CAS Information Engineering

## Leistungsnachweis für Modul DB & DWH

**Gruppenmitglieder**:

Andreas Fischer

Bernd Novotny

Tobias Schieferdecker

Inhaltsverzeichnis:

Leistungsnachweis für Modul DB & DWH 1

1. Übersetzung des gegeben ERMs in den uns bekannten Dialekt 1

2. Auslagern der relevanten Daten in csv-Dateien 1

3. Erstellung der staging-area 2

4. Reparieren der korrupten Daten 2

5. Konzipierung des Sternschemas für das DWH 2

6. Erstellung und Befüllen des DWH 2

7. Auswertungen auf Basis des DWH 2

8. Fazit und Schlusswort 2

# **Übersetzung des gegeben ERMs in den uns bekannten Dialekt**

# **Auslagern der relevanten Daten in csv-Dateien**

# **Erstellung der staging-area**

# **Reparieren der korrupten Daten**

# **Konzipierung des Sternschemas für das DWH**

Für das multidimensionale Modell haben wir ein einfaches Stern-Schema gewählt. Die Faktentabelle im Zentrum basiert auf der Relation “order\_details” aus dem Ausgangsschema. In “order\_details” finden sich die Einzelpositionen aller Bestellungen, über die sich geeignete Auswertungen durchführen lassen.

So sollen später zum Beispiel Umsätze pro Monat oder Gewinnzahlen für bestimmte Produkte ermittelt werden.

Bei den Dimensionen beschränken wir uns auf die Bereiche «Kunde», «Produkt», «Region», «Bestellung» und «Zeit». Zusätzlich wären Auswertungen über die Spediteure, die Lieferanten und die zuständigen Sachbearbeiter denkbar, was entsprechende weitere Dimensionen erfordern würde.

Bei der Dimension «Region» wäre eine Erweiterung auf das Schneeflocken-Schema denkbar gewesen, da es sich bei Ländern, Bundesstaaten und Städten um unterschiedliche Granularitäten räumlicher Angaben handelt. Allerdings erlaubt die geringe Datenmenge ohnehin keine fein granularen Auswertungen, so dass wir auf diese Unterteilung verzichtet haben.

Abbildung 5‑1zeigt das vollständige Schema.

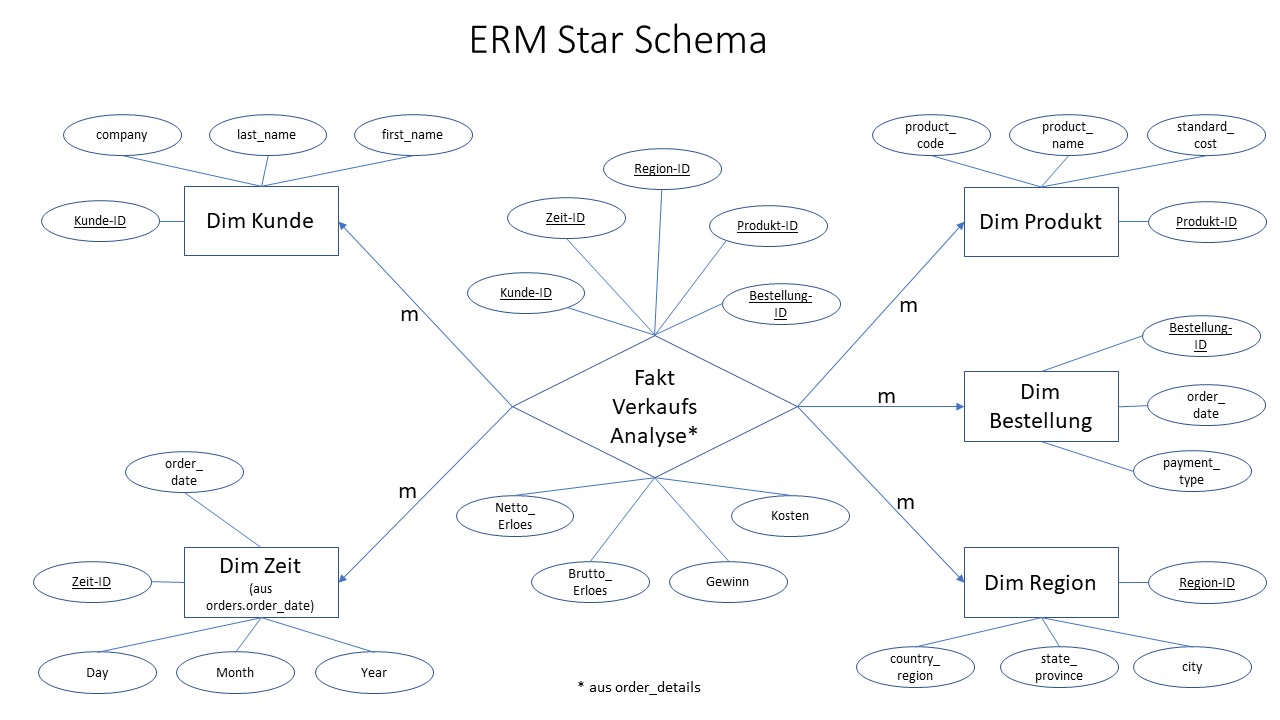


Abbildung 5‑1

# **Erstellung und Befüllen des DWH**

Das Ziel-Schema wurde direkt durch SQL-Befehle in MySQL implementiert. Die notwendigen Erweiterungen, insbesondere für “Slowly Changing Dimensions” werden durch Pentaho vorgenommen. Hier die SQL-Befehle zum Erstellen des Data Warehouse:

DROP DATABASE IF EXISTS NORTHWIND\_DWH;

CREATE DATABASE NORTHWIND\_DWH;

USE NORTHWIND\_DWH;

CREATE TABLE kunde\_dim (

KUNDE\_ID\_DIM INT(11) NOT NULL AUTO\_INCREMENT,

CUSTOMER\_ID INT(11) NOT NULL,

COMPANY VARCHAR(50) NULL DEFAULT NULL,

FIRST\_NAME VARCHAR(50) NULL DEFAULT NULL,

LAST\_NAME VARCHAR(50) NULL DEFAULT NULL,

CONSTRAINT PK\_KUNDE\_DIM PRIMARY KEY (KUNDE\_ID\_DIM)

);

CREATE TABLE bestellung\_dim (

BESTELLUNG\_ID\_DIM INT(11) NOT NULL AUTO\_INCREMENT,

BESTELLUNG\_ID INT(11) NOT NULL,

PAYMENT\_TYPE VARCHAR(11) NULL DEFAULT NULL,

ORDER\_DATE DATE,

CONSTRAINT PK\_BESTELLUNG\_DIM PRIMARY KEY   
 (BESTELLUNG\_ID\_DIM)

);

CREATE TABLE produkt\_dim (

PRODUKT\_ID\_DIM INT(11) NOT NULL AUTO\_INCREMENT,

PRODUKT\_ID INT(11) NOT NULL,

PRODUKT\_CODE VARCHAR(25) NULL DEFAULT NULL,

PRODUKT\_NAME VARCHAR(50) NULL DEFAULT NULL,

STANDARD\_COST DECIMAL(19 , 4 ) NULL DEFAULT '0.0000',

CONSTRAINT PK\_PRODUKT\_DIM PRIMARY KEY (PRODUKT\_ID\_DIM)

