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

Datenbanksysteme WS 1998/99

Praktische Übung 4: Physischer Datenbankentwurf
Zu bearbeiten bis: 10.02.1999

Anwendung: Videothek

Herr Glotzkist, der Betreiber einer modernen Videothek, arbeitet mit dem Datenbankschema, das Sie aus der dritten praktischen Übung kennen. Er hat jedoch ein schwerwiegendes Performance-Problem mit seinen Datenbankanwendungen. Ein Freund von ihm, der leider nicht in Saarbrücken studiert hat, hatte Herrn Glotzkist für seine Anwendung das Schema erzeugt und die entsprechenden Anwendungsprogramme geschrieben. Der Normalbetrieb hat ergeben, daß die Anfragen, wie sie in den Anwendungsmodulen anfallen, nicht optimal vom physischen Datenbankentwurf unterstützt werden. Da der Freund von Herrn Glotzkist mittlerweile als Schaafzüchter arbeitet, wendet sich Herr Glotzkist nun an *SIE* !

Hinweise zu Schema und Datenbank

Das Schema der Videotheks-Datenbank ist auf dem Beiblatt *Relationenschema der Videothek* gegeben. Die Primärschlüssel-Attribute sind unterstrichen. Eine Analyse der Semantik der Anwendung sowie des bisherigen Inhaltes der Datenbank ergibt folgende Zusatzinformationen:

(1) Relation KUNDEN

Die Relation Kunden enthält 40'000 Einträge, wobei nur die Hälfte der Kunden unterschiedliche Namen tragen und diese Namen unter den Kunden gleichmäßig verteilt sind. In der Relation kommen außerdem ca. 1000 verschiedene Orte mit unterschiedlichen Postleitzahlen sowie ca. 1300 verschiedene Strassen vor, die ebenfalls gleichmäßig über die Kunden sind.

(2) Relation FILME

In der Videothek sind aktuell 40'000 Filme gespeichert. Das Attribut *ProdJahr* enthält das Produktionsjahr des Filmes. In der Datenbank sind Filme mit Produktionsjahren von 1940 bis 1995 verzeichnet. 20% der Filme sind zwischen 1940 und 1970 produziert, 30% zwischen 1971 und 1985 und 50% zwischen 1986 und 1995. Alle Filme sind über 100 Genres und 10 verschiedene *Preiskategorien* gleichverteilt. Jeder Film besitzt eine *Plazierung* innerhalb der Hitliste. Man beachte aber, daß dieselben Hitparadenplätze mehrfach vorkommen können. So könnten z.B. zwei Filme dieselbe Plazierung besitzen, falls sich für sie dieselbe Anzahl von Ausleihen innerhalb eines Zeitraumes ergeben hat.

(3) Relation PERSONEN

Die Relation PERSONEN enthält eine Auswahl von *Personen* zu einem Film, die in diesem Film eine bestimmte *Funktion* bzw. Rolle haben. So hat z.B. die Person 'Wolfgang Petersen' in dem Film 'Das Boot' die Funktion 'Regisseur'. Personen kommen in der Relation normalverteilt vor. Die 30 verschiedenen Funktionen und alle Filme kommen gleichmäßig verteilt vor.

(4) Relation BAENDER

Die Anzahl der Bänder pro Film ist normalverteilt. Im Mittel sind jedem Film 10 Bänder zugeteilt; mit einer Standardabweichung von 3 Bändern. Die Bänder sind gleichmäßig über 300 verschiedene Standorte verteilt. Das Attribut *Ausleihen* enthält die Information, wie oft ein Band bereits ausgeliehen wurde. Zu einem Film wird jeweils das Band vergeben, das bisher am wenigsten ausgeliehen wurde.

(5) Relation AUSLEIHEN

Die Attribute *Von* und *Bis* enthalten konzeptionell nur Tagenummern zwischen 0 und unendlich, da die Videothek hier an keiner feineren Auflösung interessiert ist. Die Werte von *Von* sind gleichmäßig verteilt. Die Differenz zwischen *Bis* und *Von* (Leihdauer) ist normalverteilt mit einem Mittelwert von einem Tag und einer Standardabweichung von ebenfalls einem Tag. Die Relation enthält 100'000 Ausleihen und 5000 Reservierungen. Kunden kommen innerhalb von Ausleihen ungefähr normalverteilt vor. Alle Einträge mit einem *Bis*-Datum vor dem aktuellen Datum sind zurückgegeben (*Status*='z'), während alle Einträge mit einem *Bis*-Datum nach dem aktuellen Datum als noch ausgeliehen gelten (*Status*='a'). Im Mittel sind die Häfte aller Einträge mit einem *Bis*-Datum gleich dem aktuellen Datum schon zurückgegeben, die andere Hälfte ist ausgeliehen. Reservierungen haben ein *Von*-Datum, das gleich dem aktuellen Datum ist oder danach liegt. Reservierungen haben einen *Status*-Wert='r'.

