DATA WAREHOUSE MANAGEMENT SYSTEM– A CASE STUDY

DARKO KRULJ
Trizon Group, Belgrade, Serbia and Montenegro.

MILUTIN CUPIC
MILAN MARTIC
MILIJA SUKNOVIC
Faculty of Organizational Science, University of Belgrade, Serbia and Montenegro

Abstract
In the last decade data warehouses have become the basis for support in business decision making. This paper will show our framework for the data warehouse management system as well as advantages and disadvantages of this approach in relation to the other data warehouse management systems. Many approaches to the development of data warehouse emphasize the importance of meta data yet none of them sticks to that principle. The basic characteristic of our approach is that it is fully based on XML meta data. We have also implemented significant changes in dimensional model based upon flexible dimensional hierarchies which enabled us to use special methods of creating aggregations and important advantages in data visualization. What makes this approach different is the concept of intelligent creation of aggregates based on observation of history of users queries and the use of modified data mining algorithm APRIORI.

Keywords*: data warehouse, meta data, dimensional modeling, data mining*

1. INTRODUCTION

Occurrence of the very large databases and diverting the focus of attention from collecting data to data processing caused development of data warehouse (DW). According to [1] DW is defined as “overall consistence warehouse, got from different data sources, accessible to the final users, and in comprehensive way usable for work”.

Reaching consistence and overall of data in information systems is very complex task, especially when it is comes to big companies information systems. DW can be conceived as a possibility to transcend the problems of integrations of different information systems.

In [6] Kimball gave general definition of DW as “copy of transactional data specially designed for queries and analyze”.

In [5] Inmon defined DW as “subject oriented, integrated, stable and non volatile collection of data special organized as support to businesses requests”. According to this definition it can be concluded that the DW mainly represent process then specific kind of data base. DW represent technique which provides correct collecting and managing large amount of data from different sources, in the purpose of finding right answer for business tasks and support in making decisions which cannot be obtained from operational databases.

Based on our experience in realization of dozens of projects of design and developing of DW we have developed our own framework, database management system for DW realization and appropriate tools for data analysis and data warehouse modeling. In this paper we will present advantages of our framework and software tools compared to Microsoft method.

2. FRAMEWORK

Our framework for DW realization is shown on Figure 1.

