

Datenbank-Praktikum

Wintersemester 2002/03

2. Teilaufgabe

Aufgabenstellung

In dieser Aufgabe soll ein Ausschnitt der in der 1. Teilaufgabe entworfenen Datenbank mit einem gegebenen Datenbestand geladen werden. Anschließend soll demonstriert werden, daß die Datenbank den bei der Informationsanalyse deutlich gewordenen Anforderungen genügt.

In einem ersten Schritt ist zu diesem Zwecke ein dem Entwurf entsprechendes Datenbankschema zu erzeugen und ein gegebener Rohdatenbestand in die Datenbank einzubringen. Dazu ist ausschließlich der Teil des Entwurfs, der sich mit den Staaten und Städten der Erde beschäftigt, zu realisieren und zu laden, alle Angaben zu Reisebüros bleiben unberücksichtigt.

Zweitens sind die Anfragen auf der Datenbank durchzuführen.

In einem dritten Schritt der Teilaufgabe sollen Änderungsoperationen auf der geladenen Datenbank durchgeführt werden. Dabei ist insbesondere auf die Wahrung der Konsistenz zu achten.

a) *Laden des Datenbankschemas / der Datenbank*

Schema laden: Erstellen eines Scripts zum Laden des Datenbankschemas.

Daten laden: Schreiben Sie als Ladeprogramm ein Java-Programm, welches Rohdatensätze aus XML- sowie zeichenseparierten-Dateien (CSV) einliest und im Falle syntaktischer und semantischer Korrektheit in Ihre Datenbank einspeichert. Die Konsistenz eines Satzes muß festgestellt werden, indem zum einen die Korrektheit des Satzes an sich untersucht wird und zum anderen seine Verträglichkeit mit den bereits

in der Datenbank enthaltenen Daten geprüft wird. Als inkonsistent erkannte Sätze sollen dabei zusammen **mit einer sinnvollen Begründung** der Ablehnung in eine zu erzeugende Datei ausgegeben werden.

Betrachten Sie die unten aufgeführten, potentiellen Ursachen für Inkonsistenzen und überlegen Sie sich, welche Konsistenzprüfungen beim Laden der DB durchgeführt werden müssen und wie/wann diese in die Datenbank eingebracht werden müssen.. Wählen Sie eine **sinnvolle Ladereihenfolge** der Datensätze (s. Anhang), um alle evtl. Abhängigkeiten zwischen zu ladenden und bereits in der Datenbank existierenden Sätzen berücksichtigen zu können.

Potentielle Ursachen für Inkonsistenzen sind:

- Syntaxfehler
- Duplikatsätze
- Datumsangaben müssen sinnvoll sein (bzgl. des aktuellen Datums, Altersangaben etc.)
- Datentypen müssen eingehalten werden bzw. bedürfen einer Konvertierung
- NULL-Werte erkennen/beachten
- Fehler **inhaltlicher** (auch Relationen-übergreifender) Art
-

Bei den Konsistenzprüfungen ist zu beachten, daß ggf. als fiktives 'heutiges' Datum der 1.11.2002 anzunehmen ist.

Die Daten der Flaggen und Karten stehen als JPG-Dateien bei *dbprak00* und sollen dort referenziert werden (nicht kopieren und keine BLOBs in der Datenbank dafür anlegen).

XML - Datei bearbeiten

Um Ihnen eine umfangreiche Einarbeitung in XML zu ersparen, wird beim Nutzer *dbprak00* im Verzeichnis *prak/Aufgaben/2.aufg/bsp_programme* ein Beispielprogramm in Java abgelegt (*XMLParserDemo.java*), welches Sie als Grundgerüst für Ihren Parser nutzen können. Desweiteren wird ihnen in *DB2Query.java* gezeigt, wie Sie mittels Java auf die Datenbank zugreifen können.

b) Anfragen auf der Datenbank

In diesem Teil sollen einige SQL-Anfragen auf der von Ihnen in Teil a) erzeugten Datenbank durchgeführt werden.

