Data Management and Visualization

Politecnico di Torino

Data warehousing in Oracle – Practice 1

The practice purpose is to first build a data warehouse compliant with the specifications listed in the following points, using Oracle. You then write some queries, in extended SQL, to retrieve data from the design data warehouse.

The outline of the practice is as follows:
1. Problem specifications
2. Description of the OLTP database
3. Exercise: design of the data warehouse
4. Exercise: comparison with the logical schema of the data warehouse
5. Exercise (SQL queries): query of the data warehouse

1. Problem specifications

A telephone company is interested in analyzing its own data to improve customer services. At present, the company has a database with call logs. For each call, the caller and receiver phone numbers, the duration, the type of charge (e.g., peak, off-peak rates), the start time (date, hour, minute, second) are known. The managers want to obtain very fast the information about the telephone traffic on the company lines and the daily income based on the caller location, the day and the phone rate.
In particular, the managers want to analyze the following situations:

- Monthly net income and number of calls for each caller city.
- Monthly net income and number of calls for each receiver city.
- Monthly net income and number of calls for each caller province and region.
- Monthly net income and number of calls for each receiver province and region.
- Daily net income and number of calls for each caller province.
- Yearly net income and number of calls for each caller province and region.
- Monthly net income and number of calls for each phone rate (type of charge).
- Net income and number of calls for each day of the week and phone rate.
- Daily number of calls for each caller region.
- Daily number of calls for each receiver region.

2. Description of the OLTP database

The OLTP database of the telephone company is reported in Table 1.

Tables	Description

DWABD.PHONERATES
`( phoneRateType INT NOT`
`NULL, phoneRateName VARCHAR(20)`
`NOT NULL, phoneRate_CostPerSecond FLOAT`
`NOT NULL,`
`PRIMARY KEY(phoneRateType) );`
	Different phone rates
	
	7 rows

DWABD.PLACES
`(`
`Places_ID INT NOT NULL,`
`City VARCHAR(20) NOT NULL,`
`Province VARCHAR(20) NOT NULL,`
`Region VARCHAR(20) NOT NULL,`
`PRIMARY KEY(Places_ID) );`
	Places
	
	1500 rows

DWABD.CALLS
`(`
`CallerPhoneNumber VARCHAR(20) NOT NULL,`
`ReceiverPhoneNumber VARCHAR(20) NOT NULL,`
`CallerLocation INT NOT NULL,`
`ReceiverLocation INT NOT NULL,`
`FullDate DATE NOT NULL,`
`StartTimeHour INT NOT NULL,`
`StartTimeMinute INT NOT NULL,`
`StartTimeSecond INT NOT NULL,`
`CallDuration FLOAT NOT NULL,`
`phoneRateType INT NOT NULL,`
`PRIMARY`
`KEY(CallerPhoneNumber,ReceiverPhoneNumber,FullDate,StartTimeHour`
`,StartTimeMinute,StartTimeSecond),`
`FOREIGN KEY(phoneRateType) REFERENCES PhoneRates(phoneRateType)`
`ON DELETE CASCADE,`
`FOREIGN KEY(CallerLocation)REFERENCES Places(Places_ID) ON`
`DELETE CASCADE,`
`FOREIGN KEY(ReceiverLocation) REFERENCES Places(Places_ID) ON`
`DELETE CASCADE`
`);`
	Calls in 2003 and 2004
	
	~ 1300000 rows

Table 1 – Source data base with single call information

3. Exercise: design of the data warehouse

Design the conceptual scheme of a data warehouse for managing the issues discussed above. The designed scheme must be designed to allow:

- The analyzes requested by the mobile phone company
- The ETL (extraction, transformation, loading) phase to import the data from the OLTP base (Table 1) to the data warehouse.

