Klausur Datenbanken
Wintersemester 2013/2014
Prof. Dr. Wolfgang May
29. Januar 2014, 14-16 Uhr
Bearbeitungszeit: 90 Minuten

Vorname:

Nachname:

Matrikelnummer:

Studiengang:

Bei der Klausur sind **keine Hilfsmittel** (Skripten, Taschenrechner, etc.) erlaubt. Handies müssen ausgeschaltet sein. Papier wird gestellt. Benutzen Sie nur die **ausgeteilten**, zusammengehefteten **Blätter** für Ihre Antworten. Schreiben Sie mit blauem/schwarzem Kugelschreiber, Füller, etc.; Bleistift ist nicht erlaubt.

Zum **Bestehen** der Klausur sind **45** Punkte hinreichend.

☐ meine Note soll mit Matrikelnummer so bald wie möglich auf der Vorlesungs-Webseite veröffentlicht werden.

☐ meine Note soll nicht veröffentlicht werden; ich erfahre sie dann aus FlexNever oder beim zuständigen Prüfungsamt.

	Max. Punkte	Schätzung für "4"
Aufgabe 1 (ER-Modell)	16	12
Aufgabe 2 (Transformation in das Relationale Modell)	19	15
Aufgabe 3 (SQL und Relationale Algebra)	40	24
Aufgabe 4 (Verschiedenes)	15	4
Summe	90	55

Note:

Themenstellung: NSA-Telefongesprächsdatenbank

Alle Klausuraufgaben basieren auf einem gemeinsamen "Auftrag": In der Klausur soll eine Datenbank zur Speicherung von Daten über Mobiltelefongespräche (und Aufenthaltsorte) entworfen werden.

1. Handy-Telefonnummern bestehen aus einer Landesvorwahl und einer Nummer, z.B. 0049-12345 (Handynetzvorwahl und Endnummer werden nicht unterschieden; es werden keine Festnetztelefongespräche berücksichtigt.)
   Zu jedem Handy ist der Hersteller bekannt. (In diesem Szenario wird angenommen, dass man SIM-Karten nicht in ein anderes Handy umstecken kann; hier geht es nur darum, dass das Handy ein Attribut hat).

2. Jedes Land hat eine eigene Landesvorwahl:

   - 0049 ist *Deutschland*, 0093 ist *Afghanistan*.

3. Zu jedem Land ist die Region gespeichert. *Deutschland* liegt in *Mitteleuropa*, *Afghanistan* liegt im *Mittleren Osten*.

4. Von manchen Handynummern ist zumindest teilweise bekannt, von welchen Personen sie benutzt werden. Eine Handynummer kann von mehreren Personen benutzt werden (bzw. benutzt worden sein). Eine Person kann mehrere Handynummern benutzen.

5. Personen haben einen (Vor+Nach)Namen.

   - Die Handynummer 0049-10000 wird/wurde von *Angela Merkel*, *Ronald Pofalla* und *Peter Tauber* benutzt (und vielleicht noch von weiteren Personen). Es ist ein *Nokia*-Handy.
   - Die Handynummer 0049-12345 wird/wurde von *Angela Merkel* und *Joachim Sauer* benutzt. Es ist ein *Apple*-Handy.

6. Handies können geortet werden, wenn sie etwas senden oder empfangen. Für jede Ortung sind Tag, Uhrzeit, sowie die geographischen Koordinaten gespeichert.

7. Koordinaten werden als Längen- und Breitengrad (mit 2 Nachkommastellen) angegeben.

   - Das Handy 0049-10000 wurde am 20.12.2013 um 09:00 an den Koordinaten (13.37, 52.52) geortet.

8. Jedem solchen Koordinatenpaar ist das Land zugeordnet, in dem es sich befindet (Annahme: bei Grenzgebieten wird nur ein Land gespeichert; bei Koordinaten in internationalen Gewässern ist kein Land zugeordnet) sowie optional die Stadt (falls dort eine Stadt ist; ebenfalls maximal eine).

   - Das Paar (13.37, 52.52) befindet sich in *Berlin* in *Deutschland*.

