3. Versuch: Updates, Schemaänderungen, Referentielle Integrität

Aufgabe 3.1 (Views/Tabellen für Zwischenergebnisse, 10 P.)
Ein afrikanischer Häuptling veranstaltet ein Häuptlingstreffen in Khartoum. Dazu lädt er alle afrikanischen Häuptlinge ein, die Khartoum auf einem Weg mit zwei Grenzübertritten erreichen können, aber nicht aus einem Nachbarland sind (mit den Kollegen aus den Nachbarländern hat man nur Ärger). Es wird angenommen, dass auf je 100.000 Einwohner ein Häuptling kommt. Wieviele Personen kommen zu dem Treffen?

Zweiter Teil: Modifikation der obigen Annahme: In Ländern mit einer Bevölkerungsdichte von mehr als 60 Personen/Quadratkilometer ist nur jeder 200.000ste ein Häuptling.

Aufgabe 3.2 (Definition von Referentiellen Integritätsbedingungen, 20 P.)
Das Datenbankschema MONDIAL enthält in der einfachsten Version keine referentiellen Integritätsbedingungen. Erweitern sie die Schemadefinition um die entsprechenden Integritätsbedingungen einschließlich referentieller Aktionen.

Das Skript `create.sql`, mit dem die Erstellung der Datenbasis gesteuert wird, finden Sie in `/home/db-prakt/Mondial` auf *dbpraktikum*:

```
start mondial-drop-tables;
start mondial-schema;
start mondial-inputs;
```

Zum "Abbau" des Schemas dient das Skript `mondial-drop-tables` (entfernt alle MONDIAL-Tabellen). Das MONDIAL-Schema *ohne* Integritätsbedingungen wird von `mondial-schema` erstellt.

- Erzeugen Sie sich auf dieser Basis ein eigenes `XYcreate.sql`.
- Fügen Sie an einer geeigneten Stelle ein Skript hinzu, das entsprechende `ALTER TABLE`-Befehle enthält,
  oder
- Ersetzen Sie `mondial-schema` durch ein eigenes Skript `XYmondial-schema`, welches Sie aus `mondial-schema.sql` durch Veränderungen an den `CREATE TABLE`-Anweisungen entwickeln.

Berücksichtigen Sie die Tatsache, dass beim Einspielen der Tupel in der Datenbasis einige referentielle Integritätsbedingungen zeitweilig verletzt werden können.

- Wenn ein Land oder eine Provinz gelöscht wird, soll alles, was in diesem Land(esteil) liegt, auch gelöscht werden. Das gleiche gilt für alle Daten, die nur in Zusammenhang mit einem Land(esteil), einem Kontinent oder einer Stadt relevant sind.
- eine Stadt kann nicht gelöscht werden, wenn sie Hauptstadt von einem Land ist, das nicht gelöscht wird, oder wenn eine Organisation ihren Sitz dort hat.
- Eine Organisation kann nur gelöscht werden, wenn sie keine Mitglieder besitzt.
- Nachbarschaftsbeziehungen entfallen, wenn eines der beteiligten Gebiete gelöscht wird.
- Mit der Löschung eines Berges o.ä. wird auch seine geographische Lage überflüssig.
- Es darf keine kaskadierenden Löschungen von Informationen geben, die von der vom Benutzer geforderten Löschung unabhängig sind.

Erstellen Sie außerdem ein Skript `drop-constraints`, das diese referentiellen Integritätsbedingungen löscht.

Aufgabe 3.3 (View Updates, 20 P.)

a) Erzeugen Sie ein View, das für alle Länder einige der wichtigen Attribute aus *Politics*, *Economy* und *Population* sowie die Einwohnerdichte enthält. Formulieren Sie eine Anfrage, die zu jedem Land das Bruttosozialprodukt pro Person, die Kindersterblichkeit und das Bevölkerungswachstum geordnet nach BSP/Einwohner ausgibt. Was fällt auf?

Stellen Sie fest, auf welchen Spalten des Views Updates erlaubt sind, und begründen Sie dies. Aktualisieren Sie die Datenbasis so, dass sie die Bevölkerungszahl im nächsten Jahr angibt.

b) Erzeugen Sie ein zweites View, das zusätzlich zu dem aus a) zu jedem Land den Kontinent angibt, auf dem das Land hauptsächlich liegt.

