Data warehouse in Oracle – Practice 1

🗄 Data	https://dbdmg.polito.it/dbdmg_web/wp-content/uploads/2021/10/DW_CELLS_sql.zip

Data Management and Visualization - *Politecnico di Torino*

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 analysing 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 analyse 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**.

Table 1 – Source data base with single call information

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
	
	~1,300,000 rows

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 analyses 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.

> ⚠️ **Warning:** Table 2 is the logical schema of the Data Warehouse. Try to design the conceptual schema *before* looking at it.

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

Table 2 – Proposed solution - Data warehouse tables

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

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 logical 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

You can find the tables corresponding to the schema in **Table 2** in an SQL script provided as supplementary material. Use these tables as source for the following queries. The suggested tool for these Labs is https://livesql.oracle.com/. A short guide on how to import them and query the DB is available (**pdf**).

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.