9. Zu jedem abgehörten Gespräch ist der Zeitpunkt (Datum und Ortszeit des Anrufers), die anrufende Handynummer und die angerufene Handynummer, sowie eine Referenz auf eine Audiodatei gespeichert.

   - Am 27.12.2013 wurde von der Telefonnummer 0049-45454 um 12:00 (Ortszeit; das Handy befand sich an den Koordinaten (9.72, 52.16) in Hannover) die Nummer 0049-98765 angerufen; die Audio-Datei liegt unter `271213-4945454-1200.mp3`.

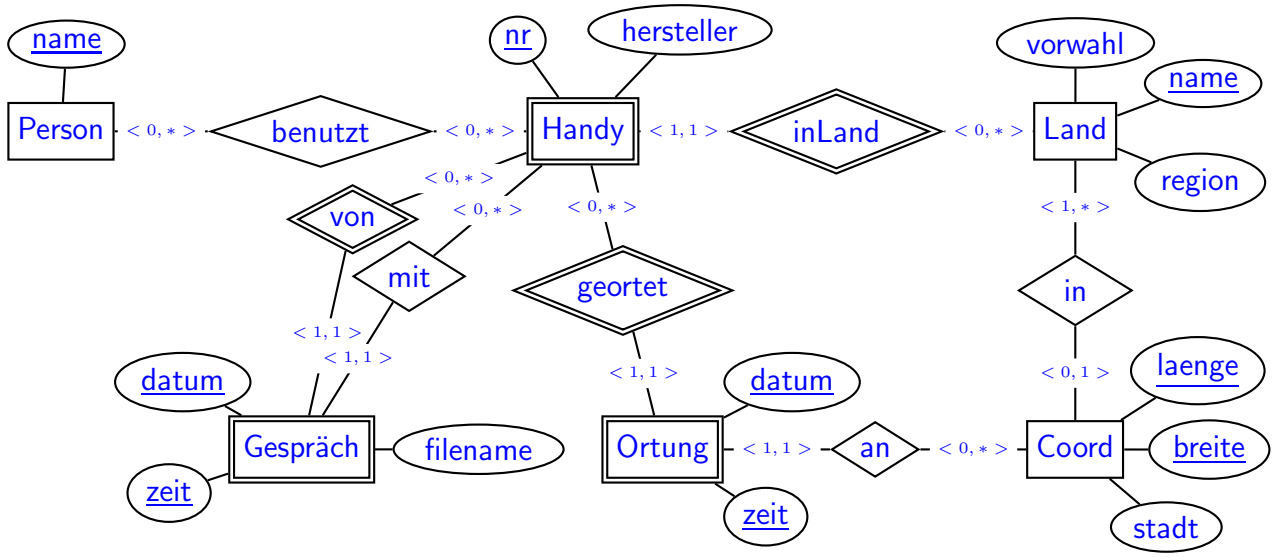
- Das dabei angerufene Handy 0049-98765 befand sich zu diesem Zeitpunkt an den Koordinaten (55.30, 25.27) in den *Vereinigten Arabischen Emiraten*, in der Stadt *Dubai*, wo es gerade 15:00 Ortszeit war.
- Das Handy mit der Nummer 0093-10101 (von dem bekannt ist, dass es von *Izmir Übül ben Müsli* benutzt wird) wurde am 27.12.2013 um 16:00 (Ortszeit) ebenfalls an den Koordinaten (55.30, 25.27) geortet, als von ihm die Handynummer 0093-121212 angerufen wurde; die Audio-Datei liegt unter `271213-9310101-1600.mp3`.
- Das dabei angerufene Handy mit der Nummer 0093-121212 wurde am 27.12.2013 um 21:00 (Ortszeit) an den Koordinaten (69.10,31.60) geortet.

  Dieses Koordinatenpaar befindet sich in *Afghanistan*; dort ist keine Stadt.

Hier ist auch gleich Platz für Aufgabe 1:

Aufgabe 1 (ER-Modell [16 Punkte])
Entwickeln Sie ein ER-Modell für das Szenario. Geben Sie darin die Schlüsselattribute sowie die Beziehungskardinalitäten an.