);

CREATE TABLE zeit\_dim (

ZEIT\_ID\_DIM INT(11) NOT NULL AUTO\_INCREMENT,

ORDER\_DATE DATE,

ORDER\_DAY INTEGER,

ORDER\_MONTH INTEGER,

ORDER\_YEAR INTEGER,

CONSTRAINT PK\_ZEIT\_DIM PRIMARY KEY (ZEIT\_ID\_DIM)

);

CREATE TABLE region\_dim (

REGION\_ID\_DIM INT(11) NOT NULL AUTO\_INCREMENT,

CITY VARCHAR(25) NULL DEFAULT NULL,

STATE\_PROVINCE VARCHAR(2) NULL DEFAULT NULL,

COUNTRY\_REGION VARCHAR(15) NULL DEFAULT NULL,

CONSTRAINT PK\_REGION\_DIM PRIMARY KEY (REGION\_ID\_DIM)

);

CREATE TABLE verkauf\_fact (

KUNDE\_ID\_DIM INT(11) NOT NULL,

PRODUKT\_ID\_DIM INT(11) NOT NULL,

ZEIT\_ID\_DIM INT(11) NOT NULL,

REGION\_ID\_DIM INT(11) NOT NULL,

BESTELLUNG\_ID\_DIM INT(11) NOT NULL,

BRUTTO\_ERLOES INTEGER,

NETTO\_ERLOES INTEGER,

KOSTEN INTEGER,

GEWINN INTEGER

);

Das Befüllen des Schemas mit den Daten erfolgt über zwei Pentaho Transformationen:

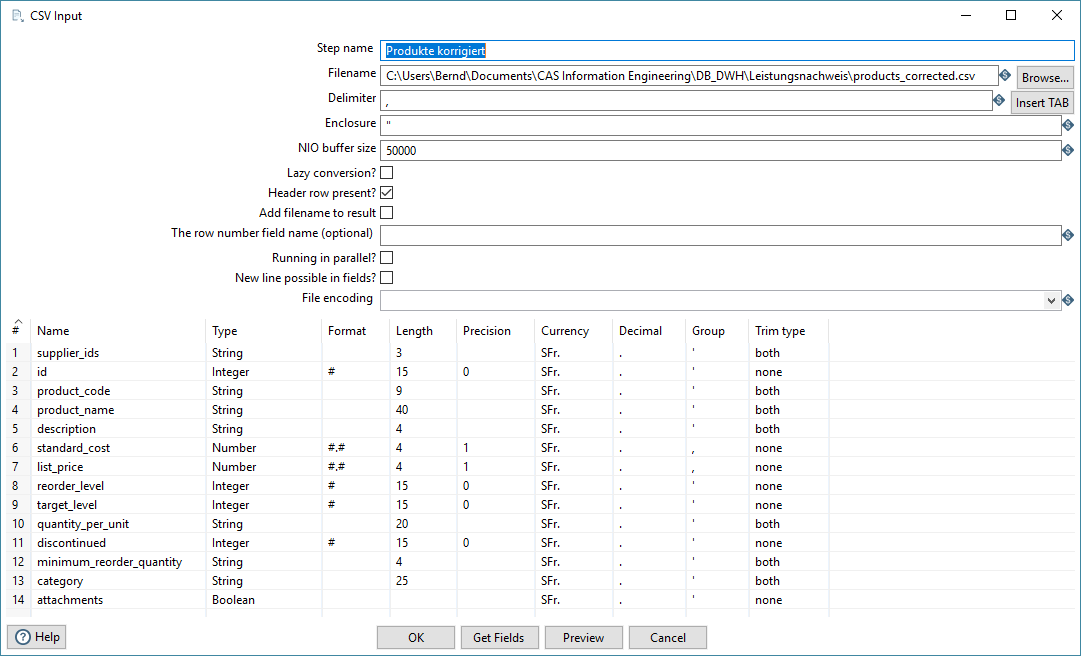
1. “star-schema”: diese Transformation befüllt die Dimensionstabellen
2. “”facts-table”: durch diese Transformation wird die Fakten-Tabelle befüllt

**Befüllen der Dimensionstabellen**

Wir beginnen mit dem Befüllen der Produkte-Dimension. Diese besteht aus zweit Schritten:

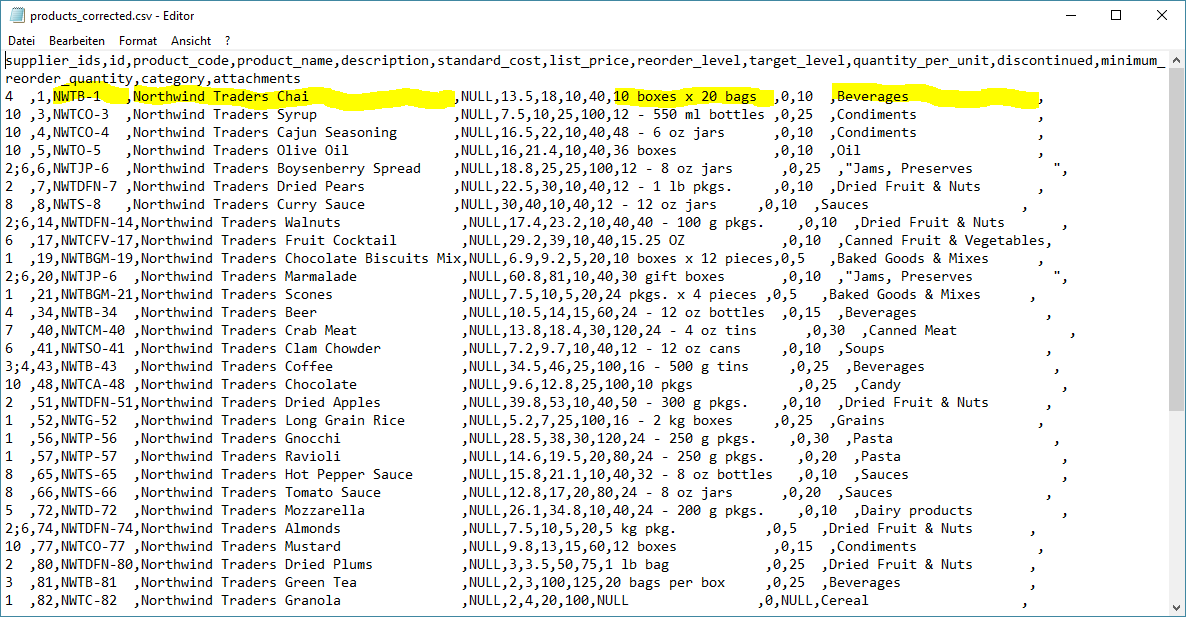


Der Schritt “Produkte” liest die Informationen aus der Staging Area ein, d.h. das csv-File das wir in den Schritten 2 bis 4 erstellt haben. Im konkreten Fall das File “products-corrected.csv”. Dazu wird in Pentaho das Element “Input -> CSV file input” gewählt und mit einem Doppelklick geöffnet. Der Schritt wird umbenannt, der Pfad auf das CSV-File selektiert und als Delimiter das Komma gesetzt.



