbank beträgt ca. **14.440 Kilobytes**. Somit ist die tatsächliche Größe um ca. 4.600 Kilobytes kleiner.

Bei der Ermittlung der Größe wurde als Engine InnoDB verwendet. Nach Umstellung auf MyISAM betrug die Größe der Datenbank laut dem DBMS nur noch 11.880 Kilobyte.

Dadurch, dass sich der Speicherplatzbedarf verändert hat, kann man schließen, dass die beiden Engines die Datenbank unterschiedlich verwalten und daher unterschiedlich viel zusätzlichen Speicher benötigen.

Implizit bedeutet das, dass generell zusätzlicher Speicher für die Verwaltung der Datenbanken benötigt wird, weshalb die tatsächliche Größe der Datenbank letztlich von der geschätzten abweicht.

Diese Verwaltung schließt beispielsweise die Benutzer, Berechtigungen, Verwaltung der Primär- und Sekundärschlüssel usw., ein.

**Teilaufgabe c): Laufzeitoptimierungen**

* Erste Messungen auf Entwicklermaschinen: ca. 1675 Sekunden für n = 10

**Optimierung 1: Verwendung von „Prepared Statements“:**

* Idee: Datenbank kennt den SQL-Ausdruck, nur noch Werte müssen übertragen werden
* Effekt:
  + Weniger Netzlast
  + Überprüfung der Syntax erfolgt nur noch einmal und nicht bei jedem SQL-Statement
* Zeit für n = 10 auf gewertetem System: 1215,83 Sekunden

**Optimierung 2: Deaktivierung von „Auto-Commit“ im Quelltext:**

* Idee: DBMS verwendet Zwischenspeicher, sodass Änderungen noch nicht eingetragen werden. Dieser muss für jeden Commit neu angelegt werden. Beim Ausführen des Commits werden die Änderungen erst übernommen.
* Effekt: Durch Deaktivierung muss die Zwischenspeicherung nur einmalig für einen Haufen an Statements in die DB übernommen werden. Dadurch soll die Verarbeitung beschleunigt werden.
* Zeit für n = 10 auf gewertetem System: 412,567 Sekunden

**Optimierung 3: Änderungen an Datenbankeinstellungen**

* Idee: Veränderungen von Parametern der Datenbank, beispielsweise verschiedene Puffergrößen, Dateigrößen oder Deaktivierung des Transaktionslogs.
* Effekt: Dank mehr Ressourcen, die zur Verfügung stehen, können Vorgänge schneller bearbeitet werden.
* Zeit für n = 10 auf gewertetem System: 209,089