Tables	Description

DWABD.TIMEDIM
`(`
` ID_time        INT       NOT NULL,`
` DayDate        DATE      NOT NULL,`
` DayOfWeek      CHAR(15)  NOT NULL,`
` DateMonth      CHAR(15)  NOT NULL,`
` DateYear       INT       NOT NULL,`
` PRIMARY KEY(ID_time)`
`);`
	Time dimension
	
	10 rows

DWABD.PHONERATE
`(`
` ID_phoneRate        INTEGER     NOT NULL,`
`phoneRateType       VARCHAR(20) NOT NULL,`
` PRIMARY KEY(ID_phoneRate)`
`);`
	Phone rate dimension
	
	7 rows

DWABD.LOCATION
`(`
` ID_location          INTEGER     NOT NULL,`
` City                 VARCHAR(20) NOT NULL,`
` Province             CHAR(20)    NOT NULL,`
` Region               CHAR(20)    NOT NULL,`
` PRIMARY KEY(ID_location)`
`);`
	Place dimension
	
	1500 rows

DWABD.FACTS
`(`
` ID_time                  INTEGER NOT NULL,`
` ID_phoneRate             INTEGER NOT NULL,`
` ID_location_Caller       INTEGER NOT NULL,`
` ID_location_Receiver     INTEGER NOT NULL,`
` Price                    FLOAT   NOT NULL,`
` NumberOfCalls            INTEGER NOT NULL,`
`PRIMARY`
`KEY(ID_time,ID_phoneRate,ID_location_Caller,ID_location_Receiver),`
`FOREIGN KEY(ID_time)  REFERENCES timeDim(ID_time),`
` FOREIGN KEY(ID_phoneRate)   REFERENCES phoneRate(ID_phoneRate),`
` FOREIGN KEY(ID_location_Caller) REFERENCES location(ID_location),`
`FOREIGN KEY(ID_location_Receiver) REFERENCES location(ID_location)`
`);`
	Fact table
	
	7809 rows

Table 2 – Proposed solution - Data warehouse tables

4. Exercise: comparison with the logical schema of the data warehouse

Compare the conceptual scheme designed in the previous exercise with the logical scheme proposed in Table 2. Check you have correctly chosen the measures and the level of data aggregation. Looking at the logic schema in Table 2, answer the following questions:

- What are the measures chosen for the data warehouse?

- What is the minimum level of aggregation in the data warehouse? Does it correspond with what was designed in the conceptual schema?

5. Exercise (SQL Developer): Querying the data warehouse

The tables corresponding to the schema in Table 2 have already been created in Oracle and they contain some sample data. Use these tables as source for the following queries.

1. Select the yearly income for each phone rate, the total income for each phone rate, the total yearly income and the total income.
2. Select the monthly number of calls and the monthly income. Associate the RANK() to each month according to its income (1 for the month with the highest income, 2 for the second, etc., the last month is the one with the least income).
3. For each month in 2003, select the total number of calls. Associate the RANK() to each month according to its total number of calls (1 for the month with the highest number of calls, 2 for the second, etc., the last month is the one with the least number of calls).
4. For each day in July 2003, select the total income and the average income over the last 3 days.
5. Select the monthly income and the cumulative monthly income from the beginning of the year.

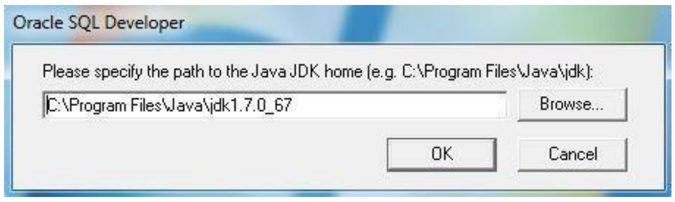
6. Connection to the database a Using Live SQL

With Oracle Live SQL, available at https://livesql.oracle.com/, you can create tables and query your database. To set up the environment, you must download the SQL scripts to create and populate the database, then you can directly query your database via the **SQL Worksheet** tab. A detailed guide is available here. **This is the suggested option to follow.**

[legacy] Using Oracle SQL program (on the Lab)