Nachdem mit “Get Fields” die Attribute aus dem CSV File identifiziert wurden, werden die Datentypen überprüft. Wo nötig werden Anpassungen vorgenommen. Vor allem bei String Feldern könnten in Zukunft längere Werte vorkommen, so dass sich unter Umständen eine Erhöhung der Feldlänge lohnt.

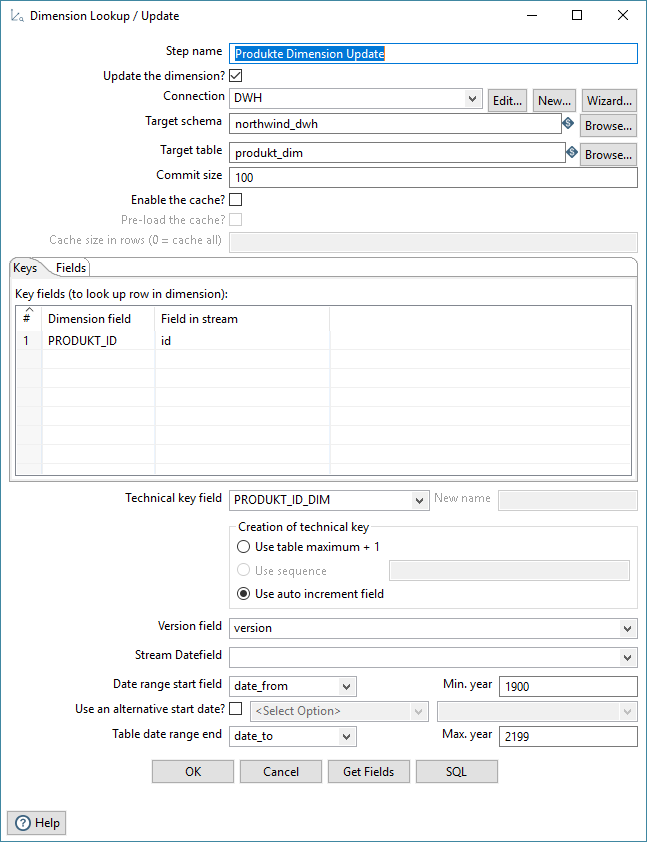
Ein Blick auf das CSV File zeigt zudem, dass sich durch den Korrekturschritt im CSV File Leerzeichen eingeschlichen haben:



Daher wird im Transformationsschritt beim Datentyp String unter «Trim type» die Option «both» gewählt, um die Leerzeichen zu entfernen. Die Leerzeichen scheinen zwar nur am Ende eingefügt zu sein, aber um auf sicher zu gehen werden Leerzeichen am Anfang und am Ende des Strings entfernt.

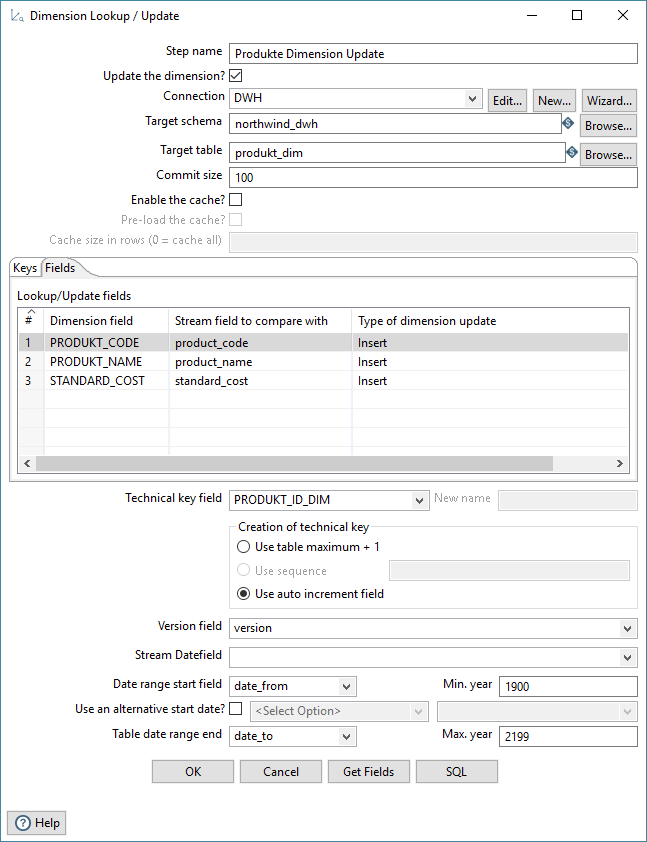
Mit «Preview» werden die Daten nochmal überprüft und dann mit «OK» der Schritt geschlossen.

Im zweiten Transformationsschritt werden aus dem Eingabe-Stream die für die Dimensionstabelle relevanten Attribute extrahiert und dann in die Datenbank geschrieben. Dafür wählen wir das Transformationselement «Data Warehouse -> Dimension lookup/update» und verbinden es mit dem vorherigen Schritt. Ein Doppelklick öffnet folgendes Fenster:



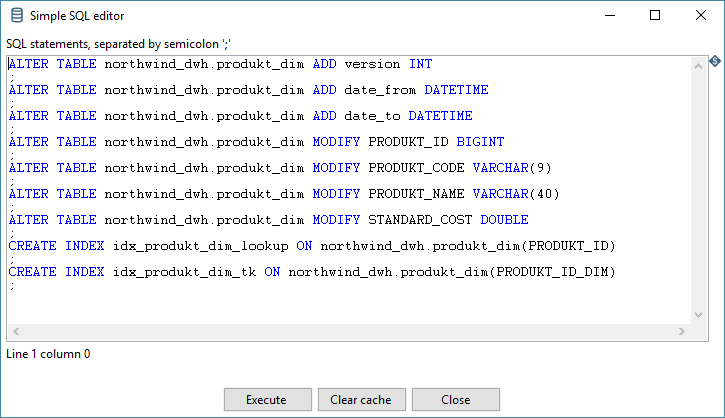
Wir lassen Pentaho den Surrogat-Schlüssel «PRODUKT\_ID\_DIM» befüllen und verwenden den Auto Increment der Datenbank.