Lösung



Alternative Modellierungen und Kommentare:

- als Schlüssel für *Land* kann man *Vorwahl* oder *Name* nehmen, man muss sich bei der Umsetzung in das relationale Modell dann eine geeignete Abbildung überlegen.
- Alternativ auch *Gespräch* als zweistellige Beziehung zwischen zwei Handies mit Attributen *Datum*, *Zeit* und *Filename*.
- *Ortung* kann auch als Beziehung zwischen *Handy* und *Coord* mit Attributen *Datum* und *Zeit* modelliert werden.
- *Stadt* kann auch als (schwacher) Entitätstyp mit Beziehung *inLand* modelliert werden; muss aber nicht, da Städte keine weiteren Eigenschaften haben (und ihre Landeszugehörigkeit aus den Koordinaten hervorgeht).
- Bei *Gespräch* kann man theoretisch *filename* als Key nehmen. Dies bildet aber die Semantik der Anwendung schlecht ab, und wäre auch problematisch, wenn man igendwann die mp3-Dateien löschen will, und die Gesprächsdaten aber behalten möchte.

Aufgabe 2 (Transformation in das Relationale Modell [19 Punkte])

a) **Lösen Sie diesen Aufgabenteil auf dem *letzten* Blatt und trennen dieses ab (und geben es am Ende mit ab!).** Dann haben Sie dieses Blatt separat zugreifbar um später damit die Aufgaben 2b, 3 und 4 (SQL, Relationale Algebra+SQL, Diverses) zu lösen.

Geben Sie an, welche Tabellen (mit Attributen, Schlüsseln etc.) Ihre Datenbank enthält (keine SQL `CREATE TABLE`-Statements, sondern einfach grafisch). (12 P)

Markieren Sie dabei auch Schlüssel (durch unterstreichen) und Fremdschlüssel (durch überstreichen).

Geben Sie die Tabellen mit jeweils mindestens zwei Beispieltupeln (z.B. denen, die sich aus dem Aufgabentext ergeben, und weiteren erfundenen) an.

Lösung

Handy		
Vorwahl	Nr	Hersteller
0049	10000	Nokia
0049	12345	Apple

benutzt		
Nr	Vorwahl	Person
12345	0049	Angela Merkel
12345	0049	Ronald Pofalla
10101	0093	Izmir Übül ben Müsli

Land		
Name	Vorwahl	Region
Deutschland	0049	Mitteleuropa
Afghanistan	0093	Mittlerer Osten

Ortung					
vorwahl	nr	datum	zeit	Länge	Breite
0049	10000	20.12.2013	09:00	13.37	52.52
0049	98765	27.12.2013	15:00	55.30	25.27
0093	10101	27.12.2013	16:00	55.30	25.27

Coord			
Länge	Breite	Land	Stadt
13.37	52.52	Deutschland	Berlin
55.30	25.27	V.A.E.	Dubai

Gespräch						
vonVorw	vonNr	mitVorw	mitNr	Datum	Zeit	Filename
0049	45454	0049	98765	27.12.2013	12:00	271213-4945454-1200.mp3
0093	10101	0093	121212	27.12.2013	16:00	271213-9310101-1600.mp3

Tabellen: 1+1+1+2+2+2 P, Tupel jeweils 1/2P pro Tabelle.

- je nachdem, was man als Schlüssel für *Land* definiert, muss man die passenden Werte auch bei den entsprechenden Fremdschlüsseln verwenden!

b) Geben Sie die `CREATE TABLE`-Statements für diejenige Tabelle, in denen die Informationen über die Gesprächsdaten abgespeichert sind, so vollständig wie möglich an (verwenden Sie u.a. die Datentypen DATE und TIME; 7 P).

