

Übung 1:

Abgabe vor der zweiten Vorlesung (d.h. vor 05.12.2002).

Hinweise:

- Unbedingt die Hinweise bezüglich Installation auf Win XP Systemen beachten
- Teradata Demo 4.1 funktioniert nicht auf Windows 2000 System mit installiertem ServicePack 3. Ein Patch ist derzeit nicht verfügbar. Einzige Lösung: SP3 deinstallieren.
- Alle Produkte der Teradata Demo CD installieren (Teradata Demo, Samples, Java Runtime, Teradata Utilities)
- "Typical installations" während der Installation auswählen
- Rechner booten
- Die Demobeispiele ansehen: "Programs/Teradata for MS Windows Demo/Sample Applications"!
- Probiert insbesondere das "Data Load" Beispiel aus!
- Während die Applikation läuft, seht euch die Dateien in folgendem Verzeichnis an:

C:\Program Files\NCR\Teradata for Microsoft Windows Demo\bin\data

Es gibt dort folgende Dateien:

- *.bat → Jeder Menüeintrag in der Demo-Applikation startet ein Batch-File
- *.btq → BTEQ command files
- *.fl → fastload command files
- *.dat → Datendateien

Aufgabenstellungen Teil 1:

1. Beispielprogramme der Teradata Demo ausprobieren
2. Modifikation der Beispielprogramme, sodass ihr in selbsterstellte Datenbanken/User und Tabellen laden könnt.
3. Schreibt komplexe Queries und verwendet das Visual Explain Tool (VEComp) um den Execution Plan zu analysieren.

Aufgabenstellungen Teil 2:

1. Die Datei <http://www.ifs.tuwien.ac.at/~bruckner/dwh/lab1.zip> herunterladen. Die Datei enthält spezielle Daten, die in ein DWH das nach Star-Schema-Modell modelliert ist, geladen werden. In der ZIP-Datei ist auch eine **README**-Datei enthalten, die unbedingt gelesen werden sollte.
2. Mit den Testdaten sollen folgende Schritte durchgeführt werden:
 - Erzeugen einer neuen Datenbank
 - Anlegen der notwendigen Tabellen des Star-Schemas
 - Laden der Tabellen mit **Fastload** (je ein Fastload-Job pro Tabelle)
 - Zumindest eine der Tabellen soll mit **Multiload** geladen werden
 - Durchführung & Dokumentation von diversen Tests: eigene zusätzliche Select-Statements ausprobieren, Execution Plans, eventuell Vergleich Multiload - Fastload
3. Erstellung eines kurzen Protokolls (für das Abgabegespräch) worin die notwendigen Schritte beschrieben sind, die ihr durchgeführt habt, um das Datenfile zu laden und eure Experimente auszuführen.

README zum Lab1.Zip-File:

- * tables.sql
Definition der Tabellen (in einer Pseudo ANSI-SQL Notation)
 - für den Datentyp [money] könnt ihr den Decimal(10,2) Datentyp in Teradata verwenden.
 - die Constraint-Definition für die Primary Keys kann in Teradata für den Primary Index verwendet werden.Achtung: das Teradata-SQL hat einige Abweichungen gegenüber der Pseudo ANSI-SQL Notation, die im File verwendet wird. Dokumentation verwenden!

- * OLAPselect.sql
Beispiel für einen typischen OLAP-Select (On-Line Analytical Processing),
der durch das fertig geladene Beispiel-DWH beantwortet werden kann.
Schreibt zusätzlich eigene, komplexe Selects.

- * .CSV - Files
Die CSV (comma separated files)-Files enthalten die Rohdaten für das DWH, und sollen mittels Fastload geladen werden.
Für jede Tabelle (1 Fact-Tabelle, 5 Dimensions-Tabellen) gibt es jeweils ein CSV-File

- * time_dim.fld, time_dim.out
Beispiel für das Laden der Zeitdimension mit Fastload.
Datenbankname, User/Pwd müssen natürlich angepasst werden.
Anhand dieses fertigen Beispiels sollten die übrigen Dimensions- und die Faktentabelle "relativ" einfach zu laden sein. Ihr müsst nur die entsprechenden (fixen) Spaltengrößen im CSV-File beachten!
Mein mitgeliefertes Fastload-Skript ist ziemlich minimal, d.h. es sind beispielsweise keine Grant-Statements enthalten, wie bei den Demobeispielen.