Um die für uns interessanten Felder der Dimension zu wählen und «Slowly Changing Dimensions» vom Typ 2 zu implementieren, wechseln wir ins Tab «Fields»:

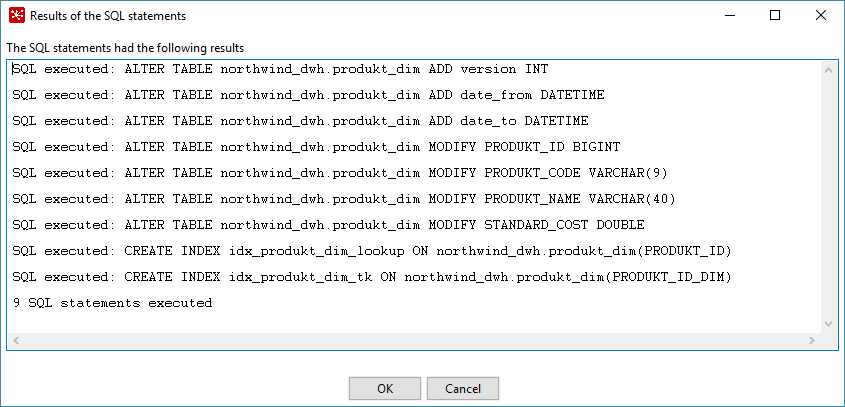


Wie gut zu erkennen ist, beschränken wir uns auf drei Felder aus dem Eingabe-Stream (product\_code, product\_name und standard\_cost) und mappen sie auf die Felder unseres Datenbankschemas. Für die Typ 2 SCD wird als «Type of dimension update» der Wert «Insert» gewählt.

Bevor die Transformation ausgeführt werden kann, muss dass generierte SQL ausgeführt werden. Sonst fehlen die notwendigen Schemaänderungen in der Datenbank. Dazu wird auf den Button «SQL» geklickt:



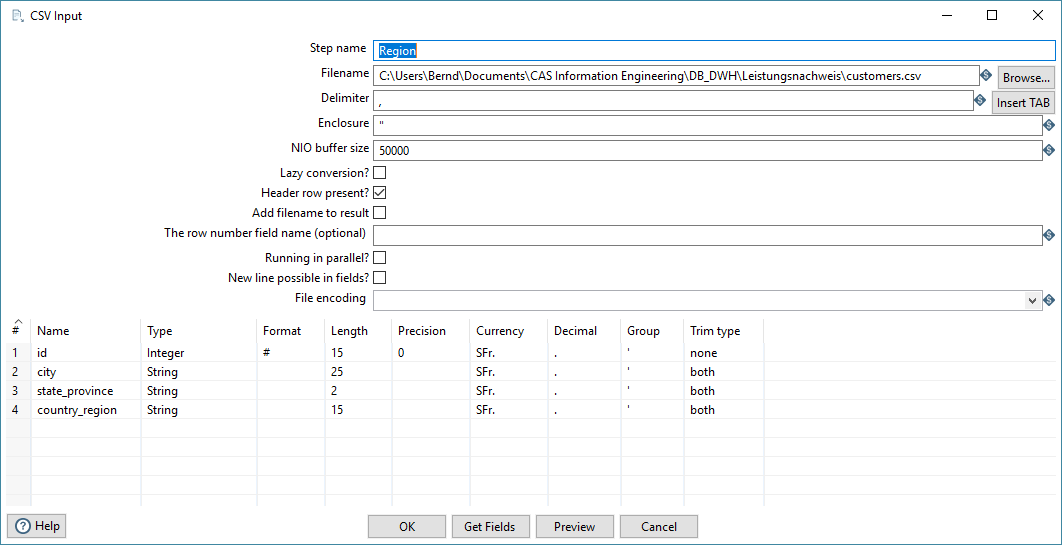
Ein Klick auf «Execute» führt die notwendigen Änderungen in der Datenbank durch. Pentaho bestätigt uns die erfolgreiche Ausführung:



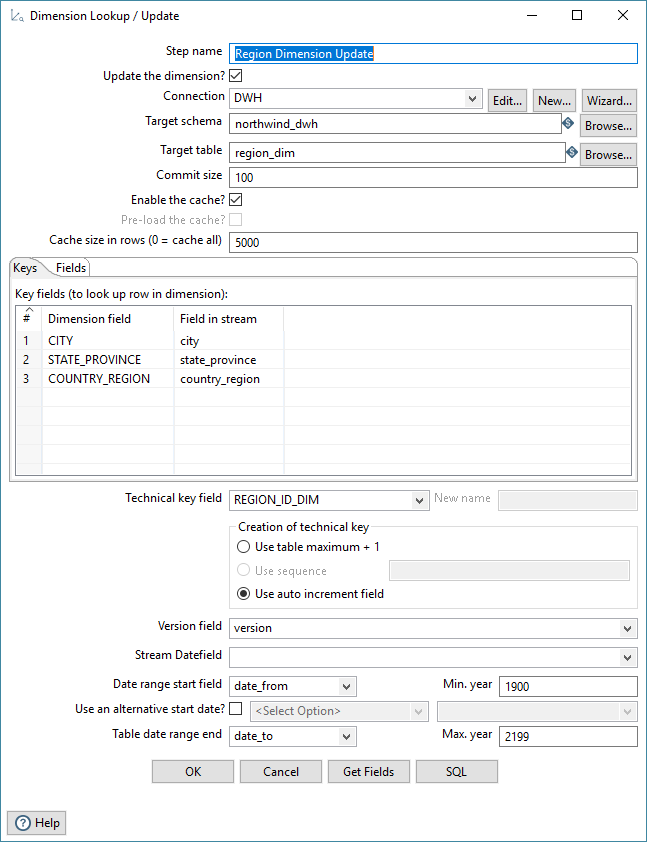
Danach können die Fenster durch «OK» bzw. «Close» wieder geschlossen werden.

In analoger Weise wird für die Dimensionen «Kunde» und «Bestellung» vorgegangen.

Bei «Region» ist zu beachten, dass dafür in der Ausgangs-DB keine eigene Relation vorliegt. Wir beziehen diese Daten aus der Relation «customers» über die Felder «city», «state\_province» und «country\_region».

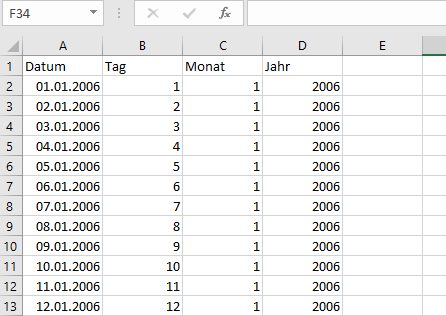


Wir dürfen für die Identifikation der Zeilen nicht den Schlüssel «id» dieser Relation heranziehen, da wir sonst doppelte Einträge in der Dimensionstabelle erhalten würden. Stattdessen wählen wir die Kombination der drei Attribute als Key für die Identifikation von Datensätzen in der customers-Relation.

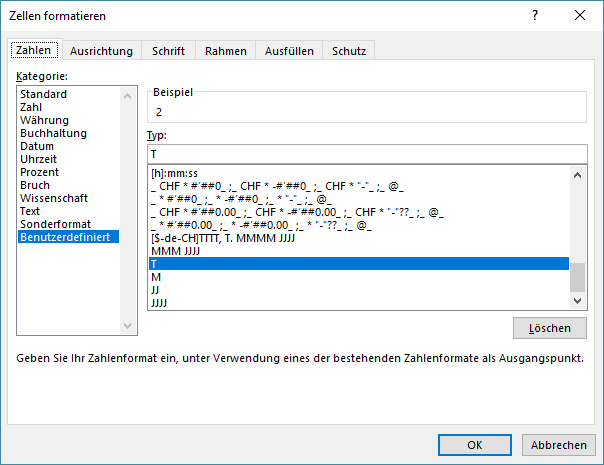


Zusätzliche Felder sind nicht zu wählen, da wir in der Dimensionstabelle nur an diesen Werten interessiert sind.