Lösung

```
CREATE TABLE Gespraech                          Basis 4P; je ca. 1/2 pro Attribut
( vonVorw  VARCHAR2(5),
  vonNr    VARCHAR2(20),
  mitVorw  VARCHAR2(5)  NOT NULL,       1/2P
  mitNr    VARCHAR2(20) NOT NULL,
  Datum    DATE,
  Zeit     TIME,
  filename VARCHAR2(30) UNIQUE,                                                  1/2P
  CONSTRAINT gkey (vonVorw, vonNr, datum, zeit),                                 1P
  CONSTRAINT fkvon FOREIGN KEY (vonVorw, vonNr) REFERENCES Handy(vorwahl, nr),   1/2P
  CONSTRAINT fkto FOREIGN KEY (mitVorw, mitNr) REFERENCES Handy(vorwahl, nr));   1/2P
```

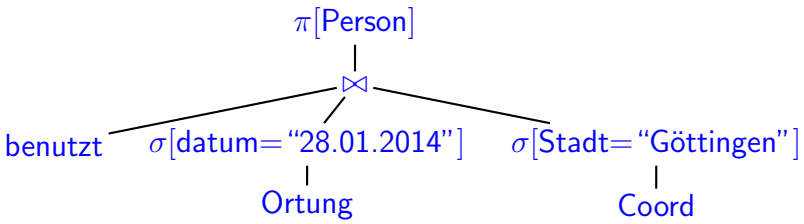
Aufgabe 3 (SQL und Relationale Algebra [40 Punkte])
Verwenden Sie für diese Aufgabe die von Ihnen entworfene relationale Datenbasis. Keine der Antworten soll Duplikate enthalten.

a) Geben Sie **eine SQL-Anfrage und einen Algebra-Ausdruck oder -Baum** an, die die Namen aller Personen ausgibt, die ein Handy benutzen, das am 28.1.2014 im Bereich der Stadt *Göttingen* geortet wurde. (3+3 P)

Lösung

```
SELECT DISTINCT Person
FROM Ortung, benutzt, coord
WHERE Ortung.vorwahl = benutzt.vorwahl
  AND Ortung.nummer = benutzt.nummer
  AND Ortung.laenge = coord.laenge
  AND Ortung.breite = coord.breite
  AND Ortung.datum = '28.01.2014'
  AND coord.Stadt = 'Goettingen'.

SELECT DISTINCT Person
FROM benutzt
WHERE (vorwahl,nummer) IN
  (SELECT vorwahl, nummer
   FROM Ortung, coord
   WHERE Ortung.laenge = coord.laenge
     AND Ortung.breite = coord.breite
     AND Ortung.datum = '28.01.2014'
     AND coord.Stadt = 'Goettingen').
```



b) Geben Sie eine **SQL-Anfrage** an, die alle deutschen Telefonnummern ausgibt, von denen mindestens 3 Anrufe durchgeführt wurden, bei denen sich der Angerufene zu diesem Zeitpunkt in *Nahost* oder im *Mittleren Osten* aufhielt. (4 P)

Lösung

```
SELECT Gespraech.vonNr
FROM Gespraech, Vorwahl, Ortung, Coord, Land
WHERE gespraech.vorwahl = '0049'    -- sauberer: join mit Land fuer "Deuts
  AND vorwahl.land = "Deutschland"
  AND gespraech.mitVorwahl = Ortung.vorwahl
```

```
  AND gespraech.mitNr = Ortung.nr
  AND gespraech.datum = Ortung.datum
  AND gespraech.zeit = Ortung.zeit
  AND Ortung.laenge = coord.laenge
  AND Ortung.breite = coord.breite
  AND coord.land = Land.name
  AND (Land.Region = 'Nahost' OR Land.Region = 'Mittlerer Osten')
GROUP by gespraech.vonNr
HAVING COUNT(*) > 2

SELECT vonNr
FROM Gespraech
WHERE gespraech.vorwahl = '0049'    -- sauberer: join mit Land fuer "Deutschland"
  AND (mitVorwahl, mitNr, datum, zeit) IN
      (SELECT vorwahl, nummer, datum, zeit
       FROM Ortung
       WHERE (laenge, breite) IN
             (SELECT laenge, breite
              FROM coord
              WHERE land IN
                    (SELECT name
                     FROM land
                     WHERE region IN ('Nahost', 'Mittlerer Osten'))))
GROUP by vonNr
HAVING COUNT(*) > 2
```