1. Open the Oracle SQL Developer program (from Start Menu-All programs)

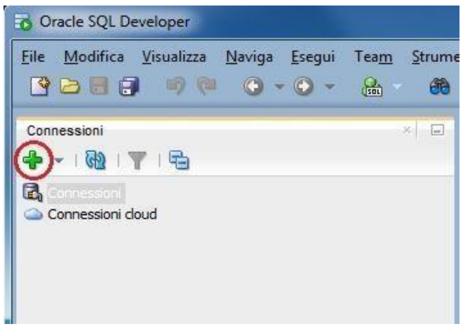
2. Select the Java SDK path


Oracle SQL Developer

Please specify the path to the Java JDK home (e.g. C:\Program Files\Java\jdk):

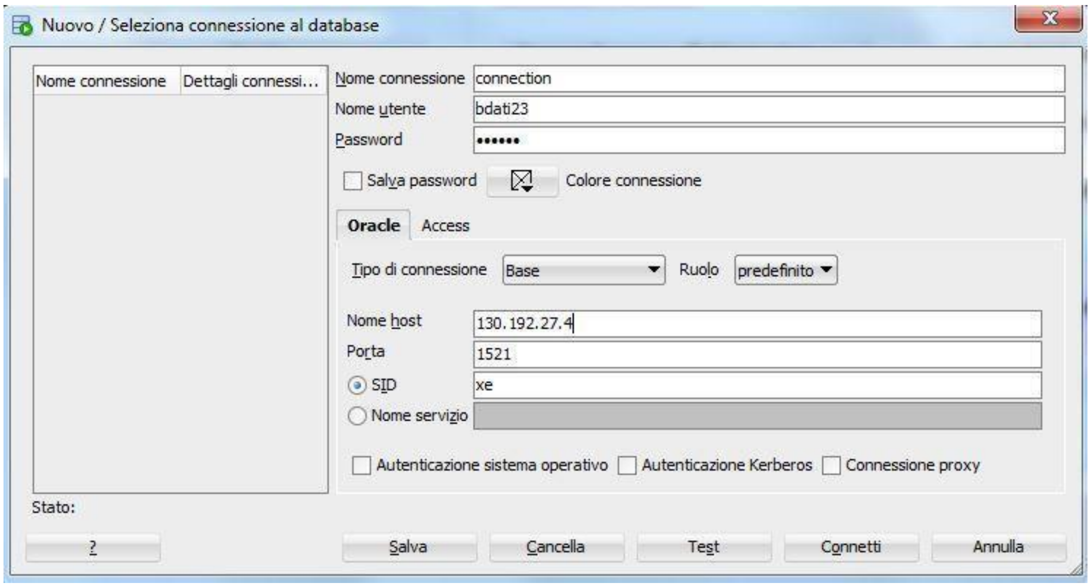
C:\Program Files\Java\jdk1.7.0_67

3. Click on the green "plus" button on the left to create a new connection



4. Login
    a. To logon you have to insert the following parameters:
        i. Nome utente (username): bdati[choose a number between 1-100]
        ii. Password: orac[choose a number between 1-100]
        iii. Nome host (host name): 130.192.27.4 iv. Port: 1521 v. SID: xe

For example, if you are working on pc number 23, the corresponding username is bdati23 and the password is orac23.


Nuovo / Seleziona connessione al database

Nome connessione	Dettagli connessi...

Nome connessione: connection
Nome utente: bdati23
Password: ••••••
☐ Salva password — Colore connessione

Oracle | Access

Tipo di connessione: Base — Ruolo: predefinito
Nome host: 130.192.27.4
Porta: 1521
● SID: xe
○ Nome servizio:

☐ Autenticazione sistema operativo ☐ Autenticazione Kerberos ☐ Connessione proxy

Stato:

5. Check DB availability or import data in your database

To continue the practice, you need the data warehouse tables on your database. The data are already available in the DB you connect. You can check the table availability in the "Tables" section of Oracle SQL Developer.
In the case the tables are not available, you have to import them. The material is available on the course website. After downloading the material, you can import the comma separated values (csv) files into the database. You can follow the tutorial available in the course website.
Pay attention to the data types associated with each column during the import phase. In particular, check that the date field of the time is interpreted as Date.

6. Execute the queries by means of the interface.