Aufgabenblatt 1: SQL, ETL, Graphvisualisierung

Klemens Böhm, Matthias Bracht und Frank Eichinger

Praktikum Data Warehousing und Mining, Sommersemester 2010
Institut für Programmstrukturen und Datenorganisation (IPD)
Karlsruher Institut für Technologie (KIT)

1 SQL-Anfragen

Lösen Sie alle folgenden Aufgaben in Zweierteams. Der im Folgenden verwendete <Teamname> ergibt sich aus „dwmteam“, der Nummer Ihres Tutoriums sowie a, b, oder c für die jeweiligen Teams (also „dwmteam1a“ bis „dwmteam4c“). Teilen Sie Ihrem Tutor zeitnah mit, in welchen Teams Sie arbeiten.

Für die folgenden Anfragen können Sie den Oracle SQL Developer nutzen. Verwenden Sie die unter A. angegebene Datenbank mit dem Benutzer „lpt_lek_logo“ sowie dem gleichnamigen Schema. Speichern Sie bitte Ihre Anfragen und führen Sie sie Ihrem Tutor bei der Abgabe vor.

a) Erstellen Sie eine Liste aller Postleitzahlen sowie tatsächlicher Ankunfts-/Abfahrtszeiten für alle Tourhalte von Aktualtouren (C_ISACTUAL = 1), aufsteigend sortiert nach Tour-ID und Reihenfolge der Halte.

Beispieltupel:	*73f3...*	*0*	*90443*	*29.03.10 00:00:00*	*29.03.10 00:00:00*
	73f3...	*1*	*92422*	*29.03.10 01:53:15*	*29.03.10 02:43:15*
	…				

b) Geben Sie die Anzahl der Halte für Aktualtouren mit weniger als zehn Stopps aus, absteigend sortiert nach Anzahl der Halte.

Beispieltupel:	*73f3…0002*	*9*
	73f3…00d	*9*
	…	

c) Geben Sie an, welche Postleitzahl an welchem Wochentag in Aktualtouren wie oft angefahren wurde, und sortieren Sie das Ergebnis absteigend nach Häufigkeit!

Beispieltupel:	*70806*	*FREITAG*	*36*
	70806	*MITTWOCH*	*30*
	70806	*DONNERSTAG*	*30*
	74354	*DONNERSTAG*	*30*
	…		

d) Ist es möglich, das Ergebnis der vorherigen Teilaufgabe so zu modifizieren, dass für jede Postleitzahl genau das Tupel mit demjenigen Wochentag ausgegeben wird, an dem diese Postleitzahl am häufigsten angefahren wurde? (Von den vorherigen Beispieltupeln wären nur noch Tupel 1 und 4 enthalten.) Wenn ja, wie sieht die entsprechende Anfrage aus? Wenn nein, warum nicht?

2 ETL-Prozess

Die bisher genutzten Daten sollen nun aus der Oracle-Datenbank in die PostgreSQL-Datenbank übertragen werden, unter Verwendung des in der Vorlesung vorgestellten generischen Graph-Schemas. Dies soll mit einem wiederverwendbaren ETL-Prozess geschehen, der jederzeit manuell angestoßen werden kann.

Dem Tutor ist das erstellte Schema vorzuführen sowie der ETL-Prozess zu zeigen. Zusätzlich sind der ETL-Prozess sowie eine Beschreibung der verwendeten Attribute analog zu der im Wiki via E-Mail an Matthias Bracht und Frank Eichinger einzureichen.

Jedes Team hat andere Graphen zu erstellen. Folgende Parameter spielen dabei eine Rolle:

- Aufteilen in Einzelgraphen:
  - Nein: Graphen müssen nicht zusammenhängend sein.
  - Ja: Jede Zusammenhangskomponente bildet einen separaten Graph.
- Zusammenfassen von Mehrfachkanten:
  - Nein: Es darf mehrere Kanten geben, die die gleichen zwei Knoten verbinden.
  - Ja: Mehrere Kanten zwischen den gleichen zwei Knoten werden zu einer einzigen Kante zusammengefasst.
- Zeiteinheit:
  - Pro Tag: Alle Fahrten an einem Kalendertag bilden einen Graphen.
  - Pro Woche: Alle Fahrten einer Kalenderwoche bilden einen Graphen.
  - Pro Wochentag: Alle Fahrten am gleichen Wochentag bilden einen Graphen.

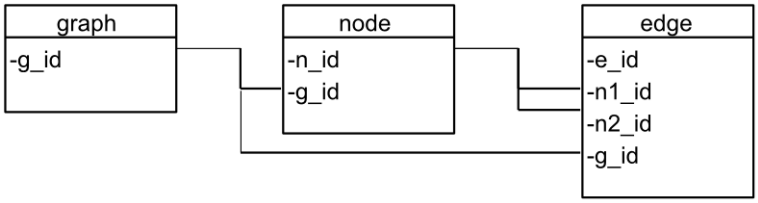
Die Aufgaben für jedes Team sind in der folgenden Tabelle zusammengefasst.

