Big Data und Business Intelligence

Aufgabenblatt 01 (Datenintegration, DWH)

Prof. Dr.-Ing. Heiko Tapken

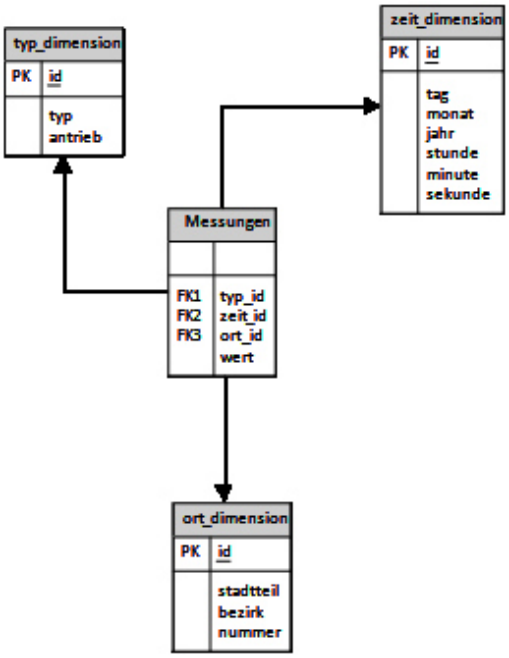
Wintersemester 2016/17

Testat: s. Planungen

Hochschule Osnabrück
University of Applied Sciences

Die Verkehrsleitzentrale (VLZ) von Osnabrück verfügt über einen enormen Datenbestand. Dieser befindet sich leider in verschiedenen Datenquellen und kann so nicht für ein Data Warehouse verwendet werden.

Ziel dieser Aufgabe ist es, die Daten in ein Star-Schema zu überführen, welches wie folgt aussieht:



Wie man sehen kann, stellt dieses Starschema einen Data Cube mit 3 Dimension (Ort, Zeit und Typ) dar. Die Faktentabelle „Messungen“ enthält die eigentlichen Messwerte, die jeweils einen Messzeitpunkt, eine Messstelle als Ort und einen Messtypen haben.

Um nun dieses Schema für einen Data Cube benutzen zu können, müssen die fehlenden Werte integriert werden.

Datenbank

Sie verfügen pro Gruppe über den bekannten Zugang zum Oracle Datenbankserver. Für eine Verbindung zu Ihrer Datenbank sind folgende Informationen notwendig:

Datenbanktyp:	Oracle
Server:	oracle-srv.edvsz.hs-osnabrueck.de
Port:	1521
Benutzername:	bd<gruppennummer>
Passwort:	bd<gruppennummer>

Wobei Sie <gruppennummer> durch Ihre Gruppennummer austauschen müssen, in der Sie sich im OSCA eingetragen haben (bzw. vorher sollten, damit es keine Kollisionen zwischen verschiedenen Gruppen gibt).

Sie haben mit diesen Zugangsdaten Zugriff auf drei Datenbanken. Zum einen lesenden Zugriff auf die Datenbanken fgdb_`verkehrsdaten` und fgdb_osnabrueck. Zum anderen einen lesenden und schreibenden Zugriff auf die Datenbank `g<NR>`, welche nur für Ihre Gruppe gedacht ist.

Die Datenbank `verkehrsdaten` hat eine Tabelle `Messung` mit folgendem Schema:

```
Messwerte(messstellennr,typ,wert,datum uhrzeit)
```

Es gilt dabei folgende Semantik:

messstellennr: Die Nummer der Messstelle mit der gemessen wurde.

typ: Gibt an, ob ein *LKW*, ein *PKW*, ein *MOTORRAD*, ein *FUSSGAENGER* oder ein *FAHRRAD* gemessen wurde. Entsprechend hat jede Messstation zu einem Zeitpunkt immer je einen Messwert pro Typ.

wert: Ist der Messwert, also die Anzahl der von einem Typ zu dem angegebenen Zeitpunkt durch die Messstation gezählten Einheiten.

tag, monat, jahr,…: Das Datum bzw. die sekundengenaue Uhrzeit der Messung.

Die Daten für die Stadtteilbezirke finden Sie in der Datenbank fgdb_osnabrueck. Diese wurde wie folgt erzeugt:

```
CREATE TABLE Stadtteil (
  Nr Number(5),
   Name varchar2(255),
   PRIMARY KEY (Nr)
);

CREATE TABLE StatistischerBezirk (
   Bezirksnummer Number(4),
   Stadtteil Number(5),
   PRIMARY KEY (Bezirksnummer),
   foreign key (stadtteil) references stadtteil (Nr)
);

create table messstelle
(
  MessstellenNr Number(4),
  BezirksNr Number (4),
  primary key (MessstellenNr, BezirksNr),
  foreign key (BezirksNr) references StatistischerBezirk(Bezirksnummer)
)
```

Die Dimension typ-Dimension wird wie folgt spezifiziert:

```
Id, typ, Antrieb
1, LKW, Motor
2, PKW, Motor
3, MOTORRAD, Motor
4, FUSSGAENGER,  Muskel
```

Überprüfen Sie, dass Sie zu allen Datenbanken Zugriff haben und machen Sie sich mit den Daten vertraut.

