Star Schema Advantages on Data Warehouse: Using Bitmap Index and Partitioned Fact Tables

Emany Sidi
Laboratory modeling and information theory Abdelmalek Essaadi University, Tétouan, Morocco

Mohamed El Merouani
Laboratory modeling and information theory Abdelmalek Essaadi University, Tétouan, Morocco

El Amin A. Abdelouarit
Laboratory modeling and information theory Abdelmalek Essaadi University, Tétouan, Morocco

ABSTRACT
The data warehouse designer should consider it effectiveness while the design process, this might be a part of it work by analyzing the update frequency of production databases.

Actually, to decide in a small time interval becomes the most important concern for deciders, this because it's always depending on data warehouse refresh using the ETLs and the cube generate process based on data warehouse schema type.

To present the KPI's to the management quickly, we need to minimize the query execution time.

This papers shows that the star schema is more advantageous when using a bitmap index based on ETL query execution time and data access time.

General Terms
Business Intelligence, Data Warehouse.

Keywords
Data Warehouse, DBMS, Indexes, Business Intelligence

1. INTRODUCTION
The data warehouse performance is considered the main concern for designers. This performance is based in first on the query execution time and the data access complexity.

For that, physical design is considered as the most important task, including how to improve access to this data.

To minimize the query execution time, it depends on data warehouse design type (how many joins in tables). For that we will work on the problem of choosing the type of schema in the data warehouse design phase.

There is many types of data warehouse schemas. We have chosen two type of schemas to study: star schema and snowflake schema, these two kind of schema are made for ROLAP systems.

2. ROLAP SYSTEMS
It uses a relational representation of a data cube, constituted from fact and dimension tables. The fact table contains in their attribute values of activity results and a foreign key to each dimension. The ROLAP has as advantage the use of existing databases which reduces it implementation cost [7] [9].

This study compares between two schemas: Star and Snowflake schema.

3. STAR SCHEMA
In this model, each group of dimensions are placed in a dimension table, the facts are placed in fact table. The result is a star schema where the fact table are in center rounded by the dimensions tables [10], the dimension tables contains qualitative data represented in a big number of attributes. These qualitative data supports many analyze processes. In other side, the fact table has an important number on instances, each tuple in fact table has two kind of attributes:

- Foreign keys referencing to the dimension table
- A set of measures that can be aggregated to perform treatments.

The fact table is generally normalized, but the dimension ones are not; the queries that are used in these schema are called "Star join query".

The figure 1 shows an example of a star schema.

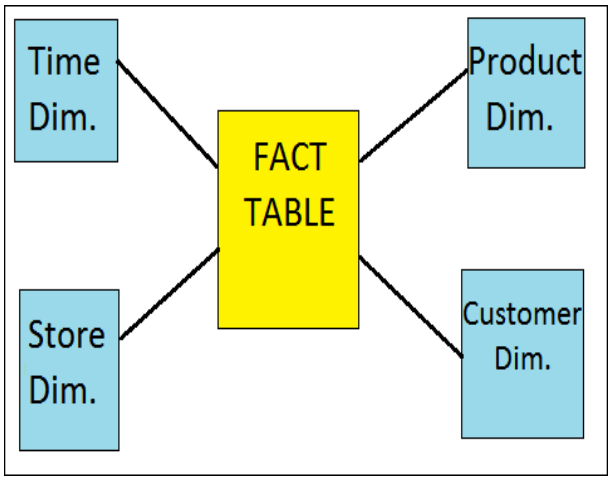

Fig 1: Example of Star Schema

4. SNOWFLAKE SCHEMA
The snowflake schema reflects the hierarchies associated with each dimension.

Each dimension table is split into a plurality of hierarchies. This diagram normalized dimensions, reducing the size of each of the connections, thus allowing to formalize the concept of hierarchy within a dimension [7] [8].

Tables representing the finest hierarchy are directly linked to the fact table. The tables representing other hierarchies are linked to each other according to their level in the hierarchy.

Figure 2 shows a snowflake schema.

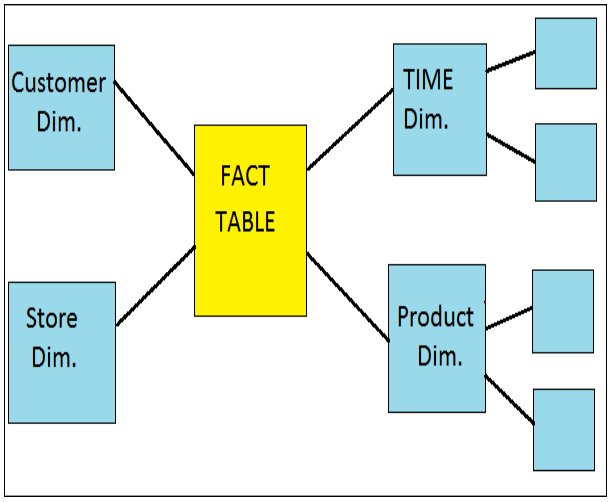

Fig 1: Example of Snowflake Schema

5. HYPOTHESIS

The bitmap index permits to a fast access to data [1][2][3][4][6], but the star schema is not efficient when the fact table contains a large data [5]: Less number of foreign keys and hence shorter query execution time (faster), to make it real, we look forward the use of partitioned tables, this will reduces the data volume in table: data will be distributed in many tables, so it conduces to less foreign key.

To do that, we have to divide the fact table in many partitions based on a dimension, and would be easier to take the time dimension to partition it (in months, years, decade…)

So I demonstrate that by using bitmap index and filtering and partitioned tables in a star schema is more efficient than using the snowflake one based on star join query optimization.

Note that we take two similar environment with SQL Server as RDBMS (Relational Data Base Management System).