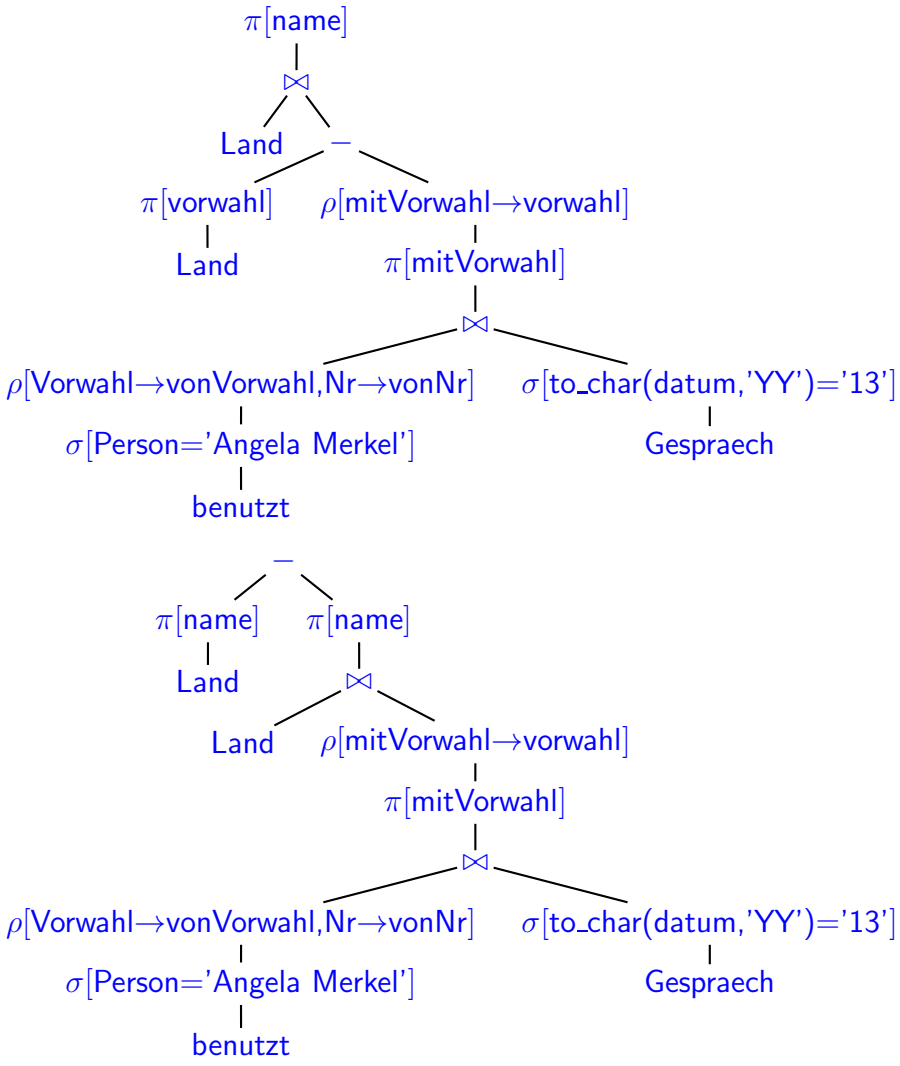
c) Geben eine **eine SQL-Anfrage und einen Algebra-Ausdruck oder -Baum** an, die die Namen aller Länder ausgibt, zu deren Landesvorwahlbereich im Jahr 2013 keine Gespräche von Telefonen (als Anrufende), die von *Angela Merkel* benutzt werden, abgehört worden sind. (4+5 P)

