

Übungen zur Vorlesung Datenstrukturen Prof. R. Bayer, WS 2001/02

Übung 1: Kap.1.1: analysieren Sie eine Bank Anwendung mit folgenden Kennzahlen:

Die Bank hat 2 Millionen Kunden, ein Kunde führt durchschnittlich 3 Buchungstransaktionen pro Tag aus, die überwiegend während der Geschäftszeit von 8 Stunden anfallen.

Um Spitzenbelastungen abzufangen soll der Computer der Bank im Normalbetrieb nur zu 40 % ausgelastet sein.

Pro Buchung fallen 150 Bytes Daten an, diese müssen in einer Datenbank für den laufenden Monat und für die zurückliegenden zwei Monate online sein, um Kontoauszüge erstellen und Reklamationen schnell bearbeiten zu können.

Dann müssen alle Buchungsdaten für das Finanzamt und die Bankaufsicht 10 Jahre lang aufbewahrt werden.

Führen sie bitte folgende Analysen durch:

1. Für welche maximale Transaktionslast (t/s = Transaktionen/Sekunde) muß der BankComputer ausgelegt sein?
 Recherchieren Sie im Internet, welche Kombinationen aus Rechner und Datenbanksystem heute solche Transaktionsraten erbringen können: $2 \cdot 10^6 \cdot 3 \text{ t/Tag} = 2 \cdot 10^6 \cdot 3 \text{ t/8 hours} = 6 \cdot 10^6 / 8 \cdot 60 \cdot 60 \text{ t/s} = 10^6 \cdot 4800 \text{ t/s} = 10^4 \cdot 48 \text{ t/s} = 208,33 \text{ t/s}$
 bei 40% Auslastung ergibt sich die Rohleistung zu $208/0,4 = 521 \text{ t/s}$
2. Wie groß wird die online Datenbank? **Bei 20 Geschäftstagen/Monat:**
 $2 \cdot 10^6 \cdot 3 \text{ t/day} \cdot 20 \text{ day/month} \cdot 3 \text{ months} \cdot 150 \text{ B/t} = 6 \cdot 20^3 \cdot 150 \cdot 10^6 \text{ B} = 540 \cdot 10^8 \text{ B} = 54 \text{ GB}$
3. Wie groß wird das Archiv? **18 GB/Monat * 120 Monate/10 Jahre = 2,160 TB / 10 Jahre**
4. Was sind die reinen HW Kosten mit heutigen Festplatten? **Bei derzeitigen (Herbst 2001) Plattenkosten von ca 50 DM/GB ergeben sich 2160*50 DM = 108 TDM**

1

Übung 2: Wie lautet das von OMNIS/Elektra erzeugte SQL Statement für die OMNIS Anfrage:

Gray & Reuter & transaction%

select My11.Doc# **from**

My0 My01, My1 My11, My1 My12, My0 My02, My1 My13, My1 My14, My0 My03

where My01.Wort = ,Gray' **and**
 My01.W# = My11.W# **and**
 My02.Wort = ,Reuter' **and**
 My02.W# = My12.W# **and**
 My03.Wort LIKE ,transaction%' **and**
 My03.W# = My13.W# **and**
 My11.Doc# = My12.Doc# **and**
 My12.Doc# = My13.Doc#

Übung 3: Machen Sie einen Vorschlag für ein relationales OMNIS Schema zur Unterstützung von Stopwörtern aber ohne Nachbarschaftssuche

2

Bisheriges Schema für Text Repräsentation für Freitext Recherche:

My0 (Wort, W#) **My1** (W#, Doc#, Pos)

Schema für Stop-Wörter:

My0 (Wort, W#) enthält alle nicht-stop Wörter

My1 (W#, Doc#) enthält Vorkommen von nicht-stop Wörtern,
Tabelle wird dehalb erheblich kleiner, auch wegen Duplikaten

MyS (StopWort, W#) enthält alle Stop-Wörter

Einfügen neues Stop-Wort: **insert xx in MyS** mit alter oder neuer W#;
 delete from My0;
 delete from My1

Query-Transformationen: Stop-Wörter müssen entfernt werden, ist schwierig!