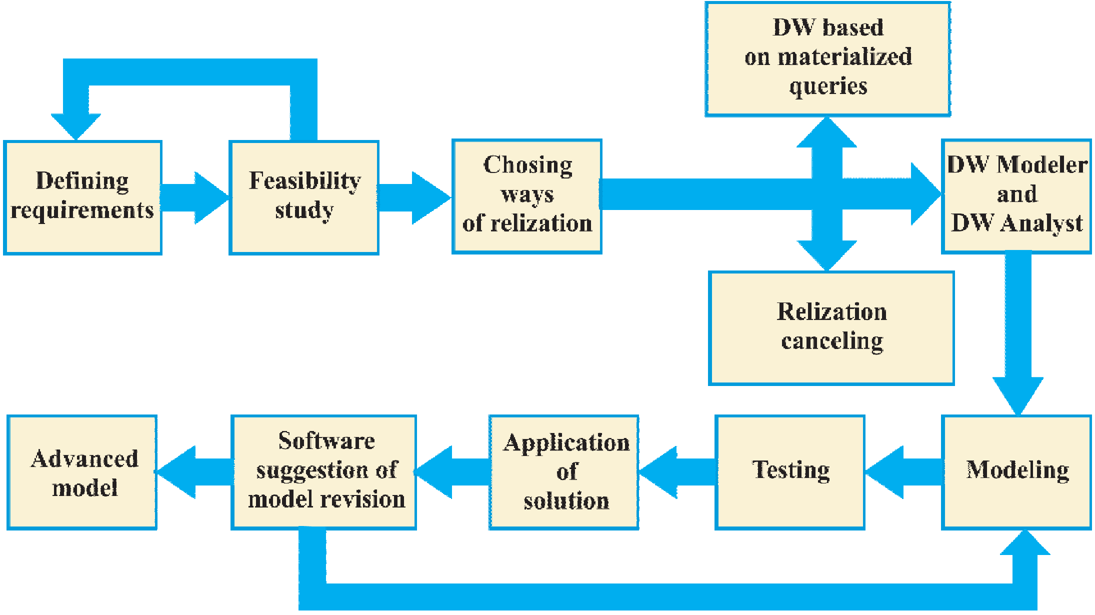

DW based on materialized queries
Defining requirements
Feasibility study
Chosing ways of relization
DW Modeler and DW Analyst
Relization canceling
Advanced model
Software suggestion of model revision
Application of solution
Testing
Modeling

Figure 1 Framework for DW realization

The first stage of developing DW is defining requirements. In this stage the aim is to collect, as much as possible, information about the DW that has to be realized. Interviewing of all stake holders interested in development of DW (analysts, managers, IT departments) is of large importance. Next stage is feasibility study, in which the collected data from previous stage have to be analyzed for evaluation of the possibility to realize the DW. Often there is a need go back to the previous stage for solving some inconsistencies or to collect some more information about the realization. After collection of necessary information about the DW development project, there has to be found a suitable way of realization. If the evaluation of DW shows that it can be realized successfully, the possible ways of doing it could be: realization though DW based on materialized queries which we showed in [8], or development based on DW modeler and DW analyst represented in [10]. If the realization goes through DW modeler tools, next step is modeling of DW. After creation models have to be tested, and if they are valid, the application can be delivered to end users after the training. The basic difference between the suggested model above and the most frequently used frameworks presented in [12] is the stage in which the software revision of model is performed. There has to be pointed out that many implemented models, beside the basic purpose, can be significantly improved. In this stage, on the basis of historical users queries software for managing DW gives the suggestion for model improvement. Actual results of using improving models are shown in Table 2. The main criteria, on the basis of which the suggestions are made is frequency of dimension and dimension level usage. The experience has shown the existence of some dimensions and levels that are rarely in use.

As we can see on Figure 2, our solution and Microsoft solution are the same in the first three steps: choosing of data sources, process of extraction, transformation, loading and creating of staging base. In the next few phase we will see the main difference: Microsoft solution are using Analysis server for modeling DW and MS Excel or MS Data Analyst for presenting data to clients, while our solution is using DW modeler application for modeling DW and DW analyst application for presentation multidimensional (MD) data.