Aufgabe 1: Datenbank anlegen

Erstellen Sie das eingangs beschriebene Starschema unter Berücksichtigung von Fremdschlüsseln. Wählen Sie geeignete Datentypen.

Aufgabe 2: Integration der Dimensionsinformationen

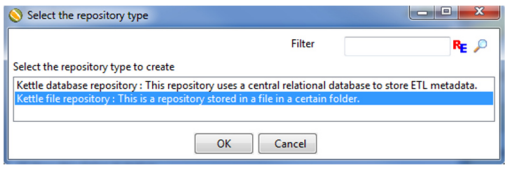
Füllen Sie nun die denormalisierten Dimensionstabellen mit geeigneten Daten, sodass die in der Datenbank Verkehrsdaten zur Verfügung stehenden Messwerte in der nächsten Aufgabe in die Faktentabelle eingetragen werden können. Sie können hierfür das Werkzeug pentaho nutzen, müssen es aber nicht.

Aufgabe 3: Integration der Fakten (Daten Tabelle Mesungen)

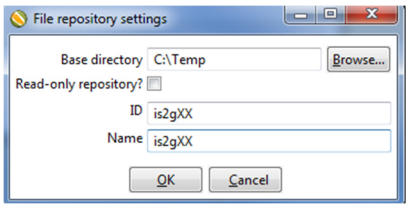
Ziel der Aufgabe ist es nun, die Daten aus der Datenbank fgdb_`verkehrsdaten` zu lesen (**E**xtraction), sie in ein entsprechendes Format zu transformieren (**T**ransform) und die Daten dann in ihre Datenbank zu laden (**L**oad). Nun ist der Einsatz von Pentaho verpflichtend.

Dazu können Sie wie folgt vorgehen:

a) Sofern Sie Java installiert haben, können Sie es je nach Betriebssystem über Spoon.sh starten. In den Pools steht PDI zur Verfügung. Rufen Sie zunächst das Skript zum setzen der Umgebungsvariablen au. Zunächst müssen Sie ein Repository anlegen, welches Ihre Projekte enthält. Klicken Sie dazu auf das „grüne Pluszeichen“ und wählen „Kettle file repository“ aus:



Geben ein Verzeichnis an, in dem PDI seine Daten speichern kann. Geben Sie als ID und Name ihre Gruppenidentifikation g<gruppennummer> mit entsprechender Ersetzung von <gruppennummer> an.



Klicken Sie OK und wählen Sie dann Ihr erstelltes Repository aus. Statt C:\\Temp wählen Sie bitte ein Netzlaufwerk. Sie können auch ein synchronisiertes Netcase-Verzeichnis wählen, um den Austausch in der Gruppe zu erleichtern.

b) Pentaho Data Integration (PDI) hilft Ihnen bei der Integration der Daten. Dazu müssen Sie zunächst eine neue Transformation erstellen. Dann können Sie verschiedene Operationen auswählen und so ihre Datenintegration durchführen. Eine Dokumentation über die Funktionen finden Sie online[1] und im Ordner *docs* von PDI. Hierzu einige nützliche Hinweise:

   i. Sie benötigen ein `Table input`, um aus der Datenbank `verkehrsdaten` zu lesen.
   ii. Sie benötigen ein `Table output`, um in Ihre Datenbank zu schreiben.
   iii. Sowohl beim Schreiben als auch beim Lesen aus einer Datenbank müssen sie eine entsprechende Verbindung anlegen!

1 http://wiki.pentaho.com/display/EAI/Latest+Pentaho+Data+Integration+%28aka+Kettle%29+Documentation

iv. Mit dem Schritt `Combination lookup/update` können Sie für bestimmte Daten einen Lookup machen. Dabei können sie für eine Auswahl an Attributen in einer Tabelle den Datensatz finden, der dieselben Attribute besitzt.
Dies ist z.B. nützlich, um die Dimension nachzuschlagen. Wenn die Dimension gefunden bzw. eingetragen wurde, wird diese dem Datensatz angehängt. Dabei wird der Technical Key zum Anhängen verwendet, der dem Schlüssel der Dimensionstabelle entsprechen sollte.

v. Mit `Select values` lassen sie einzelne Attribute herausfiltern (=Projektion) oder auch umbenennen (dies ist z.B. notwendig, um nicht mehrmals das Attribute id aus einem lookup zu haben; Vermeidung von Doppeldeutigkeit).

vi. Gegebenenfalls müssen Sie eine Datenbanktabelle beim Schreiben leeren (`truncate`), damit sie nicht veraltete Daten enthält.

vii. Mit dem Schritt `Calculator` können Felder mit vordefinierten Funktionen bearbeitet werden (z.B. kann aus einem Datum Jahr, Monat und Tag berechnet werden).

Das Ergebnis können Sie sich dann wiederrum über ein Datenbank-Tool anschauen.
Als Tipp: Die Lösung kann mit 10 Schritten mit den oben genannten Funktionen (`Table input, Table output, Select values, Calculator und combination lookup/update`) gemacht werden.