Visualization of Relational Database with Relational Online Analytical Processing Approach

Gemeng Qin, Álvaro Peiró Burriel, David Escobedo Ortiz, and Shuang Xiao
School of Computer Science and Communication (CSC), Royal Institute of Technology KTH, Stockholm, Sweden

Abstract. This paper presents a database application developed to visually represent randomly generated data about the people’s working details, education information and living conditions, etc. This data is stored in a database. PostgreSQL is used as DBMS to manage data and a Java database application is developed to do data manipulation. The idea is to be able to make queries to that database via the user interface, and retrieving the information represented in the forms of chart or graph.

Keywords: Database Application, ROLAP, Visual Representation

1 Introduction

OnLine Analytical Processing (OLAP) is a supportive approach for knowledge worker (executive, manager, analyst) to make effective decisions. An OLAP application is a business intelligence to use to retrieve and present the queried result effectively.

One of OLAP’s subcategory is Relational OnLine Analytical Processing (ROLAP), which is specifically associated with relational database.

1.1 ROLAP Schema

Star Schema Star schema is the key schema for ROLAP. The database with star schema consists of a single fact table and a single table for each dimension. Each tuple in the fact table consists of a pointer (foreign key - often uses a generated key for efficiency) to each of the dimensions that provide its multidimensional coordinates, and stores the numeric measures for those coordinates. [1] The schema is shown as Figure 1.

1.2 Advantages of Using ROLAP Approach

- ROLAP approach can handle large amounts of data. The limitation of the size of a dataset is the limitation of the database only. In other words, ROLAP doesn’t have limitation for dataset size.
- It can leverage functionalities inherent in the relational database. Often, relational database already comes with a host of functionalities. ROLAP technologies, since they sit on top of the relational database, can therefore leverage these functionalities. [2]

2 Implementation

2.1 Database Design

Schema Design As inspired by the website of Statistics Sweden, we suppose to have a relational database based on population social studies. We select 4
distinct stages that one may experience in their life as our basic data model: Living condition, education and training, employment, and receiving pension. For each entity-relation, we do concern about how to present the query result into a visualize form. So the attribute we select for the database is on the main feature of the entities or relations. In Figure 2, this is our final schema graph. When combined the expected RO-LAP theory and the schema we want to have, we draw a schema that does not totally obey the rules of star schema design. In each branch of person, the relation could be consider as the fact table of star schema and the entity connect to this relation is the dimension table. The fact table here only connects to 2 dimensional tables.

Data Generation In this part of work, we created our table and inserted the data into PostgreSQL database. In order to generate data which would match the practical issues, we did it in two steps: first generate the data from data generator named Datamatic then fix the dataset by programming.
a) In the data generator, we built up our original data either randomly or sequentially according to different types of attribute. We would have our database model split into two parts, dataEntities and dataRelations, to make the foreign keys available in our created tables with the requirement of the data generator, and data are stored into sql file so that it is more general to use in the future.

b) After having the original dataset, we used Java to fix our data into a reliable social data model. Then we connected Java program with the database to create the tables and insert the SQL file into the PostgreSQL.

2.2 Visualization

**JFreeChart** For the graphical part we thought since the beginning about programming it on Java, due to our team is familiar of doing queries and processing the results of an SQL database in Java with JDBC. Furthermore, it’s easier to develop a graphical interface providing the queried results represented in charts (among some predefined queries) with Java Swing. Therefore, after researching, we narrowed down the search to only 2 libraries: charts4j and JFreeChart.

We decided to use JFreeChart because charts4j was incapable of managing large data representations. It uses a Google Service to obtain the charts so the application will fail to work without internet connection. Furthermore, it was assured that JFreeChart was compatible with PostgreSQL, which provided us connivence and strong Internet community support.

**Pentaho** We also considered to use an open source ROLAP tool, Pentaho. This tool provides friendly user interface and many practical features for OLAP design. However, because of the limitation of our database design, the database we developed doesn’t cope well with Pentaho.

In the end, we show the charts using the JFreeChart library.

3 Output of Test Run

The test results are shown as figure 3, figure 4, figure 5 and figure 6.

![Fig. 3. Employees’ Average Ages in Different Companies(Bar Chart)](image1)
![Fig. 4. Employees’ Average Ages in Different Companies(Pie Chart)](image2)

4 Further Improvement

**Star Schema Design** In the database design part, our schema can be considered as a star schema when break it down into several parts. If we want to use ROLAP tools to do our application, we expect to contain more business issues around one fact table while cutting down other branches. In another way, we might not to choose ROLAP tools to complete our visualization part.
Data Generation  The data we used were almost randomly generated with several constraints and triggers we developed now. We hope that for each person the relation could be more like a real human life. To achieve that, we are required to design more thoroughly than current stage when developing constraints.

5  Conclusion

The database application we developed aims to visualize the queried results of a predefined database in order to provide support for effective data analysing and decision making. The application works well with a nice user interface. However, there are still several points can be further improved when considering database design based on star schema and simulating data based on real life situation.

References

3. charts4j - Let the computer in the cloud build your charts, https://code.google.com/p/charts4j/
4. JFreeChart, http://www.jfree.org/jfreechart/