Führen Sie die Anfragen 1 bis 14 aus dem unter *prak/Aufgaben/2.aufg/fragen.txt* stehenden Fragekatalog aus und speichern Sie sowohl die Anfragen als auch die

Ergebnismengen in Dateien (s.u.).

c) **Änderungen in der erzeugten Datenbank**

In Ihrer Datenbank wurden (z.B. durch Schreibfehler) falsche Daten eingebracht. Werden die korrekten Daten mittels SQL-Anweisungen eingefügt, so sollen die durch eine solche Datenmanipulation in der Datenbank entstandenen Änderungen in temporären Tabellen automatisch protokolliert werden. Entwerfen Sie ein Format für die Protokolle.

Implementieren Sie dieses Verfahren für die Korrektur der Einwohnerzahl einer Stadt.

Erläuterungen und Hinweise zu den Teilaufgaben, befinden sich im Anhang.

Abgabe

Zum u.g. Zeitpunkt der Fertigstellung sollen in den Verzeichnissen ~/afg2/{a,b,c} Ihres Logins die nachfolgend beschriebene Dateien verfügbar sein. **Abzugeben** ist eine Liste mit den Namen dieser Dateien.

Für Teil a):

- Datei zur Erzeugung Ihres **Datenbankschemas**, inklusive aller dort enthaltenen Statements zur Integritätssicherung
- **Ladeprogramm**, *ausführlich kommentiert*
- Datei mit den **abgelehnten Datensätzen**, die für jede gefundene Inkonsistenz folgendes Format aufweisen soll:

ERROR:<Leerzeichen><Stadtname | Landname><Leerzeichen><Name des fehlerhaften Attributs><Leerzeichen><System- oder selbst formulierte Fehlermeldung><Leerzeile>

Zusätzlich soll diese Datei eine *Statistik* über die **Anzahl der Ablehnungen und Einspeicherungen pro Satztyp** (Typen: s. Anhang) enthalten (am Dateiende). Diese Fehlerdatei soll durch Ihr Lade-Programm *automatisch* erzeugt werden.

Für Teil b):

- **SQL-Anfragen** (Namen der Dateien: Q1.qry, Q2.qry, ...)
- **Ergebnisse** der Anfragen (Namen der Dateien: Q1.res, Q2.res, ...)

Für Teil c):

- Datei mit notwendigen **SQL-Statements** (für ein ausgewähltes Beispiel)
- Datei mit **Anweisungen zur Integritätssicherung**

Zum Zeitpunkt der Fertigstellung der Aufgabe soll die Datenbank den Status nach Durchführung der Teilaufgabe **2b** haben (geladene DB). Außerdem müssen Skripte existieren, die die in Aufgabe 2c geforderten Maßnahmen durchführen können (Funktionstest zum Testat).

Termine

Beginn der Bearbeitung:

ab sofort

Fertigstellung/Abgabe der Lösungen:

bis **9.12.2002**

Durchführung der Testate: n.V. (genaue Uhrzeiten pro Gruppe nach Absprache mit jew. Betreuer, Termine auch vor dem 9.12. möglich)

Anhang

(I) Allgemeines

- Jeder Gruppe des Datenbank-Praktikums ist ein eigenes Schema in der Datenbank prak0203 zugeordnet, das mit **dbprakXX** benannt ist, wobei XX die Gruppennummer darstellt. Beim Nutzer *dbprak00*, auf den Sie lesend zugreifen können, finden sie die zu ladenden Daten bzw. Musterdateien. Unter Ihrem dbprakXX-Login können Sie sich mittels der Anwendung **db2** und der Anweisung *connect to prak0203 user dbprakXX* mit der Datenbank verbinden. Ohne der Angabe eines Schemas arbeiten Sie automatisch im Kontext Ihres Schemas.
- Speicherplatzhinweise: Die Beachtung folgender Hinweise werden Ihnen Ihre Kommilitonen danken:
Löschen Sie unmittelbar alle nicht mehr benötigten Zwischenrelationen.
Beachten Sie, daß Sie in der from-Klausel einer SQL-Anweisung nur diejenigen Tabellen benennen, deren Attribute Sie auch (in select-Klausel oder join-Bedingungen) verwenden !! Ansonsten werden automatisch übermäßig große Zwischentabellen (kartesisches Produkt !) angelegt.
- Beim Auftreten **technischer** DB-Probleme (Server-Absturz, Einloggen nicht möglich etc.) schicken Sie bitte eine e-mail an **dbprak@informatik.uni-leipzig.de** oder nehmen Sie telefonischen Kontakt mit Herrn Jusek auf (Hausapp. 32 225).