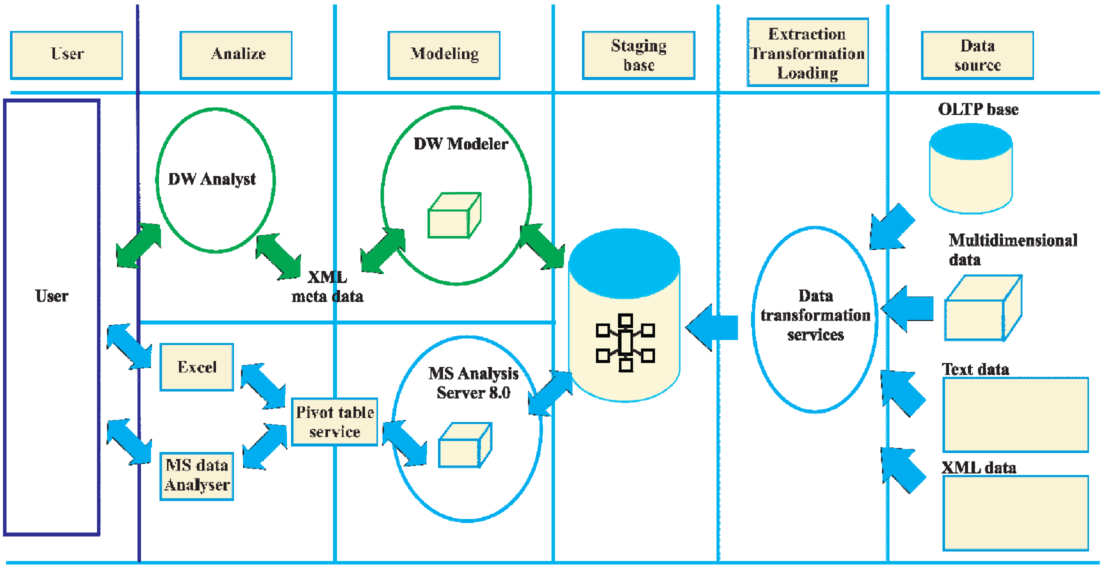


Figure 2 Architecture of the our solution compared with the Microsoft architecture

DW modeler and DW analyst applications completely rely on meta data which are located in XML formats, that provides easy manipulation of meta data. Each development approach provides ease of creation of DW, nevertheless, our approach gives several advantages. First advantage is usage of the FIRST and the LAST functions, which aren't provided in the other tools, made for developing of DW**.** In our opinion, limited number of functions for data aggregation is the biggest disadvantage for creating DW solutions. Precisely the main motivation for introduction of the new functions was this one. The second advantage is based on complete reliance on the XML meta data. Complete separation of the meta data provides more efficient conduction of changing structure of MD cube. Maybe the best example is that any kind of change made in structure of MD cube of Microsoft's solution would require new processing of it, even thou the change was connected to display data format, which didn't affected on context of data in MD cube at all. In our approach the changes, which have been made, are even more significant, such as changing the name of dimension or dimensional level that didn't require new processing of MD cube. By that it gains significant advantage in testing phase and adjusting of DW to users, the main reason for that is there is no waste of time on unnecessary processing any more, which are one of the main required operations in DW. In addition, our model is simpler and adaptable for comprehension of MD data models. Our approach is based on MD model that is suggested in [10] and is something that makes our model special and different from others. Usage of the MD model has fulfilled the essential assumption for creation of new algorithm for processing MD cube, which is based on history of user queries. The algorithm is, in fact, very similar to data mining algorithm APRIORI, which is according to [3] and [10] one of the most famous and mostly used data-mining algorithm. Differences are based on input data format, which uses frequencies for queries aggregation, and because of that all of candidates for aggregation are known in advance. Additionally frequencies of user queries can be pondered with average data size for the purpose of better space usage. Based on modified algorithm we could generate frequent candidates for creation of aggregation. Table 1 shows the comparison between our algorithm and algorithms that are mainly used for DW realization.

Method of creation aggregation	***Algorithm input***	***Choosing an aggregate***	***Result***	***Algorithm asumptotion***
Common algorithm eg. (BUPS, BPS)	Dimensions and cardinality	Based on optimal ratio between number of aggregates and memory space usage	The greater number of aggregations at given limitations	Each dimension has a same usage frequency

Method of creation aggregation	***Algorithm input***	***Choosing an aggregate***	***Result***	***Algorithm asumptotion***
Algorithm based on query history	Dimensions, cardinality and query history	Based on usage frequency and average data size	the lesser number of the more frequent aggregation that take up as little space as possible	Each query has different appeara nce rate and after several created queries

Table 1 Comparison of algorithms based on query history and the algorithms which are based on optimization ratio of number of aggregated data and necessary memory space.

Applying of a proposed algorithm based on history of queries makes possible the improvement of developed DW by removing dimensions and dimension levels which are rarely used in queries. Concrete results are shown in table 2 on example of five developed DW.