Für die Zeit-Dimension schauen wir zuerst in den Ausgangsdaten, welcher Zeitraum relevant ist. Offensichtlich handelt es sich um das Jahr 2006. Die Dimensionstabelle soll als mit den Tagen dieses Jahres gefüllt werden. Der einfachste Ansatz wäre gewesen, eine solche Tabelle mit ein paar Mausklicks in Excel zu erzeugen, als CSV zu speichern und dieses dann in gewohnter Weise einzulesen und in das DWH zu schreiben.



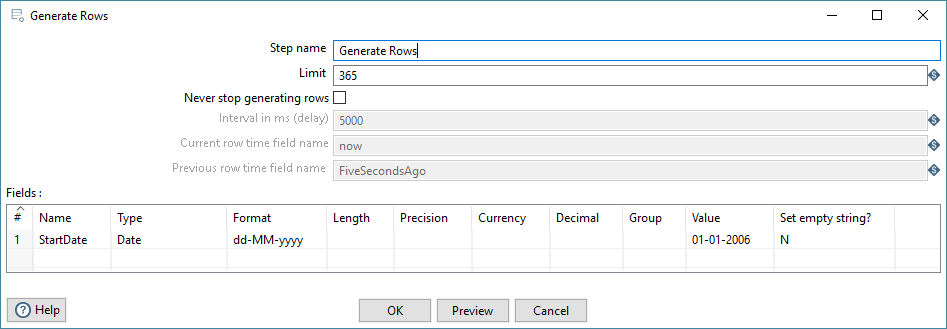
Die Spalte B lässt sich leicht duch eine benutzerdefinierte Formatierung füllen. Gleiches gilt dann für die Spalten C und D.



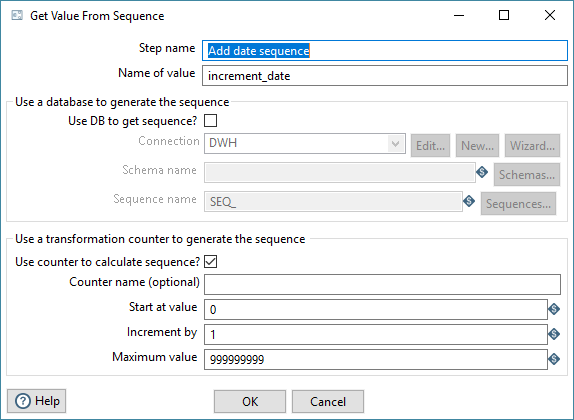
Statt des Ansatzes mit Excel haben wir einen Ansatz mit Pentaho Transformationsschritten gewählt:



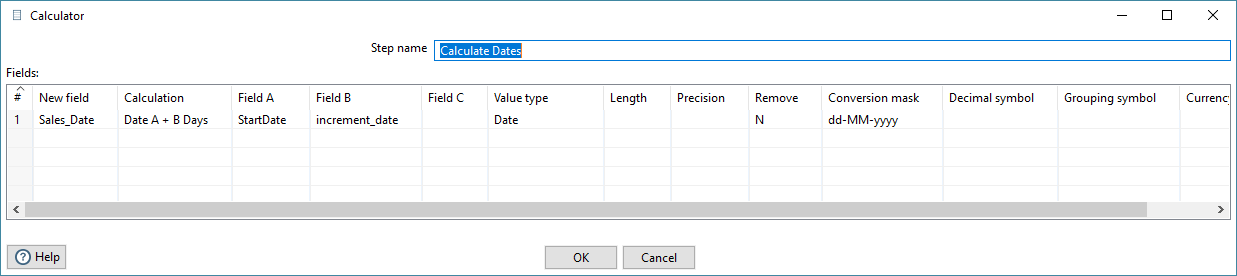
Im ersten Schritt werden mit dem Element «Generate Rows» 365 Zeilen erzeugt, für jeden Tag des Jahres eine, beginnend am 1 Januar 2006:



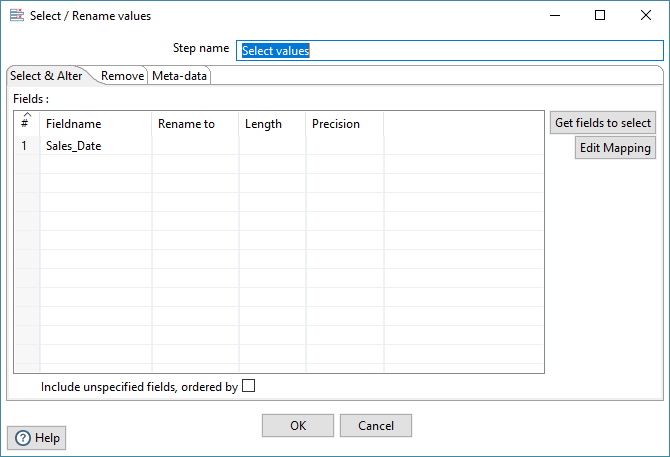
Nach diesem Schritt steht zunächst in jeder Zeile der 1. Januar 2006. Im zweiten Schritt wird ein Zähler generiert («Add Sequence»), der für jedes Element, das er aus dem Eingangsstream erhält jeweils um 1 hochzählt. Wir erhalten also eine Sequenz von 0 bis 364



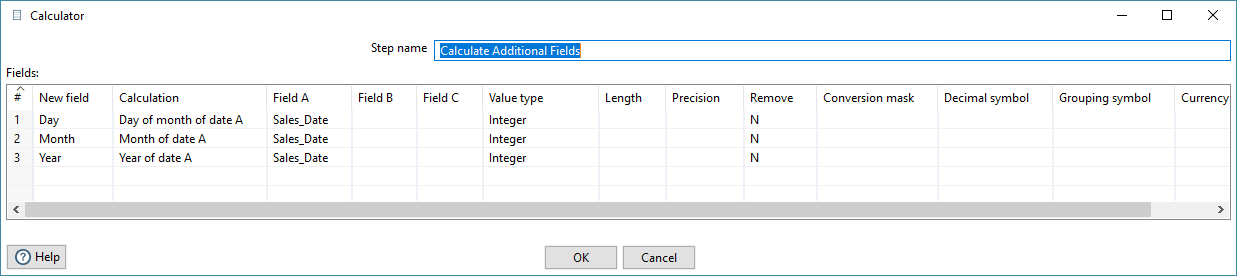
Im dritten Schritt wird dieser Zähler auf die Datumssequenz angewendet («Calculator» Schritt). Wichtig ist es, bei «Conversion Mask» das gleiche Format wie im ersten Schritt zu definieren.