(II) Erläuterungen und Hinweise zur Aufgabe:

- Das in Teil 1 des Praktikums erstellte **Datenbank-Schema** darf mit Ausnahme von
 - Attributtypen, die an die Datentypen der Ladedaten angepaßt werden müssen
 - nicht benötigten Attributennicht mehr geändert werden. Wenn Sie beim Betrachten der Rohdatenstruktur oder später beim Ausführen des Ladeprogramms feststellen, daß die von uns erstellten Daten trotz der Abstimmung im Testat zu Teilaufgabe 1 nicht mit Ihrem Verständnis der Informationsstrukturen übereinstimmen, so sollten Sie besser auf die Speicherung dieser Zusammenhänge verzichten. Das ist immer noch besser, als irgendwelche dubiosen "Dummy-Sätze" in der Datenbank abzulegen, deren Existenz am Schema nicht mehr zu erkennen ist und allein von den Anwendungsprogrammen kontrolliert werden muß.
In diesen Fällen ist aber ausführlich zu begründen, warum die Speicherung nicht wie vorgesehen möglich ist. Als Gründe sind bspw. eine unzureichende Beschreibung der Miniwelt oder zu viele eigene Ideen bei der Gestaltung des Schemas möglich.
Bei evtl. auftretenden **drastischen** Problemen kann eine Schema-Änderung durch-

geführt werden, allerdings erst **nach Rücksprache** mit dem Betreuer.

- Wollen Sie nur **alle Tupel einzelner Relationen** löschen, ohne deren Definition zu verlieren, so benutzen sie das SQL-Kommando 'delete * from <table>'.

(III) Benötigte Dateien

Alle benötigten Dateien liegen in */home/dbprak00/prak/*. In *Dokumentation.txt* sind Verweise auf Hilfstexte für Java, DB2 und XML enthalten. *Hinweise.txt* enthält eine Sammlung wichtiger Informationen zum Bearbeiten der Aufgaben. Diese Datei wird während des Praktikums nach Bedarf aktualisiert. Die für Aufgabe 2 benötigten Dateien liegen unter *2.aufg.* Dort gibt es ein Verzeichnis *rohdaten*, welches die Quelldaten für die Datenbank enthält. Diese Daten werden in der Datei *README* sowie in *beschreibung_laenderdaten.txt* beschrieben. Außerdem existiert ein Verzeichnis *bsp_programme*, welches je ein Java-Beispiel für eine Datenbankverbindung sowie zum Parsen einer XML-Datei enthält. Siehe auch hier die README-Datei.

(IV) Entwicklungsumgebung

Als Java-Entwicklungsumgebung empfehlen wir *Netbeans*, welches von der Kommandozeile mittels `netbeans` aufgerufen wird. Um den Speicherverbrauch dieser Anwendung zu minimieren, wurde sie so konfiguriert, daß sie nur die minimal notwendigen Module lädt. Dieses läßt sich jedoch zur Laufzeit ändern. Netbeans bringt eine umfangreiche Dokumentation mit, die vom Programm aus aufgerufen werden kann.

Als Alternative zu Netbeans, welches sehr speicherintensiv ist, können `nedit` oder `xemacs` als Editor empfohlen werden, da sie die Javasyntax durch Hervorhebungen unterstützen.