Data warehouse	***Rarely used dimension levels***	***Frequency***	***Total number of queries***	***Aggregation***
Windows logs [7]	Time(minute)	10	720	Don’t create

Faculty of organizational science – student data service [11] and [13]	Statute(name)
	PlaceOfHabitation(Name)
	PlaceOfBirth(Name)
		0
		10
		2
			4562
			4562
			4562
				Don’t create
				Don’t create
				Don’t create

FOREX [9]	Time(Quartal)
	Time(Second)
		5
		35
			2850
			2850
				Don’t create

Financial accounting	-	-	-	-
Merchandise Accounting	-	-	-	-

Table 2 Improving realization of DW using a query history

Based on a result from Table 2, we conclude that in three DWs (Windows Logs, FOS student data service and FOREX) there are some dimensions and dimensional levels that have been rarely used. Algorithms based on history of queries would not generate aggregation for those dimensional levels, whereas usual algorithms will start from cardinal dimensional levels and in many cases will generate exactly those aggregations which users wouldn't use in their analysis.

E.g. In ten cases of processing MD cube based on Microsoft solution (with different requests for taking place in workspace and performances), dimension *statute* from DW for student data service of FOS was in the aggregations. According to the Table 2 it is easy to notice that in data processing that dimension aren't used at all. Table 3 represents basic results of algorithm based on history of users queries in relation to algorithms based on optimization of relationship between number of aggregation and needed memory space.

Element	***Algorithms based on optimization of ratio between number of aggregation and memory space usage***	***Algorithms based on query history***
Number of aggregations	Bigger	Smaller
Number of queries successfully resolved from aggregations	Smaller	Bigger
Memory space usage	Bigger	Smaller
Choosing an aggregate	For dimension levels with small cardinality	For most frequent dimension levels

Precondition	Nothing	Query history

Table 3 Comparison of algorithm performance

Hereafter, there will be described application that we develop for the purpose of modeling DW and data analysis.

3. SOFTWARE SOLUTION

Figure 3 shows simplified scheme DW Analyst application.

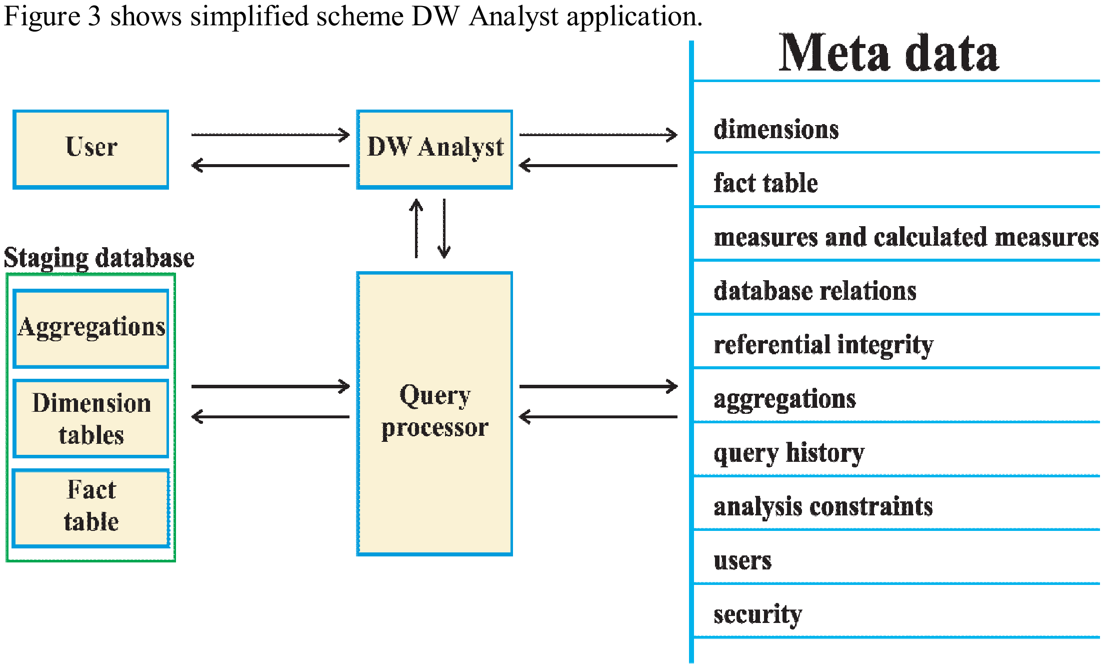


Figure 3 Simplified scheme of DW Analyst application

User makes graphic queries in DW analyst application; it is formed with help of object model represented in the Figure 4b and meta data which was represented on the right side of Figure 3.