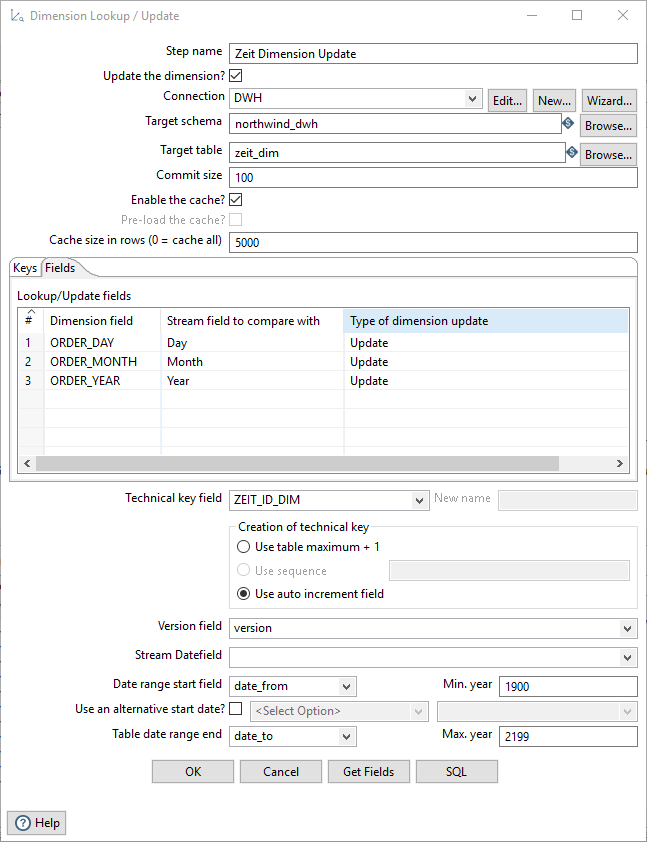
Nun wird der Datenstrom wird auf das Feld reduziert, um das es eigentlich geht: das Datumsfeld. Das geschieht mit Hilfe des «Select Values» Elements.

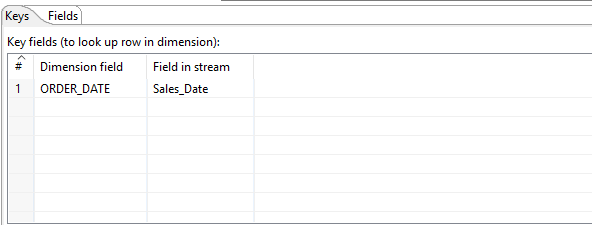


Für die Sequenz an Datumsfeldern (ein Element pro Tag im Jahr) können nun die anderen Felder der Zeit-Dimensionstabelle berechnet werden mithilfe eines «Calculator» Schritts:

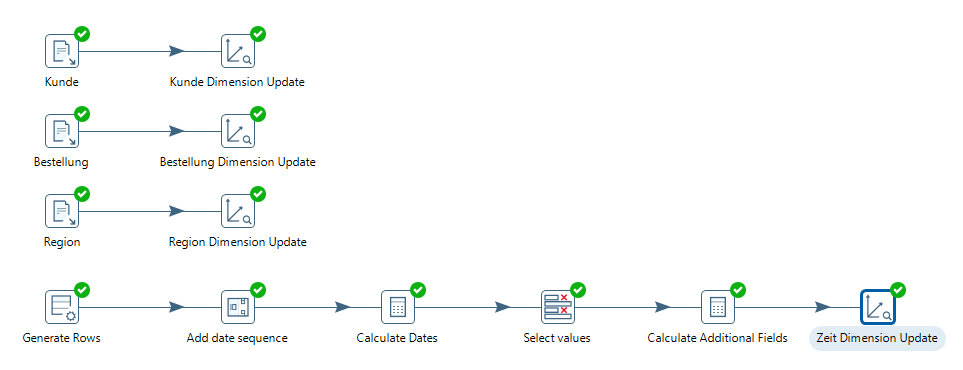


Als abschliessender Schritt wird die Dimensionstabelle wie zuvor gefüllt:

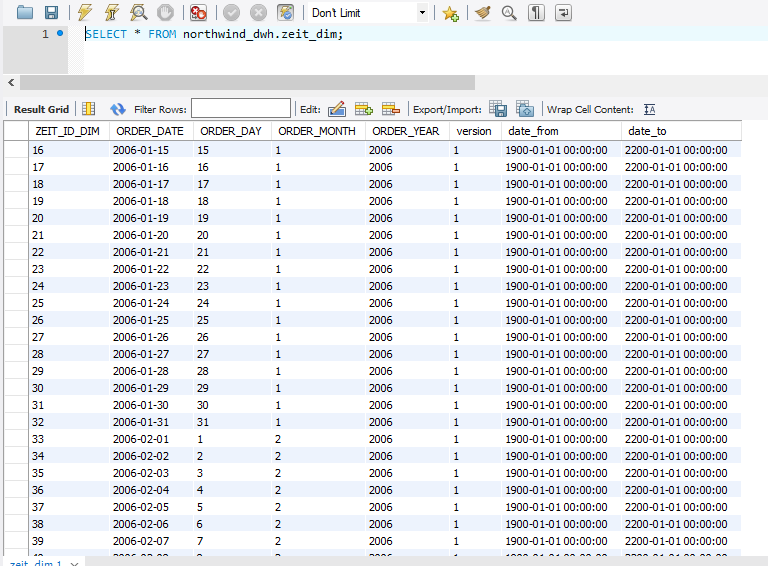




Nun kann die komplette Transformation ausgeführt werden:

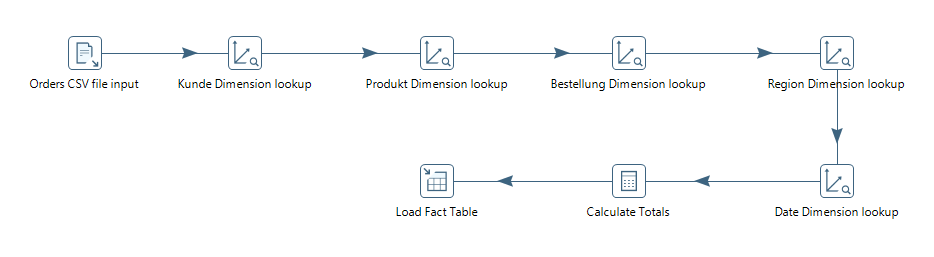


Ein exemplarischer Blick in die Datenbank auf die Dimensionstabellen am Beispiel der Zeit-Dimension:

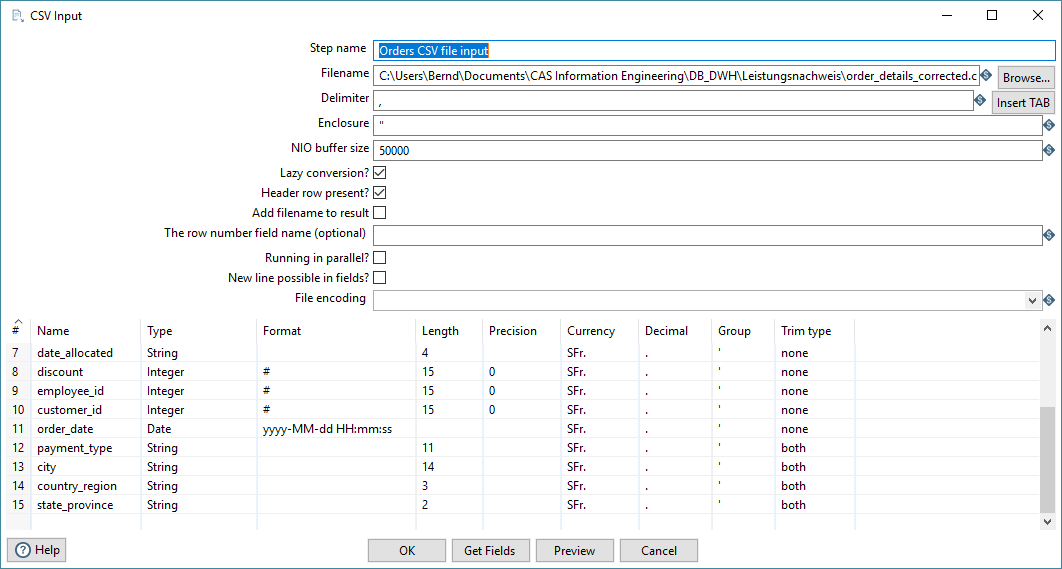


**Erzeugen der Fakten**

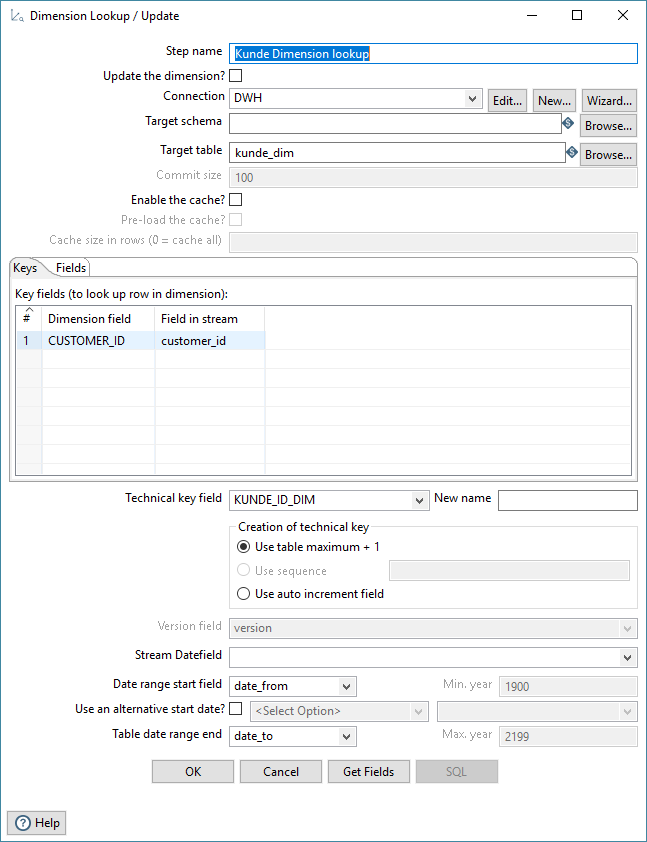
Wie eingangs erwähnt basiert die Faktentabelle auf «order\_details» aus der Ausgangs-DB. Diese Relation wird im ersten Schritt eingelesen.



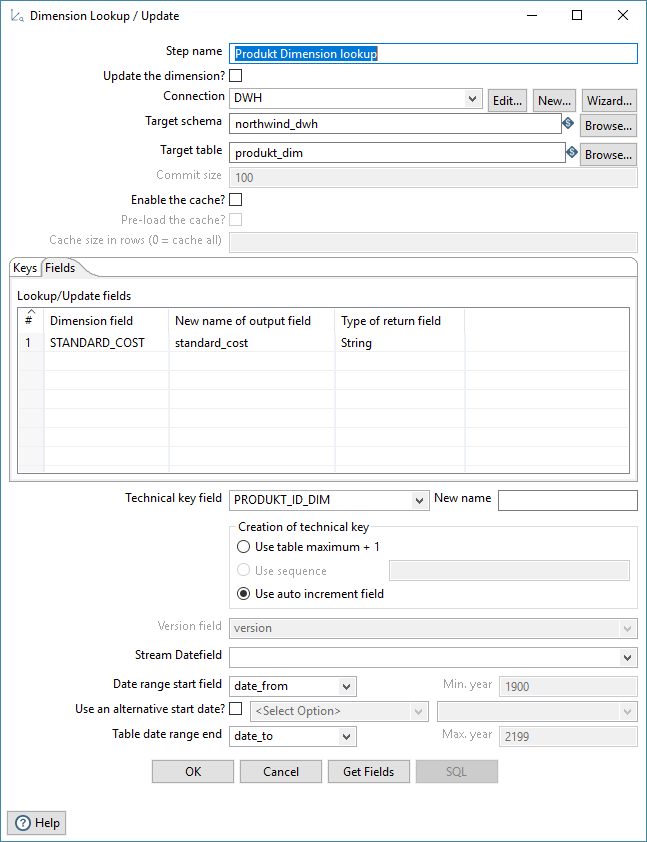
Auch hier schauen wir wieder, dass allfällige Leerzeichen am Anfang und Ende von String entfernt werden, damit das Matching sauber funktioniert.



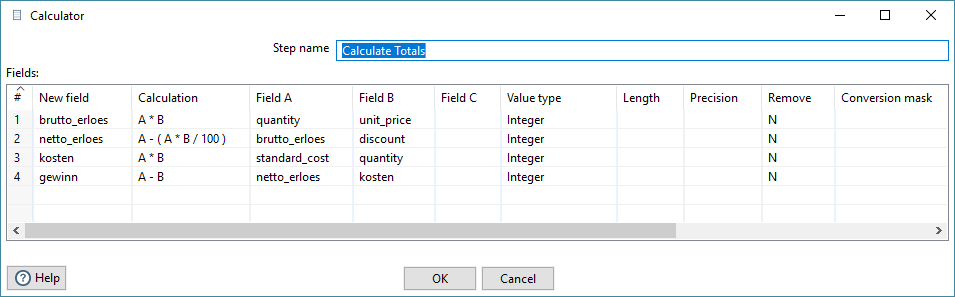
Danach werden im «Dimension lookup/update» Schritten für jeden Datensatz die Surrogatschlüssel aus den Dimensionstabellen gelesen.