Hinweis: Sie können Netbeans auch auf ihrem eigenen Rechner verwenden. Es ist Open Source und erhältlich unter <http://www.netbeans.org>.

(V) Umgang mit dem Datenbanksystem DB2

Der Zugriff auf das Datenbanksystem erfolgt zum einen über die Befehlszeile des Terminals, zum anderen über die zu erstellenden Java-Programme mittels JDBC-Treibers.

- Zugriff über Befehlszeile

Durch Aufruf von *db2* wird der DB2-Klient gestartet. Bevor SQL-Operationen eingegeben werden können, muß die Verbindung zur Datenbank hergestellt werden. Dies erfolgt mittels

```
CONNECT TO prak0203 USER dbprak??
```

Dabei steht ?? für die Gruppennummer. "prak0203" ist der Datenbankname, d.h. alle Gruppen arbeiten auf der selben Datenbank, jedoch innerhalb eigener Schemas (der Schemaname entspricht dem Loginnamen). Danach können Operationen auf dem entsprechenden Schema ausgeführt werden.

Zum Beenden kann entweder *quit* oder *terminate* eingegeben werden. Bei *quit* wird zwar der Client beendet, der Hintergrundprozeß mit der Verbindung zur Datenbank bleibt jedoch bestehen (maximal bis zum Rechner-Logout). Dadurch muss nur beim ersten Aufruf von *db2* ein Datenbank-Connect durchgeführt werden, alle weiteren Aufrufe von *db2* sind dann sofort wieder mit der Datenbank verbunden.

Neben dem interaktiven Modus kennt *db2* auch einen *Batchmodus*. Hier kann mittels *db2 "SQL_ANFRAGE"*

eine Anfrage abgearbeitet werden (vorausgesetzt ein Connect wurde vorher durchgeführt).

Es können auch Folgen von Anweisungen in einer Textdatei abgelegt werden (getrennt durch Semikolon) und diese Datei an *db2* übergeben werden

```
db2 -t -f DATEI_NAME
```

Weitere Hinweise zum Programm *db2* können der DB2-Dokumentation (siehe Literaturverweise auf der Praktikums WWW-Seite) entnommen werden.

- Zugriff über Java-Programme

Java besitzt eine standardisierte Schnittstelle zum Zugriff auf relationale Datenbanken: JDBC. Die API dieser Schnittstelle kann in der Java-API (siehe *dokumentation.txt*) unter dem Paket *java.sql* nachgeschlagen werden. Eine praktische Anwendung, die den Datenbankzugriff mittels JDBC auf die Praktikumsdatenbank verdeutlichen soll, steht unter

```
/home/dbprak00/prak/Aufgaben/2.aufg/bsp_programme/DB2Query.java.
```

Der JDBC-Treiber wird zur Laufzeit über *Class.forName(...)* geladen. Damit die Java-Runtime diesen auch finden kann, muß er in der Umgebungsvariable *CLASSPATH* enthalten sein. Für die Praktikums-Accounts wird dieses automatisch erledigt. Soll

das Programm jedoch von einem anderen Rechner oder anderen Login gestartet werden, muß die Datei *db2java.zip* vorliegen (kann von */home/dbprak00/opt/lib/* kopiert werden) und in den CLASSPATH aufgenommen werden.

Wichtig: das Demo-Programm verwendet den Treiber *COM.ibm.db2.jdbc.app.DB2Driver*, welcher einen DB2-Klienten auf dem lokalen Rechner voraussetzt und ist damit praktisch nur auf den Praktikums-Accounts lauffähig. Um das Programm auch von anderen Rechnern aus zu benutzen, muß der Treibername zu *COM.ibm.db2.jdbc.net.DB2Driver* und die URL zu *jdbc:db2://doesen:6789/prak0203* geändert werden.

Weitere Hinweise werden ggf. bei dbprak00 gegeben und zwar im Praktikumsverzeichnis zu dem aktuellen Semester.