In object model represented on Figure 4a (Microsoft solution), main object for connecting user application and MD cube is *Connection* in which is defined location of the cube (client or server), *name* of the cube and appropriate way for assessing it. *Connection* object contains catalogue where definitions of the cube are placed. In *cubes definitions* there are collections for dimensions, hierarchies and levels. Object *Cellset* represent the query result and it consists of cells, axes, positions and members that are needed to be represented.

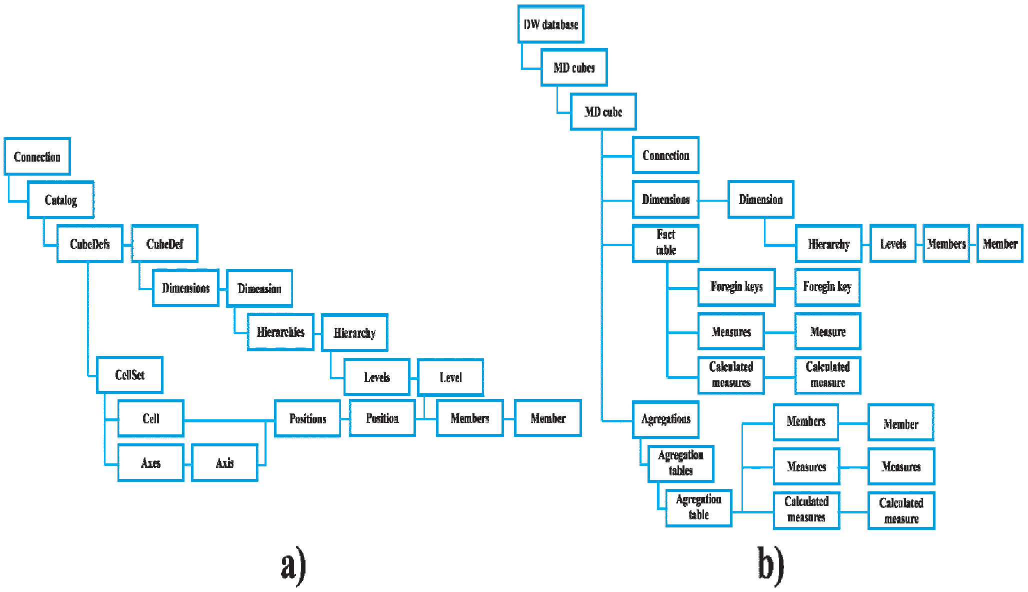


Figure 4a ADO MD object model according to [41] and 4b our object model from [10]

Figure 4b presents an object model that is described in [10]. Primary objects in the model are *MD cube*, *Connection*, *Fact table*, *dimension* and *aggregation*. ADO MD model on which Microsoft's solution is based, and the other solutions which are using the same standard are more general, but semantically substantially poor. Basic advantages of the approach that we suggested in [10] are in the use of greater amount of objects which respond to composed elements of DW and usage of aggregations as a separated element, while in the Microsoft approach they implicitly consist of objects *cellset* and *cells*. ADO MD mainly serves as a model for proceeding and working out of queries, and also for taking-over and presentation of the results. Microsoft's solution is also using Microsoft Decision Support Objects (MS DSO), which has a task to manage the DM. Object model that was suggested in [10] involves good qualities of both of models implemented in original way. MS-DSO, ADO-MD and represented object model are conceptual and they contain all necessary elements of overall MD object model. Suggested model in [10] is logically clearer and much more simple for understandings well as concrete use. Advantage of the ADO MD model is that the MDX query language is implemented on its base, while the model suggested in the [10] still doesn't have precise formulation of MD query language. Query resolution in model [10] is performed with methods contained in the MD cube.