6. ANALYSIS AND RESULTS

To satisfy this study we created the following Data warehouse sample Schemas:

- Star schema: 1 fact table 3 dimension
- Snowflake schema: 1 fact table 3 dimension and 2 sub dimensions under dimension d1.
- 1 million row inserted in both fact tables with same data (by respecting it dimensions)

To get a deep analyze we will use a query for both data warehouse schema: star join query (Table 2) and snowflake join query (Table 3).

6.1 Technical specification of lab environment

This study has been executed in similar technical environment that we present in the following table: (Table 1)

Table 1. Technical specification of lab environment

Physical Memory	Storage Disk	Operation System	RDBMS
8 Gb	1 TB	Win 2008 Server	MS SQL SERVER 2008 R2

The huge resources that we offer has for objective to make difference between old decision systems with lower resources and the new decision systems with high technology, high resources.

Table 2. Star join query used

```
SELECT d1.att2, d3.att2, d1.att1, SUM(f.measure1)
from Fact1 f inner join dim1 d1 on d1.dimkey1 = f.dimkey1
GROUP BY d1.att1
```

Table 3. Snowflake join query used

```
SELECT fact2.fact2_att1, fact2.fact2_att2,fact2.fact2_att3,
d11.d11_att1, fact2.fact2_att3, fact2.fact2.att3
FROM d11 LEFT JOIN d1
ON d11.d11_att1 = d1.d11_att1 LEFT JOIN fact2 ON d1.d11_att1 = fact2.fact2_att1
GROUP BY fact2.fact2_att1;
```

6.2 Used queries

These two queries are getting the same data in different way.

The objective is to stress these 2 data warehouses scheme before and after applying the bitmap index and partitioning table to star schemas, this application will be called in this work as optimization tasks.

The comparison will be based on execution time and memory occupation. We focus on these two factors as per the execution time depends on memory consumption.

6.3 Results before applying the optimization tasks

Before applying these predictions we will test the both queries on a standard star and snowflake schema we get results in the following table.

Query	Memory Occupied by process	Real time Query execution in seconds	DW size
Star Join	72%	21s	0.78 GB
Snowflake Join	49%	17s	0.61 GB

After analyzing this results, we may say that:

- Snowflake schema is better when dimension table is relatively big in size, because it reduces space.
- Query execution is slower when using star schema in a large data case.
- As per data redundancy the star join occupies more memory than the snowflake.

6.4 Results After Applying The Optimization Tasks

After partitioning table in star schema per time dimension d1 (per year) and added bitmap index, we got the following results:

After applying modification we found that in all factors, the star schema is recommended to use.

Query	Memory Occupied by process	Real time Query execution in seconds	DW size
Star Join	39%	16s	0.75 GB
Snowflake Join	48%	19s	0.61 GB

6.5 Results analysis

Based on this results, the factors of memory occupation, time execution and data warehouse size become efficient after partitioning tables and adding bitmap index because of:

- Less data when partitioning per table
- Bitmap index efficiency when the three is large data and less distinct values.

So, partitioning tables and using bitmap index is a tuning task that can qualify better the use of star schema on data warehouses.

7. CONCLUSION AND FUTURE WORKS

The bitmap index and partitioning the fact table per time dimension did an important role for the data warehouse performance optimization, the use of the star schema in the data warehouse has as objective is to conserve it advantages like lower query complexity and easy to understand, less number of foreign keys and hence shorter query execution time. Actually, the majority of data warehouses that are used had the objective to analyze the organization production, commercial, finance activity. This means that in the most of cases data warehouses are more efficient if designed as data mart per activity. But the work that we present is to prove that star schema can be used even if there is large data, and the bitmap index and partitioning fact tables are playing the role to create "data marts" time oriented.

As future work, we will discuss what a is given by this study on decision dashboard and business reporting, as they are considered the main window from where management can have a transparent look of what is going on their organization.

8. REFERENCES

[1] E. Abdelouarit, M. El Merouani, A. Medouri, The bitmap index advantages on the data warehouses American Academic & Scholarly Research Journal Vol. 6, No. 4, July 2014

[2] E. Abdelouarit, M. El Merouani, A. Medouri, Data Warehouse Tuning: The Supremacy of Bitmap Index International Journal of Computer Applications (0975 – 8887) Volume 79 – No7, October 2013.

[3] E. Abdelouarit, M. El Merouani, A. Medouri, The impact of indexes on data warehouse performance IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 5, No 2, September 2013.

[4] E. Abdelouarit, M. El Merouani, A. Medouri, OPTIMISATION DES PERFORMANCES DES ENTREPÔTS Rev. Ivoir. Sci. Technol., 20 (2012) 35 - 67 ISSN 1813-3290, http://www.revist.ci

[5] S. Chaudhuri, U. Dayal,An Overview of Data Warehousing and OLAP Technology., ACM SIGMOD RECORD. 1997

[6] E. E-O'Neil and P. P-O'Neil, Bitmap index design choices and their performance impli-cations, Database Engineering and Applications Symposium. IDEAS 2007. 11th International, pp. 72-84.

[7] R. Kimball, L. Reeves, M. Ross, The Data Warehouse Toolkit. John Wiley Sons, NEW YORK, 2nd edition, 2002.

[8] W. Inmon, Building the Data Warehouse., John Wiley Sons, fourth edition, 2005.

[9] C. DELLAQUILA and E. LEFONS and F. TANGORRA, Design and Implementation of a National Data Warehouse. Proceedings of the 5th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Madrid, Spain, February 15-17, 2006 pp. 342-347.

[10] R. Kimball and K. Strehlo. Why decision support fails and how to fix it. SIGMOD Record, 24(3) :92–97, September 1995.