3

Übung 4: Entwerfen Sie einen Algorithmus, um aus einer OMNIS Datenbank ein vollständiges Dokument mit 5000 Wörtern zu löschen und schätzen Sie den Zeitaufwand bei einer 10 GB großen My1 Relation

Finde Doc# des zu löschenden Dokumentes, z.B. 1253

Lese My1 sequentiell und entferne alle Triple mit Doc# = 1253:

```
delete
from My1 where Doc# = 1253
```

Aufwand: 10 GB sequentiell zu lesen dauert bei 1 MB effektiver Transferrate bei sequentiellem Lesen und Neuschreiben mit entsprechender Pufferung (muß von DB entsprechend unterstützt werden)

ca 10 GB * 1s/MB = 10¹⁰ B * 1s/10⁶ B = 10⁴ s = 167 Min ~ 3 Stunden

Aufwand: 10 GB random zu lesen und wieder zu schreiben dauert bei 250 KB/s effektiver Transferrate werden)

ca 10 GB * 4 s/MB * 2 (Faktor 2 wegen lesen und schreiben) = 8 * 167 Min ~ 24 Stunden ~ 1 Tag

Aufwand mit Sekundärindex auf My1 auf Attribut Doc#: Es müssen maximal 5000 Seiten random gelesen und neu geschrieben werden, bei 10 ms/Seite also

10.000 * 10 ms = 100.000 * 10⁻³ s = 100 s < 2 Minuten

4

Übung 5: Kap. 1.2 Beschreiben Sie den Archivierungs-Prozeß auf Folie 19 als Ereignis/Aktionsstruktur, als Agenten-Programm, als Petri-Netz

ist nicht ausgearbeitet

Übung 6: Kap. 1.4 Ergänzen sie die Klassen Dokt und Stud auf Folie 21 um das Attribut *Abiturnote* und passen Sie die Hierarchie in KNF2 an

siehe spätere Folien

Übung 7: Konstruieren Sie für die Hierarchie auf Folie 21 die relationalen Schemata für die logischen Modelle V, Z und T

siehe spätere Folien

Übung 8: Kap. 1.6: Was ist die Maximalgröße einer UNIX Datei bei Verwendung von Blockgröße 4 KB, ohne Änderung der i-Node Struktur und bei 4 Bytes pro Blockadresse?

Die Anzahl der Datenblöcke ist:

auf Ebene 1 = 10

auf Ebene 2 = 1000

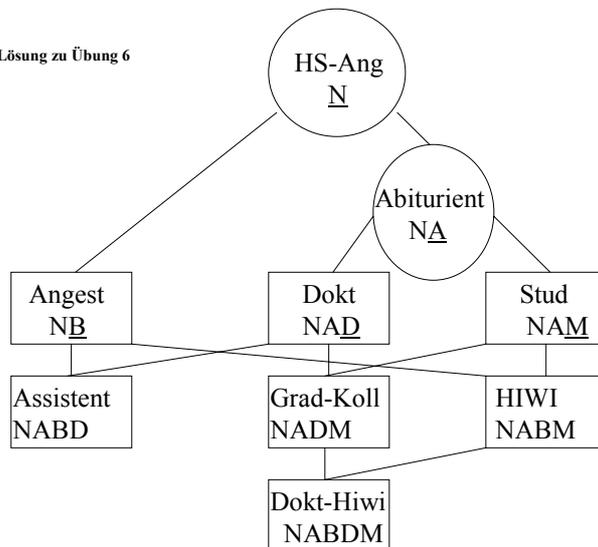
auf Ebene 3 = 10^6

auf Ebene 4 = 10^9

insgesamt also ca 10^9 Datenblöcke * $4 \cdot 10^3$ B/Datenblock = $4 \cdot 10^{12}$ B = 4 TB

5

Lösung zu Übung 6



6

Lösung zu Übung 7:

Modell V: genau 1 Relation pro vorgegebener Klasse,
z. B. Dokt mit allen zugehörigen Attributen
(private plus geerbte) plus OID

relation Dokt
(OID : integer,
N : string,
D : string) **key is** OID

relation Hiwi
(OID : integer,
N : string,
B : string,
M : integer) **key is** OID

Etc.

7

Fortsetzung Lösung zu Übung 7

Modell Z: genau 1 Relation pro Owner-Klasse,
mit neu definierten Attributen und plus OID

relation HS-Ang
(OID : integer,
N : string,) **key is** OID

relation Angest
(OID : integer,
B : string) **key is** OID

relation Dokt
(OID : integer,
D : string) **key is** OID

relation Stud
(OID : integer,
M : integer) **key is** OID

Die weiteren Relationen werden als views definiert, z.B.

define view HIWI as
select Stud.OID, N, B, M **from** HS-Ang, Angest, Stud
where HS-Angest.OID = Angest.OID
and Angest.OID = Stud.OID

8

Modell T:

Speicherung als Relation

relation T

(OID: **integer**,

Attribut: **string**,

Value: **string**) key is (OID, Attribut)

T enthält z.B. Tripel der Form:

(17, „N“, „Schmitt“)

(17, „A“, „1.9“)

(17, „D“, „UB-Bäume und XML“)