Lösung

```
SELECT name
FROM Land
WHERE NOT EXISTS
  (SELECT
   FROM Gespraech, benutzt
   WHERE gespraech.vonNr = benutzt.Nr
     AND gespraech.vonVorwahl = benutzt.Vorwahl
     AND gespraech.datum >= '01.01.2013'
     AND gespraech.datum < '01.01.2014'
     AND benutzt.Person = 'Angela Merkel'
     AND gespraech.mitVorwahl = Land.vorwahl)

SELECT name
FROM Land
WHERE vorwahl NOT IN
```

```
(SELECT mitVorwahl
 FROM Gespraech, benutzt
 WHERE gespraech.vonNr = benutzt.Nr
   AND gespraech.vonVorwahl = benutzt.Vorwahl
   AND gespraech.datum >= '01.01.2013'
   AND gespraech.datum < '01.01.2014'
   AND benutzt.Person = 'Angela Merkel')
```

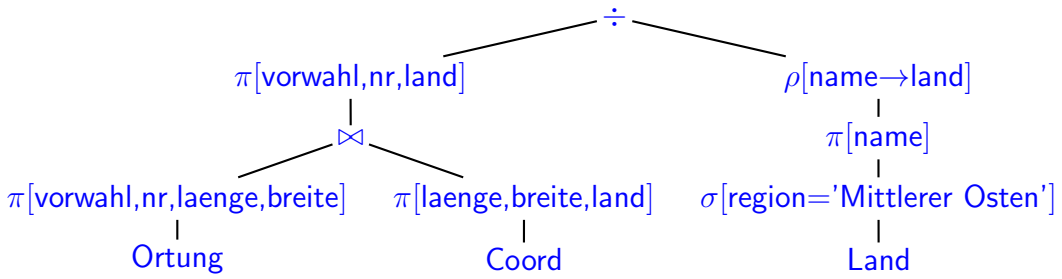


d) Geben Sie eine **eine SQL-Anfrage und einen Algebra-Ausdruck oder -Baum** an, der alle diejenigen Handynummern (Vorwahl und Nummer) ausgibt, in jedem Land des mittleren Ostens mindestens einmal geortet wurden. (5+5 P)

Lösung

```
SELECT vorwahl, nummer
FROM Handy
WHERE NOT EXISTS
  (SELECT *
   FROM Land
```

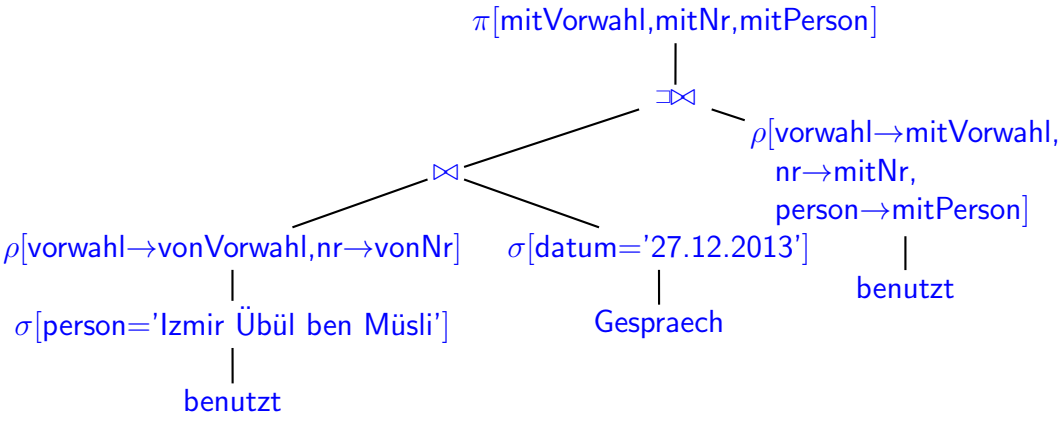
```
WHERE Region = 'Mittlerer Osten'
 AND NOT EXISTS
     (SELECT *
      FROM Ortung, Coord
      WHERE Ortung.Laenge = Coord.Laenge
        AND Ortung.Breite = Coord.Breite
        AND Coord.Land = Land.Name
        AND Ortung.vorwahl = Handy.vorwahl
        AND Ortung.nummer = Handy.nummer ))
```



e) Geben Sie **eine SQL-Anfrage und einen Algebra-Ausdruck oder -Baum** an, der die Nummern aller Handies, und wenn möglich auch die Namen ihrer Benutzer, ausgibt, die von einem Handy, das *Izmir Übül ben Müsli* benutzt, am 27.12.2013 angerufen wurden. (4+4 P)