Stellen Sie auch hier fest, auf welchen Spalten des Views Updates erlaubt sind und begründen Sie dies. Versuchen Sie, durch Veränderungen an dem View die Datenbank so zu ändern, dass Russland in Afrika liegt.

Aufgabe 3.4 (Trickreicher Gebrauch von referentiellen Aktionen, 20 P.)
In Aufgabe 2.10 wurde das Flusssystem des Zaire iterativ berechnet. Dazu müsste man vorher wissen, wieviele Iterationen benötigt werden. Lösen Sie die Aufgabe nun noch einmal anders, indem Sie eine geeignete Hilfsrelation sowie referentielle Aktionen nutzen, um daraus ein Ergebnis abzuleiten.

Zusatzfrage: Was müsste man tun, um alle Flüsse zu berechnen, deren Wasser in ein bestimmtes Meer fließt?

Aufgabe 3.5 (Europa-Update, 30 P.) In der Europäischen Union hat die Integration der Staaten so große Fortschritte gemacht, dass der Staatenbund zu einem Bundesstaat wird. Der neue Staat "Europe" besteht dabei aus den Vollmitgliedern der alten Europäischen Union. Einzig Großbritannien geht die Integration zu weit und es tritt aus der Europäischen Union aus. Auf der anderen Seite war das Referendum der Schweizer erfolgreich, weshalb sie in den neuen Bundesstaat aufgenommen werden.

Der Sitz der Europäischen Union wird dabei zur Hauptstadt, die Mitgliedsstaaten zu Provinzen. Dabei werden die Freistaaten Bayern und Sachsen ebenfalls zu Provinzen aufgewertet. Bei der Regierungsform orientierte man sich zu großen Teilen an der Bundesrepublik Deutschland mit seiner föderalen Struktur.

Bei Mittelungen gewichtet man wirtschaftliche Daten dabei sinnvollerweise mit dem Bruttosozialprodukt, für andere relative Größen bietet sich die Bevölkerungszahl als Gewichtungsfaktor an.

Bei der Mitgliedschaft in den Organisationen wurde demokratisch vorgegangen: War die Mehrheit der alten Einzelstaaten Mitglied einer Organisation (wobei die verschiedenen Mitgliedsarten nicht unterschieden werden), so wird der neue Staat Vollmitglied.

Abgabe bis 04.06.2004

4. Versuch: Zugriffsrechte

Die Bearbeitung der folgenden Aufgaben ist freiwillig!

Aufgabe 4.1 (Zugriffsrechte, 70 P.)
In dieser Aufgabe soll der Umgang mit Zugriffsrechten geübt werden. Diese Aufgabe soll von allen Teilnehmern einer Gruppe gemeinsam bearbeitet werden.

Löschen Sie zuerst Ihre MONDIAL-Datenbasis (Aufruf des Skripts `mondial-drop-tables`). In Aufgabe 3.2 haben Sie ein Skript geschrieben, das die MONDIAL-Datenbasis mit referentiellen Integritätsbedingungen erstellt. Teilen Sie dieses Skript nun so auf, dass es drei getrennte Datenbanken erstellt. In der ersten sollen alle politischen Daten gespeichert werden, in der zweiten alle rein geographischen, und in der dritten alle Tabellen, die beide Bereiche verbinden und ergänzende Daten (zu den Ländern) enthalten (Hinweis: verwenden Sie Synonyme).

- 1. Politik: Country, Province, City, Organization, borders, und Politics
  Person 1 entschließt sich, anstelle der Tabelle *Country* ein View zur Verfügung zu stellen, das die Spalten Name, Code, Population, Area, Capital, Province sowie eine Spalte Density (Einwohnerdichte) enthält.
- 2. Geographie: Sea, Lake, River, Mountain, Desert, Island, merges_with und Continent
- 3. Verbindungs- und Zusatzdaten: geo_..., located, encompasses, is_Member, Population, Economy, Religion, Language und Ethnic_Group.

