Data Warehouse Design for Croatian Students' Nourishment Information System

Nenad Milanovic, Goran Soskic, Ana Petkovic
University Computing Centre, J. Marohnica 5, Zagreb, Croatia
nenad.milanovic@srce.hr, goran.soskic@srce.hr, ana.petkovic@srce.hr

Abstract. There are two basic design models that can be used for building a data warehouse: Bottom up and Top down model. Choice between these models depends on method of data organization in transactional database and faster or cheaper database warehouse development. Depending on choice, solution has to be purposeful about data usage and later analytical operations which will be used on warehouse data. The main goal remains the same – fast and simple data access.

This paper presents a strategy and development of Warehouse model for Students' Nourishment Information System in Croatia.

Keywords. Data warehouse, data mart, reporting model, data analysis, data transformation, dimensions

1. Introduction

Information system basically consists of very large amount of data. There is a need for analyzing that data, and therefore data gets extracted, transformed and loaded into the data warehouse. Data collection is not the main purpose of data warehouse, as the data is collected to help in a later decision making and data analysis. Data warehouse is necessary part of almost every larger information system, because there is a common need for some sort of analytic services and fast viewing of prearranged data.

There is no doubt that transactional database is foundation of any information system, but at some point there is going to be a demand for building a data warehouse. It is much easier to examine and analyze data when it is stored in a warehouse, because it is filtrated, transformed and aggregated before loading into the warehouse, which provides faster data browsing.

In this paper, we are going to demonstrate the way of building and organizing data warehouse for our system, the Students’ Nourishment Information System (ISSP [7]), which has actively been used for more than 10 years. This information system has over 60 million records, more than hundred points of data capturing and relatively small number of data viewers. It’s a card system that collects students’ data for subsidized student food. From that data, depending on certain criteria, and for students who qualify for subsidized student food, certain amount of subsidy is given and can be spent using student cards (X-card). Students are paying only ¼ of the price and the the rest of the price is subsidized. Students’ restaurants, in their offer, also have unsubsidized items for which students have to pay the full price.

2. Present model of ISSP

ISSP is built as relational database on Microsoft SQL server RDBS.

The owner of the data stored in ISSP database is Ministry of Science, Education and Sports (MSES [6]).

Two main ISSP database users are student restaurants and student services at the universities. Every day, in a predefined time frame, they exchange data with ISSP database over the Internet. Student services send us information about their students so we can calculate amounts of subsidy the student should get at the beginning of every month. And from the student restaurants we collect information about their sales, subsidized and unsubsidized items, spent subsidy, etc., and we send them information about students and their subsidies. Time and other relevant data are saved for every data transaction. Exchange security is achieved by unique one-time, short-living token which is used to provide two-way authorization.

Our third important user is the “data owner”, MSES; who can browse data and create reports that are going to help them make decisions.

Due to large amount of data in transactional database and frequent requirements for specific summarized data, it was necessary to build the data warehouse.
3. Data Warehouse Design

3.1. Data Warehouse

The widely accepted definition of a data warehouse is “a subject-oriented, integrated, time variant and nonvolatile collection of data used in strategic decision making” (Bill Inmon 1980’s) [1]. The data warehouse acts as the central point of data integration.

Data warehouse delivers a usual view of enterprise data. Since these are data for the business consumers, it supports the flexibility in later data analysis. The data warehouse is a stable source of historical information that is constant, consistent, and reliable for any consumer. Because the data warehouse can grow to huge proportions its design is set up from the beginning to accommodate the growth of this information in the most efficient manner using the enterprise’s business rules for use throughout the enterprise.

3.2. Data Marts

Data marts are subsets of data stored in warehouse. That is where most of the analytical activities in the Business Intelligence (BI) environment are taking place. The data in each data mart are usually created for a particular function. They may be physically collocated with the data warehouse or on their own separate platform, and they vary in size from a few megabytes over multiple gigabytes to even terabytes.

3.3. Data Warehouse Design Approach

3.3.1. Bottom-up design

Bottom-up data warehouse design is promoted by Ralph Kimball [9].

In the so-called bottom-up approach data marts are were, in the beginning, created to provide reporting and analytical capabilities for specific business processes. Data marts contain atomic data and, if necessary, summarized data. These data marts can eventually be joined together to create a comprehensive data warehouse (Fig. 1.). The combination of data marts is managed through the implementation of what Kimball calls “data warehouse bus architecture” [8].

3.3.2. Top-Down Design

Bill Inmon, one of the leading promoters of the top-down approach in which data warehouse is designed using a normalized enterprise data model, defined a data warehouse as a centralized repository for the entire enterprise [3] (Fig. 2.). In data warehouses "Atomic" data (data at lowest level of detail) are stored. Data marts are created from the data warehouse and contain data for specific business processes or specific departments.