Team	Aufteilen in Einzelgraphen	Zusammenfassen von Mehrfachkanten	Zeiteinheit
dwmteam1a	Ja	Ja	Tag
dwmteam1b	Ja	Ja	Woche
dwmteam1c	Ja	Ja	Wochentag
dwmteam2a	Nein	Nein	Tag
dwmteam2b	Nein	Nein	Woche
dwmteam2c	Nein	Nein	Wochentag
dwmteam3a	Nein	Ja	Tag
dwmteam3b	Nein	Ja	Woche
dwmteam3c	Nein	Ja	Wochentag
dwmteam4a	Ja	Nein	Tag
dwmteam4b	Ja	Nein	Woche
dwmteam4c	Ja	Nein	Wochentag

Die für die Aufgabe relevanten Tabellen sind (jeweils im Schema „lpt_lek_logo“):

- T_TOURMGMT_TOUR (gefahrene LKW-Touren)
- T_TOURMGMT_TOURPOINT (einzelne Halte bei einer Tour)
- T_ORGLOC_LOCATION (Standorte)
- T_ORGLOC_LOCATIONFUNCTION (nur interessant wegen C_TYPETAG)
- T_ADDRESSMGMT_PADDRESS (Adressen)

Primär- und Fremdschlüssel im Zielschema sollen folgendermaßen aussehen:



Für Graphen, Knoten und Kanten sind neue, jeweils eindeutige numerische IDs zu vergeben, insbesondere für jede unterschiedliche Adresse (PLZ, Straße) in einem Graphen eine neue Knoten-ID. Zusätzlich sind passende Attribute für die Relationen graph, node und edge in den Ursprungsdaten zu identifizieren. Zu verwenden sind nach gesundem Menschenverstand hinreichend viele Attribute, die in der Oracle-Datenbank „sinnvoll“ befüllt sind, außerdem die ursprünglichen Tour-IDs. Die Bedeutung einzelner Attribute in der Oracle-Datenbank ist im IPD-Wiki aufgeführt: http://www.ipd.kit.edu/~ipd/wiki/mediawiki-1.5.6/index.php/DWM-Praktikum.

Je nach Aggregationsgrad der Graphen (siehe oben) sind andere Graph-/Knoten- bzw. Kantenattribute sinnvoll. Bei der Zusammenfassung von Mehrfachkanten könnte

die resultierende Kante beispielsweise mit Minimal-/Maximal-/Durchschnittswerten der Attribute der Einzelkanten annotiert werden.

Hinweise: Es sind nur Aktualtouren zu verwenden (C_ISACTUAL = 1). Mehrere Lieferungen zur gleichen Adresse sind in der Oracle-Datenbank als mehrere Stopps nacheinander an der gleichen Adresse erfasst. Diese sollen zu einem einzigen Stopp zusammengefasst werden, sodass es insbesondere keine reflexiven Kanten gibt.

Für die Bewertung der Lösung spielen die ausgewählten Graph-/Knoten-/Kantenattribute sowie deren Dokumentation eine Rolle. Achten Sie in der Zieldatenbank auch auf korrekte Primär- und Fremdschlüsselbeziehungen sowie Datentypen.

3 Graphvisualisierung

Ein „interessanter“ Graph soll visualisiert werden. Dazu können die Tools der Graphviz-Sammlung verwendet werden (www.graphviz.org). Abzugeben ist eine Grafikdatei. Je nach Aufgabenstellung oben haben die Teams unterschiedliche Kantenfärbungen vorzunehmen:

Tutorium	Kantenfärbung
1	keine
2	LKW-Ladung: rot/gelb/grün für viel/mittel/wenig
3	Durchschnittliche LKW-Ladung: rot/gelb/grün für viel/mittel/wenig
4	Verspätete Fahrten: rot/schwarz für ja/nein

A. Software

Zur Lösung der Aufgaben bietet sich die unten aufgeführte Software an. Für Aufgabe 2 empfehlen wir die Verwendung des schon bekannten SPSS Modelers, welcher auch für Datenbankzugriffe genutzt werden kann. Er greift mittels ODBC auf Datenquellen zu. Dazu ist zusätzlich zum SPSS Modeler 13 folgende Software zu installieren, die auf den Poolrechnern schon bereitsteht:

- Für Einblicke in die Oracle-Datenbank: Oracle SQL Developer (http://www.oracle.com/technology/products/database/sql_developer/index.html)

- Für Oracle-ODBC-Zugriff: SPSS Inc. Data Access Pack mit dem Oracle Wire Protocol (http://www.spss.com/drivers/clientCLEM.htm)

- Für Einblicke in die PostgreSQL-Datenbank: pgAdmin (http://www.pgadmin.org/download/)

- Für PostgreSQL-ODBC-Zugriff: psqlODBC (http://www.postgresql.org/ftp/odbc/versions/)

Nach der Installation der ODBC-Treiber sind die zugehörigen Quellen wie folgt einzurichten (hier das Vorgehen für Windows XP):

- Oracle:
  - Systemsteuerung, Verwaltung, Datenquellen (ODBC), Tab "User DSN",
  - Hinzufügen, „SPSS Inc OEM 5.3 Oracle Wire Protocol“
  - Data Source Name: "Oracle logotakt"
  - Host: i40db01.ipd.uka.de
  - Port Number: 1521
  - SID: „logotakt“

- PostgreSQL:
  - Systemsteuerung, Verwaltung, Datenquellen (ODBC), Tab "User DSN",
  - Hinzufügen, „PostgreSQL Unicode“
  - Data Source Name: "PostgreSQL logotakt"
  - Database: <Teamname>
  - Server: magdeburg.ipd.uka.de

Anschließend sollte es möglich sein, im SPSS Modeler eine Oracle-Datenbankquelle (User „lpt_lek_logo“) und eine PostgreSQL-Datenbanksenke (User: <Teamname>) einzurichten und eine der oben genannten relevanten Relationen testweise von Oracle nach PostgreSQL zu überspielen (sie kann mit pgAdmin wieder entfernt werden). Die zugehörigen Passwörter erhalten Sie bei Ihrem Tutor.

B. Oracle-Datenbank: Schema

Im folgenden Diagramm sind die Fremdschlüsselbeziehungen zwischen den oben angegebenen, für die Aufgabenstellung relevanten Relationen dargestellt.