Vergeben Sie die Zugriffsrechte so, dass die Betreiber der einzelnen Bereiche jeweils nur die Daten der anderen sehen/referenzieren können, die sie für eine sinnvolle Arbeit benötigen. Teilnehmer 1 darf die politischen Daten (Country, Province) in den Tabellen von Teilnehmer 3 verändern; Teilnehmer 2 darf die geographischen Daten (Namen) in den Tabellen von Teilnehmer 3 verändern.

Der 3. Teilnehmer repräsentiert zusätzlich a) eine übergeordnete Instanz, die auf semantische Konsistenz der DB achtet, und b) den klassischen Nutzer der Datenbank. Er darf alle Daten lesen, soll Kontrollanfragen stellen, und außerdem Objekte (Städte, Länder, Berge, ...), die er gerne in der Datenbank hätte, einfügen. Da er in einem solchen Fall wohl nicht alle Attribute kennt, ist das so gedacht, dass er etwa den Namen eines Berges ("Belchen") in die Relation *Mountain* einfügt, und die Betreiber der Relationen ggf. ihre Relationen (nach geeigneter Kommunikation) ergänzen.

Hinweis: In `/home/db-prakt/Mondial` auf *dbpraktikum* finden Sie Skripte, die zum Füllen der einzelnen Tabellen dienen (z. B. country.sql – Aufruf: `start country.sql`).

Aufgabe im einzelnen:

- Teilnehmer 1–3 erstellen ihren Anteil an der verteilten Datenbank, vergeben Zugriffsrechte und definieren Synonyme.
- Teilnehmer 3 überprüft die folgenden Bedingungen:
  - für alle Berge, die in mehreren Ländern liegen, sind diese Länder benachbart;
  - auf jedem Kontinent muss es mindestens ein Land geben;
  - jedes Land muss auf mindestens einem Kontinent liegen.
- Richten Sie es so ein, dass Teilnehmer 3 alle Aufgaben der Versuche 1 und 2 unverändert laufen lassen kann.
- In Aufgabe 3.5 haben Sie politische Veränderungen in Europa in MONDIAL integriert. Teilnehmer 1 und 3 führen diese Veränderungen an der aufgeteilten Datenbank durch.
- Frage: Gibt es eine einfachere Möglichkeit als diese, das Update auf der aufgeteilten Datenbank durchzuführen?

- überlegen Sie, welche Updates Teilnehmer 1 und 2 ausführen müssen, wenn Teilnehmer 3
  - einen Berg (`INSERT INTO MOUNTAIN (Name) VALUES ('Pico de Aneto')`) (liegt bei 0.6 ö. L., 42.6 n. Br. in den Pyrenäen an der Grenze zwischen Frankreich und Spanien) oder
  - eine Stadt (`INSERT INTO City (Name, Country, Province) VALUES ('Offenburg', 'D', 'Baden Wurttemberg')`) (8 ö. L., 48.5 n. Br., 70000 Ew)

  einfügt.

Aufgabe 4.2 (Auswirkungen von `GRANT REFERENCES`, 30 P.)
Untersuchen Sie mit Hilfe der Lösung zu Aufgabe 4.1 (ggf. mit geringfügigen Änderungen) die folgenden Szenarien:

1a. A erstellt eine Tabelle T_1 und erteilt B das Recht, diese zu referenzieren. B erstellt eine Tabelle T_2, die T_1 ohne Angabe einer referentiellen Aktion referenziert. A versucht, ein referenziertes Tupel in T_1 zu löschen.

1b. wie eben, mit einer `ON DELETE CASCADE`-Referenz.

1c. überlegen Sie sich, welche Folgen somit die Vergabe von `GRANT REFERENCES` und `ON DELETE CASCADE` hat und begründen Sie, warum dieses Vorgehen vernünftig ist.

2a. A besitzt eine Tabelle T, von der er B nur ein View V zur Verfügung stellt. B kennt T nicht, darf aber V lesen und schreiben.
Was passiert, wenn B ein Update auf V ausführt? Unterscheiden Sie Spalten, die direkt aus T übernommen sind und abgeleitete Spalten. Was passiert, wenn B in V etwas löscht?

2b. überlegen Sie sich, welche Folgen somit die Vergabe von Rechten auf Views für die zugrundeliegenden Basistabellen hat, und begründen Sie warum dieses Vorgehen auch vernünftig ist.