# Milestone 2

Mateusz Pawlik

December 5, 2011

## Exercises 5 - Aggregation Queries in Data Warehouses - (07.11.2011, 14:00-16:00)

### Warming-up Exercises

For the exercises below use the data warehouse that You designed.

1. Write two SQL queries. In the first query use ROLLUP extension to GROUP BY. In the second query use CUBE extension. Describe the difference between ROLLUP and CUBE extensions.
2. How GROUPING SETS expression differs from ROLLUP and CUBE expressions? Write an SQL query using GROUPING SETS expression.
3. What are the composite columns? Rewrite the ROLLUP SQL query from ex. 1 using composite columns. Describe the differences between those two queries.
4. What is the purpose of grouping functions? Write SQL queries using each of the following grouping functions: GROUPING, GROUPING_ID, GROUP_ID. Describe their usage.

### Milestone 2 Tasks

1. For the designed data warehouse write 20 aggregation queries in both natural language and SQL. Include the results (if the resulting data amount is big, include a sample).

## Exercises 6 - Analysis and Reporting Queries - (14.11.2011, 14:00-16:00)

As a reference use `http://download.oracle.com/docs/cd/B19306_01/server.102/b14223/aggreg.htm#i1007241` and `http://download.oracle.com/docs/cd/B19306_01/server.102/b14223/analysis.htm#i1006709`

## Warming-up Exercises

1. Connect to the sample database. Log in to `russel.inf.unibz.it`. Then access the database by

   ```
   rlwrap sqlplus dwdm/dummy@ alcor.inf.unibz.it:1521/orcl
   ```

2. Show Internet sales for every country and year. Show a custom description whenever a subtotal value is displayed, e.g. for all countries subtotal show 'Multi-country-sum'.
3. Show the ranking of channels over sales amount in Japan, USA and Great Britain for every year with groups subtotals.
4. Divide Internet sales amount per month in year 2000 into 4 tiles.
5. Show centered 5 days moving average of all sales during November 2000.
6. For each country find the channel resulting with the maximal sales.

## Milestone 2 Tasks

For Your data warehouse write the following queries in natural language and SQL, include the results. If some of the queries, that You prepared for the previous Milestone 2 tasks overlap with those below, count them as done.

1. 2 ranking queries.
2. 2 windowing queries.
3. 2 period-to-period comparison queries (queries comparing values across time periods, e.g. compare sales for every week of the current year with the sales of the corresponding weeks in the past year).
4. 2 dense reports (i.e. with rows for all combinations of dimension values displayed even when no fact data exist for them).

## Exercises 7 - Materialized views (21.11.2011, 14:00-16:00)

As a reference use `http://www.inf.unibz.it/dis/teaching/DWDM/slides2011/dw6_HRU_sigmod96.pdf`.

## Milestone 2 Tasks

1. For the main fact from Your data warehouse choose three most frequent queries. Measure their performance. Describe their execution plans in the DBMS.
2. Draw a multidimensional lattice of the chosen fact. Mark the group-by sets needed to answer the chosen queries.

3. Choose the candidate views, that are actually useful for answering the chosen queries.

4. From the candidate views choose a subset which in Your opinion is the most suitable. Argue the choice. Describe the overhead of creating those views.

5. Implement the materialized views of the chosen candidates.

6. Rewrite the chosen three queries to use the materialized views.

7. Measure the performance of the rewritten queries and compare it to the performance of the starting queries. Discuss their execution plans.

## Exercises 8 - Indexing in data warehouse (05.12.2011, 14:00-16:00)

## Milestone 2 Tasks

1. Check what types of indexes does Your DBMS support. Name and describe those of them which are especially useful for speeding up Your queries.

2. For the chosen queries and their versions which use the materialized views, implement indexes to improve their performance (You can try to evaluate several strategies and choose the best one, e.g. best speed/size ratio). Argue Your choice. Describe the overhead (index size, computation time, index update costs, etc.) of implementing the chosen indexes.

3. Measure the queries performance using indexes and compare it to the performance without indexes. Discuss the execution plans.