Hinweise zu Anwendungsmodulen

Eine Analyse von Herrn Glotzkists Anwendung hat nun ergeben, daß sie aus 10 Modulen besteht. Jedes Modul besteht aus einer Menge von SQL-Queries, die innerhalb des Moduls eine gewisse Wahrscheinlichkeit besitzen, daß sie bei Aufruf des Moduls zur Ausführung kommen. Das Beiblatt *Anwendungsmodule und Queries* listet alle Module zusammen mit der Query-Nummer sowie der genauen SQL-Query auf. Innerhalb der SQL-Queries bezeichnen Namen, die mit einem Doppelpunkt beginnen Variablen, die zur Laufzeit der Anwendung festgelegt werden.
Die dritte Spalte enthält jeweils die Wahrscheinlichkeit, daß die Query beim Betreten des Moduls ausgeführt wird. Beispielsweise hat im Anwendungsmodul *1:Kundenidentifikation* die Query 1.1 eine Wahrscheinlichkeit von 0.8, während die Query 1.3 bei jedem Aufruf von Modul 1 zur Ausführung kommt.

Erwartete Last

Um nun einen möglichst optimalen physischen Datenbankentwurf machen zu können, müssen Sie zunächst Informationen über die erwartete Last an die Datenbank haben. Durch Beobachtung des Normalbetriebes über einen repräsentativen Zeitraum in der Videothek hat sich folgende Lastverteilung ergeben:

Query-Typ	Aufgerufenes Modul	Anteil an Gesamt-Tageslast
Online-Queries	1	1 Prozent
	2	11 Prozent
	3	24 Prozent
	4	4 Prozent
	5	24 Prozent
	6	5 Prozent
	7	11 Prozent
	8	20 Prozent
Batch-Queries	9	1 Aufruf pro Tag
	10	1 Aufruf pro Tag

Der erste Teil der Tabelle enthält Online-Modulaufrufe, die zufällig über den Beobachtungszeitraum verteilt sind. Die dritte Spalte im diesem Teil der Tabelle bezeichnet den Anteil eines entsprechenden Modules an allen Modulaufrufen des Beobachtungszeitraumes. Die Module 9 und 10 werden nur einmal im Batch-Betrieb aufgerufen. Online-Queries und Batch-Queries wechseln sich im Normalbetrieb immer ab (Tag- versus Nachtbetrieb).

AUFGABENSTELLUNG

Für die oben beschriebene Anwendung sollen Sie nun einen physischen Datenbankentwurf erstellen (Indizes, Index-Cluster, Hash-Cluster), der die folgenden Zielsetzungen hat:

a) **Die Anzahl der Blocktransfers für den spezifizierten Mix von Online-Queries soll minimiert werden, so daß sich minimale Antwortzeiten ergeben.**

b) **Die Anzahl der Blocktransfers für die Batch-Queries darf 50 000 Blocktransfers nicht überschreiten, damit die Tagesabrechnungen der Videothek in einem vorgegebenen Zeitraum abgewickelt werden können.**

c) **Die Datenbank darf durch Ihren Entwurf nicht größer als 200 MB werden. Entwürfe, die dieses Limit überschreiten, werden nicht zu Ende ausgewertet und gelten als ungültig.**

Um den optimalen Entwurf zu finden, haben Sie **3 Versuche**, die **spätestens bis zum 10.Februar** abgeschlossen sein müssen. Aus Kapazitätsgründen können wir Ihnen leider nicht die Möglichkeit geben,

Ihren Entwurf selbst zu evaluieren. Deshalb haben wir vorgesehen, daß Sie uns ein SQL-Skript per E-Mail an **kraiss@cs.uni-sb.de** schicken (**Subject: Entwurf zu P4**), das aus den 'CREATE CLUSTER...', 'CREATE TABLE...' und 'CREATE INDEX...' Statements Ihres Entwurfes besteht.

Wichtig: Verteilen Sie die Abgabe Ihrer drei Versuche möglichst gleichmäßig über die Zeit, denn sonst kann es passieren, daß wir bei Häufungen mit der Auswertung nicht nachkommen. Geben Sie Ihren ersten Versuch also möglichst kurzfristig ab, um schnell eine Basis für Ihre weiteren Entwürfe zu haben.

Den Aufbau der Befehle sowie die Semantik der Parameter, die Sie bestimmen sollen, entnehmen Sie bitte dem Beiblatt *SQL-Befehle für Schemaentwurf*. Wir werden dann die Testdatenbank basierend auf Ihrem Entwurf aufbauen, die Query-Ausführungs-Pläne für jede einzelne Query ermitteln sowie die Anzahl der akkumuliert übertragenen Blöcke für jede Query innerhalb des Last-Query-Mixes messen. Das Ergebnis schicken wir Ihnen dann automatisch zurück. Sie können nach der Analyse dieses Ergebnisses Ihren Entwurf dann nochmals modifizieren und uns erneut zur Auswertung zuschicken, soweit Sie noch einen weiteren von den drei Versuchen übrig haben.

Wichtig: Das Skript mit Ihrem Entwurf wird automatisch ausgeführt und bewertet. Bitte stellen Sie unbedingt sicher, daß das Skript auch erfolgreich ausgeführt werden kann (keine Syntaxfehler, keine ungültigen Parameter-Werte, keine Fehler in der Erzeugungsreihenfolge etc.). Als Feedback bekommen Sie die Ausführungspläne jeder einzelnen Query in der Auswertung mitgeschickt.

BELOHNUNG!!!

Der Entwurf mit dem besten Ergebnis wird am Semesterende prämiert. Ein Entwurf, der fehlerhaft ist oder das 'Zeitlimit' für die Batch-Queries überschreitet, ist ungültig und gilt als verbraucht (tja, in der Realität haben Sie als Datenbankadministratorin bzw. -administrator auch gegen begrenzte Zeitressourcen und Deadlines zu kämpfen).
In jedem Falle bekommen Sie jedoch die Auswertung Ihres Entwurfes zugeschickt.