Lösung

```
SELECT gespraech.mitVorwahl, gespraech.mitNr, b2.Person
FROM benutzt b1, gespraech LEFT OUTER JOIN benutzt b2
ON (gespraech.mitVorw = b2.vorwahl
    AND  gespraech.mitNr = b2.Nr)
WHERE b1.vorwahl = gespraech.vonVorwahl
  AND b1.nummer = gespraech.vonNr
  AND gespraech.datum = '27.12.2013'
  AND b1.person = 'Izmir "Ub"ul ben M"usli'
```



Hinweis: man muss darauf achten, dass die *Person*-Spalten der zweimal verwendeten *benutzt*-Relation durch Umbenennung (oder rechtzeitiges wegprojizieren links) auseinandergehalten werden.

f) **Etwas Theorie:** Gegeben sind zwei beliebige Relationen $R(\bar{X})$ und $S(\bar{Y})$ mit $\bar{Y} \subseteq \bar{X}$. Wenn S nur ein Tupel enthält, wie kann der Ausdruck $R \div S$ vereinfacht werden? (3 P)

Lösung Alle Projektionen auf $\bar{X} \setminus \bar{Y}$ von R-Tupeln, deren Y-Komponente gleich dem Tupel in S ist:

$\pi[\bar{X} \setminus \bar{Y}](R \ltimes S)$ oder $\pi[\bar{X} \setminus \bar{Y}](R \bowtie S)$.

Ebenfalls richtig, aber umständlich formuliert, ist $\pi[\bar{X} \setminus \bar{Y}](\sigma[\bar{Y} = s](R))$ (wobei s das Tupel in S ist) oder
$\{\mu \in \mathsf{Tup}(\bar{X} \setminus \bar{Y}) : \{\mu\} \times \{s\} \in R\}$.

Aufgabe 4 (Verschiedenes [15 Punkte])

a) Die automatische Erfassung von Gesprächen stellt am 29.1.2014 um 14:15 ein Gespräch von 0049-12345 (das bekannte Merkelsche Partei-Handy) mit 0041-99999 (ein *Huawei*-Handy, über das bisher nichts gespeichert ist) fest und legt den Mitschnitt in der Datei `290114-4912345-1415.mp3` ab.
Leider funktioniert die geographische Ortung gerade nicht, so dass keine entsprechenden Angaben zur Verfügung stehen.

Geben Sie an, welche(s) `INSERT`-Statement(s) auf Ihrer Datenbank ausgeführt werden (3 P).

Lösung

```
INSERT INTO Handy VALUES('0041', '99999', 'Huawei');
INSERT INTO Gespraech VALUES('0049','12345','0041','99999',
                             '29.01.2014','14:15');
```

Anmerkung zur Modellierung: Wenn z.B. durch eine fehlerhafte Nummernerfassung beim Abhören eine nicht-existierende Vorwahl, z.B. 0083 (die keinem Land zugeordnet ist) zustandekommt, würde die Datenbank in der obigen Modellierung dies nicht feststellen! *Vorwahl* ist nirgends Schlüssel, also in *Handy* auch kein Fremdschlüssel.
In *Land* kann *Vorwahl* nicht Schlüssel sein, da dort schon *Name* Schlüssel ist (und ja auch in *Coord* als Fremdschlüssel benötigt wird).
Eine Möglichkeit, die Existenz der erfassten Vorwahlnummern zu garantieren, wäre, eine weitere Tabelle *Vorwahl(VorwNr, Land)* zu definieren. Dies würde aber zu Redundanz führen, ist also auch keine optimale Möglchkeit.
In diesem Fall wäre es sinnvoll, einen Trigger auf INSERT auf die Tabelle *Handy* zu definieren, der die Existenz der Vorwahl überprüft, und sonst die einfügende Transaktion scheitern läßt.

b) Nehmen Sie an, das Land *Tahiti* ist in den Ergebnissen zu Aufgabe 3c) **nicht** enthalten. Kann man daraus schliessen, dass *Angela Merkel* in 2013 nie ein Handy, dessen Landesvorwahl diejenige von *Tahiti* ist, angerufen hat?
Begründen Sie Ihre Aussage (3 P).