Some of Inmon’s states about data warehouse:

**Subject-oriented**
- Data in data warehouse is organized so that all the data elements relating to the same real-world event or object are linked together.

**Time-variant**
- Changes to the data in data warehouse are tracked and recorded so that reports can be produced showing changes over the time.

**Non-volatile**
- Data in data warehouse is never over-written or deleted - once committed, the data is static, read-only, and retained for future reporting.
Integrated

- Data warehouse contains data from most or all of an organization's operational systems and this data are made consistent.

The top-down design methodology generates highly consistent dimensional views of data across data marts since all data marts are loaded from the centralized repository. Top-down design has also proven to be robust against business changes. Generating new dimensional data marts against the data stored in the data warehouse is a relatively simple task. The main disadvantage about the top-down methodology is that it represents a very large project with a very broad scope. The up-front cost for implementing a data warehouse using the top-down methodology is significant, and the duration of time from the start of project to the point that end users experience initial benefits can be substantial. In addition, the top-down methodology can be inflexible and unresponsive to changing departmental needs during the implementation phases [2].

4. Implementation

The purpose of data warehouse is to extract relevant information out of large quantity of data continuously uploaded into the ISSP central server. Those extracted data will provide the best insight to system’s state and will assist to the process of decision making.

Besides basic data about food consumption that are relevant both to restaurant owners and to the MSES, it is possible to get data about subsidy flows and distribution among geographical regions, percentage of used subsidy and many other interesting data.

By analyzing given data it is possible to determine both positive and negative trends inside entire system which enables decision makers to make timely decisions in order to improve overall system quality by correcting negative trends that might point to system errors, conceptual flaws or even system abuse.

This warehouse is used by both the MSES and the student restaurants, but they review different data. While restaurants are interested in having exact sales income and quantity of items sold, MSES is interested in awarded and spent subsidies.

Due to different purpose of the data that is intended for student restaurants and MSES, we have decided to implement a data warehouse by using Ralph Kimball’s bottom-up approach (Fig. 3.). The fact that bottom-up approach is faster and cheaper method was also important reason to choose this approach. Given that the need, for both data marts, was not in the same time, we decided to build our warehouse starting from data marts.

That effectively means construction of two data marts both focusing on food consumption and subsidy usage, but with different data granularity. First data mart was created for the MSES.

Although MSES’s main point of interest is amount of subsidy that is spent in each restaurant, they are also interested in information about students in order to be able to track food consumption and subsidy usage and to plan future system enhancements accordingly. Therefore, it is not possible to use aggregated data for each restaurant. Instead, it is necessary to have information about each bill issued to a student, which effectively determines data granularity for the first data mart.
The whole data warehouse design process, including reporting model, is divided into three main parts:

- a) Defining and creating data marts
- b) Loading processed data into data marts - ETL
- c) Reporting model design

4.1. Creating logical and physical design of the data marts

Implementation process begins by creating logical design for each data mart. Fact tables contain data which will be measured. In our case that data is food consumption and amount of spent subsidy. Similarly, dimension tables are created containing attributes by which data in the fact table will be evaluated [5].

4.1.1. Restaurant data mart

The fact table is linked to the dimension tables which make the star-schema which is quite simple to implement (Fig. 4.).

![Figure 4. Star schema](image)

Measures used in the fact table are quantity of sold items, unit price and total subsidy amount of an item. The rest of the elements in the fact table make a compound primary key and they are as follows: sales date, restaurant id, cash desk id, item id and a bill number. Elements of the primary key correspond to primary keys in dimension tables.

Subsidized amount is calculated as a fixed percentage of total sales amounts. However, it is not enough to store only quantity and unit price of an item and then calculate subsidy, as one would assume, because there is a restriction in the system concerning subsidy. There is a daily maximum subsidized amount limit which student is allowed to spend, and if that limit is reached, subsidy amount entered in the database ensures that the limit is not overdrawn. This is why it is necessary to have subsidized amount in every entry.

Dimension tables are denormalised in accordance to the selected dimensional model for increased querying performance.

Besides standard time dimension with day – week – month – quarter – year hierarchy, there is a product (item) dimension table containing attributes such as product description, product group and others. If an item should be changed in any way, modified version of that item is added as a new item into the system. Third dimension table contains attributes relevant to restaurants, such as restaurant name and address. Change of street address or a name of a restaurant is not relevant to data users, so that data is not kept in the data mart. Exception is made in the case if restaurant relocated to another town. In that case the restaurant is added as a completely new restaurant into the system.

In conclusion, all dimensions of the model have slowly changing data without a need to maintain historical data as explained earlier.

4.1.2. MSES data mart

Dimensional model is similar to the previous data mart. The main difference is in the granularity of data stored in main fact table which contains information about each bill that is issued to a student without detailed food consumption data. However, if more detailed data is needed, MSES has access to restaurants data mart and can get data from there.