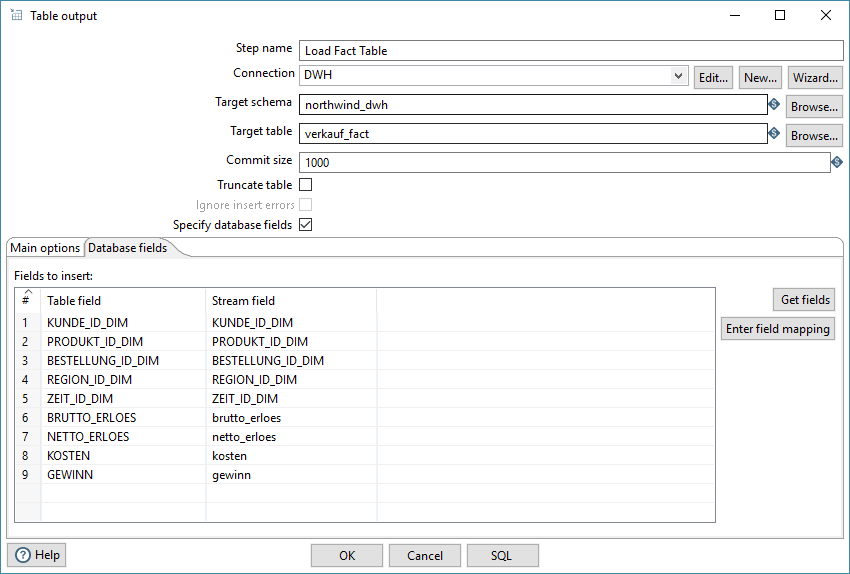
Für die abschliessende Kalkulation der für die Faktentabelle definierten Werte (Netto\_Erloes, Brutto\_Erloes, Gewinn und Kosten) wird aus den Produkten noch der Wert für die Standard-Kosten in den Datenstream eingefügt.



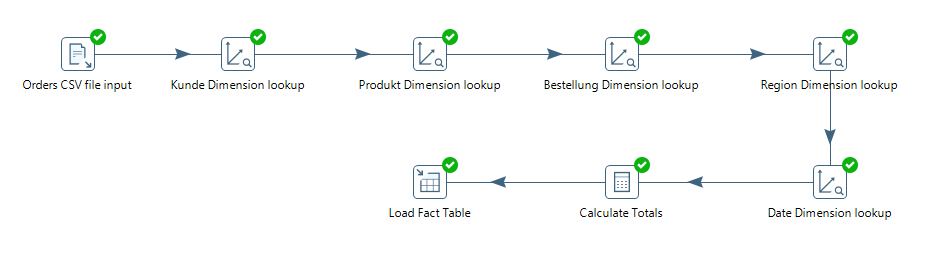
Diese Information wird dann im «Calculator» Step (vorletzter Schritt der Transformation) in den Berechnungen genutzt:



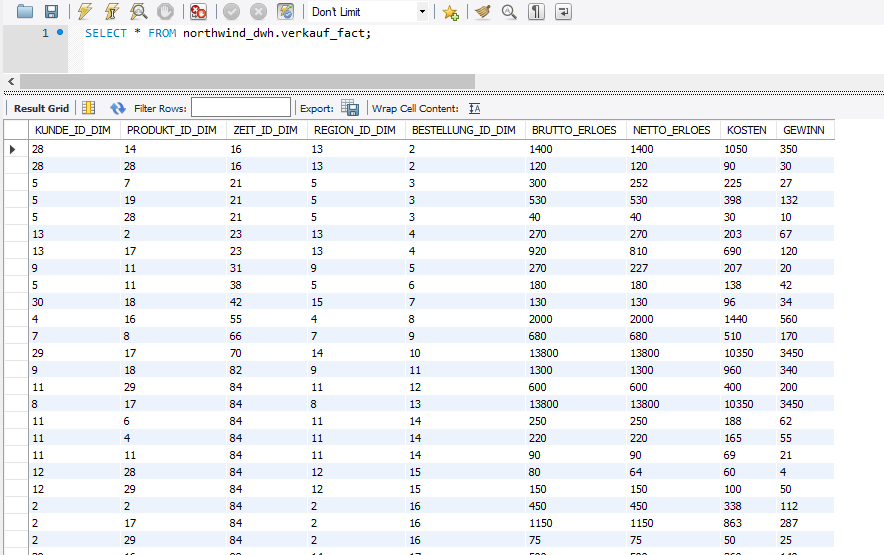
Im letzten Schritte werden dann alle Datensätze der Faktentabelle einschl. kalkulierter Attribute in das Data-Warehouse geschrieben:



Nach dem alle Schritte definiert sind, lassen wir die Transformation laufen.

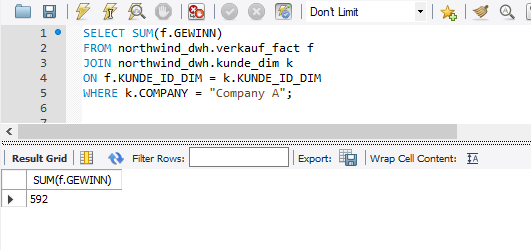


Ein Kontrollblick in die Datenbank bestätigt, dass die Faktentabelle geschrieben wurde:

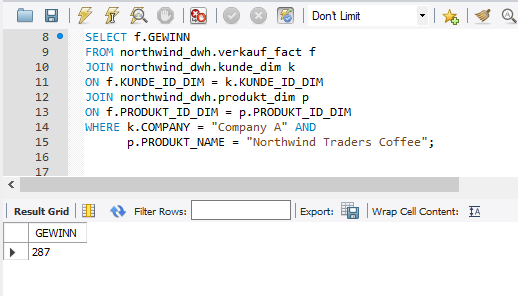


# **Auswertungen auf Basis des DWH**

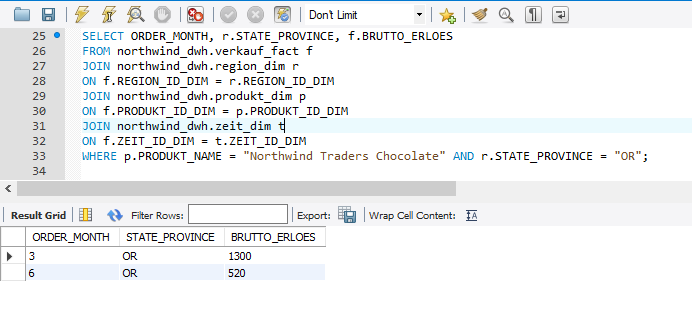
Wir beginnen mit einer sehr einfachen Auswertung, bei der wir den Gesamtgewinn berechnen, den wir mit unserem Kunden «Company A» erzielt haben:



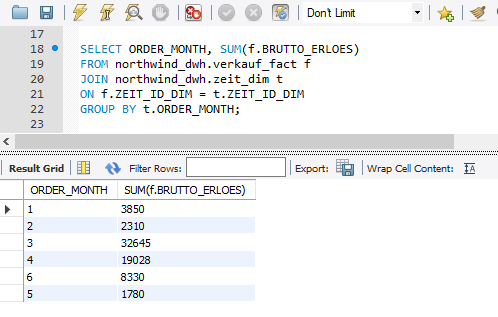
Als nächstes wollen wir uns anschauen, wieviel Gewinn wir mit diesem Kunden mit unserem Produkt «Northwind Traders Coffee» erzielt haben:



Beim Produkt «Northwind Traders Chocolate» interessiert uns nun vor allem, wie sich der Verkauf bei unseren Kunden in Oregon im Lauf des Jahres entwickelt hat.



Und last but not least wollen wir uns einen Überblick über die monatlichen Brutto-Erlöse verschaffen, d.h. die Erlöse vor Abzug aller Rabatte:



# **Fazit und Schlusswort**