Lösung Nein. Sicher ist das nicht.

- es kann sein, dass nicht von allen Telefonen, die Angela Merkel benutzt, bekannt ist, dass sie dies tut. Der Inhalt der *benutzt*-Relation ist sicher ziemlich unvollständig (das muss ja größtenteils über Stimmerkennung gemacht werden).
- das Parteihandy wurde abgehört, das Kanzler(Innen)handy nicht.
- auch der NSA kann mal ein Gespräch entgangen sein.

Dieses "Nicht-sichere Schließen" wird als "Open World" bezeichnet. Wenn etwas in der DB nicht gespeichert ist, wird angenommen, dass es nicht gilt. Ob ein solcher Schluss sicher zutrifft, kommt auf die Anwendung an.

c) Interne Auswertung/Indexe: Betrachten Sie Ihre SQL-Anfrage zu Aufgabe 3a).
Nehmen Sie folgende interne Speicherung an:

– Es sind Baumindexe auf allen Schlüsseln und allen Fremdschlüsseln vorhanden.
– Die Inhalte aller Tabellen, die Spalten *Datum* und *Zeit* enthalten, sind nach (*Datum, Zeit*) geordnet abgelegt.
– Für eine gegebene Stadt erhalten Sie in $O(\log n)$ über einen Index das Tupel mit den Koordinaten.

Beschreiben Sie *kurz*, wie Ihre Anfrage zu Aufgabe 3a) einigermaßen effizient ausgewertet werden kann. (5 P)

Lösung

1) Zugriff mit *Stadt="Göttingen"* auf *Coord*, ergibt die Koordinaten in $O(\log n)$.

2a) 2a.1) Zugriff auf alle *Ortungen* (geordnet und damit geblockt gespeichert) am 28.1.2014; diese werden sequentiell durchlaufen und auf die Koordinaten von Göttingen überprüft.

2a.2) Die gesuchten *(Vorwahl/Nummer)*-Paare effizient in eine Menge sammeln (z.B. TreeSet).

2b) Alternativ:

2b.1) Page-Adressen der (geordnet und damit geblockt gespeicherten) Ortungen am 28.1.2014 herausfinden

2b.2) Im Index auf *Ortung(Länge,Breite)* die Referenzen auf Tupel für die Göttinger Koordinaten herausfinden, und gleich nach den in 2b.1 ausgewählten Seiten filtern.

2b.3) Auf diese Seiten/Tupel zugreifen (nicht jede Seite für den 28.1.2014 wird ein Tupel für Göttingen enthalten, damit ist 2b) noch etwas effizienter als 2a)) und die gesuchten *(Vorwahl,Nummer)*-Paare effizient in eine Menge sammeln (z.B. TreeSet).

3) Über den Index *benutzt(Vorwahl,Nummer)* die Namen holen (und via Mengendatenstruktur die Duplikate eliminieren).

d) Geben Sie für die beiden typischen potentiellen Transaktionsfehlersituationen *Lost Update* und *Dirty Read* an, ob sie beim alltäglichen Betrieb der obigen Datenbank prinzipiell auftreten könnten, wenn man kein Transaktionsmanagement benutzen würde. Falls ja, skizzieren Sie eine solche Situation; falls nein, begründen Sie Ihre Aussage (4 P)

Lösung

– Lost Update: kann nicht auftreten; in dieser Anwendung werden keine Werte gelesen, bearbeitet, und dann wieder geschrieben.
– Dirty Read: Zwei mögliche sinnvolle Antworten:
  * Es gibt es keine Transaktionen, die aus mehreren Teilaktionen bestehen, und scheitern können.
  * Eine komplexere Einfügung wie in 4a) könnte teilweise gelingen, und dann z.B. wegen falsch erkannter Daten (ungültige Vorwahl!) scheitern.