Besides data about bill and subsidy amount, the fact table contains additional data about to whom the bill was issued, who issued it and whether the card number was manually entered or not.

Number of manually entered card numbers makes up about 2% of total bill issued and even though one might consider making a separate table containing those bills, that would be a wrong approach for both memory usage and querying performance as well as for design schema.

Time dimension table and restaurants dimension table are the same as described in previous data mart. Additional dimensional tables are cashiers dimension table and students dimension table.

Students dimension table, besides standard data, contains data about university that student attends and corresponding geographical data. This dimensional table has large slowly changing data but with a need to maintain data history. For
that reason, two additional attributes are created containing effective dates (null end date signifies current tuple version).

Since in ISSP subsidy is awarded on monthly basis and it is not cumulative but system owner is interested in total awarded and used subsidy amount, data mart contains additional fact table with awarded and used subsidy amount for each student and month. This table shares time and students dimension tables with the first fact table.

4.2. Extract, transform and load

After creating data marts, our next step is to load some test data and to test the model. If testing is successful, real data is loaded into data marts.

Since all data is stored in one place, meaning the system is homogeneous, the whole process is relatively simple (Fig. 5).

Due to the constant workload of operational database and in order to downgrade performance as little as possible, relevant data is extracted without doing any additional operations on data tables. Data is loaded into staging area where further necessary operations will be performed. First of all, data is checked for discrepancies and incompleteness and then corrected if needed.

After data has been checked, it is transformed and shaped to fit into data marts. In our case, most transformations are referred to joining and denormalizing data tables, while transformation of data values and types is not needed because the source of data is homogeneous. The rest of the transformations refer to aggregating monthly used subsidy for each student.

Checked and transformed data is then loaded into data marts [4].

Figure 5. Data load

The whole process of extracting, transforming and loading data into data marts is automated using MS SQL Server Integration Services 2005.

During data load it is necessary to maintain order in which data is loaded to preserve referential integrity. With that in mind and since there is a large quantity of data during initial load, the data is loaded in predetermined order: first dimensional data and then fact table data for both data marts in monthly steps.

After the initial loading, new data is loaded daily during night hours when the workload of database server is minimal.

4.3. Reporting model

Upon creation of the data marts, data warehouse is ready to be used.

Users can create their own queries or use predefined ones (usually predefined reports) to get the requested data. However, creating queries demands certain knowledge about both SQL and MDX (Multidimensional Expressions), that targeted users usually do not posses. Learning about databases and SQL for the sole purpose of querying a data warehouse, would be very tiresome and counterproductive, and users usually do not want to be bothered with that.

MSES is interested only in analyzed data to make its business decisions and not in technical details of the system and used technology. Therefore, a reporting system is created which contains predefined reports with most frequently requested data. Reporting model is created using MS SQL Reporting Services 2005 with reports available over the Web.

Some of the most frequently requested reports are:

- Awarded / used subsidy by student rights level
- Awarded / used subsidy by student rights level and university
- Awarded / used subsidy by student rights level and city / region
- Food consumption (spending) by restaurants
- Unsubsidized items by restaurants
- Sold products by restaurants / product groups
- Bills with manually entered card numbers by restaurants / cashiers

All of these reports take two input parameters, starting date and ending date, which are entered by users prior to report creation.
5. Conclusions and future work

After building and using data warehouse, stored data becomes more “readable”. The cost of maintenance can be reduced by recognizing certain anomalies which can appear in process of making the data warehouse. Furthermore, large future financial savings can be made with “what-if” solutions based on past data.

With our ISSP warehouse we managed to decrease working hours of our helpdesk and still keep our users satisfied, as now they can see their data fully transparent and over larger period of time. MSES personnel, which is mainly interested in funds invested in students, can access relevant data at anytime and from anywhere over the Internet. Even at the peak hours when the transactional database is overloaded with other queries, warehouse data analyzing is possible.

Final step is creating presentation layer for data warehouse. As the presentation layer is restricted by warehouse possibilities, decisions can only be adequate if information system collects and presents data in right way for particular user.

6. References

[1] Data Warehouse
http://en.wikipedia.org/wiki/
Data_warehouse [02/16/2009]
[2] Data Warehouse Design Considerations
http://msdn.microsoft.com/
en-us/library/aa902672(SQL.80).aspx
[02/16/2009]
[4] Extract, transform, load
[5] Mastering data warehouse design (relational and dimensional techniques) – Claudia Imhoff; Nicholas Galemmo; Jonathan G. Geiger
http://public.mzos.hr [20/03/2009]
[7] Students' Nourishment Information System
http://www.cap.srce.hr [02/16/2009]
[9] "The Bottom-Up Misnomer".
http://www.intelligententerprise.com/03091
17/615warehouse1_1.jhtml [2009-03-20]