Users interface is adapted to final users and it is very intuitive. Most of operations are performed with the help of wizard. Because of limited space we will represent only some of the most important screen forms. We will take FOREX DW as an example.

Figure 5 shows wizard for creation of meta database for relations.

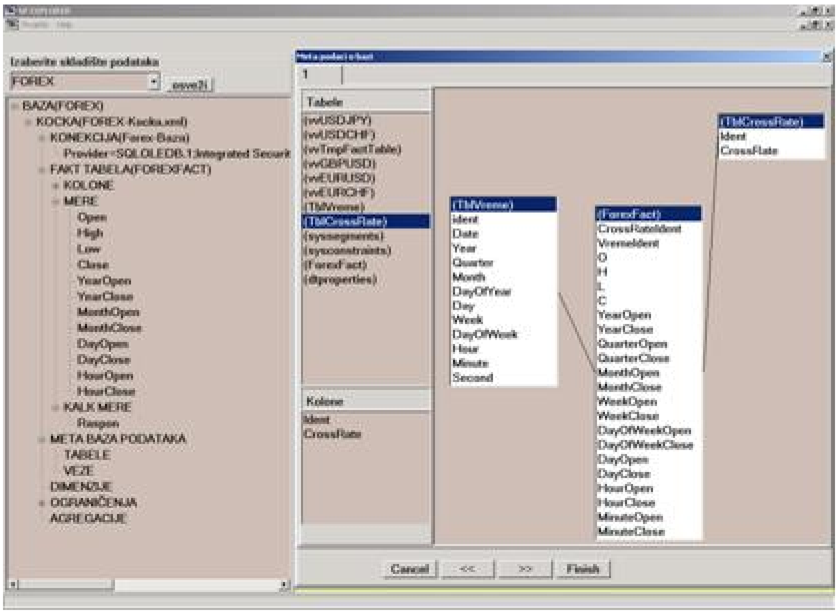
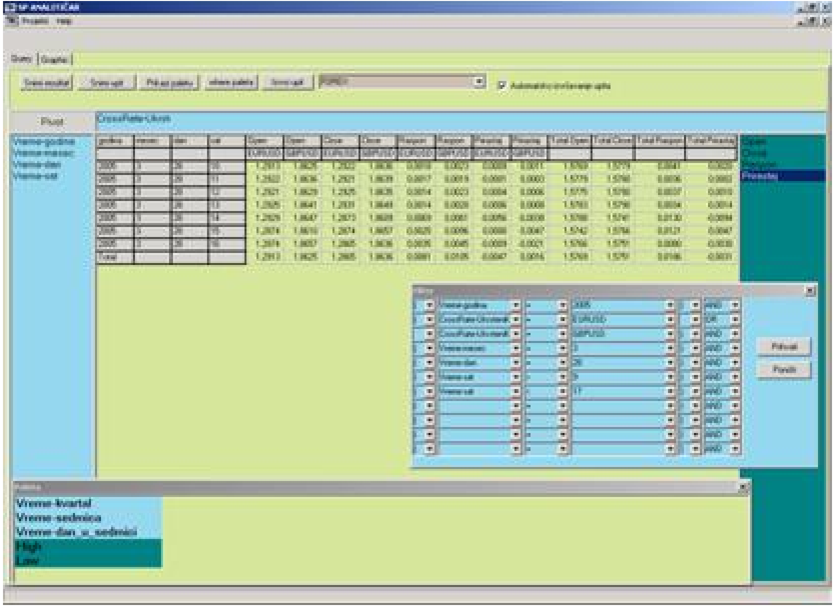


Figure 5 Wizard for creation of meta database for relations

Figure 5 shows data model for realization of FOREX DW. Model is graphically created. Objects are dragged (drag & drop) from the list placed on the left side of the screen, and then dropped on the window for drawing on the right side of the screen. Dragging one object to another creates a connection between the objects, while doing it, it is necessary to select certain columns through which they will be connected.

Figure 6 shows report from DW analyst application with complex filter structure based on hourly data. It is the report that shows paralleled movements of two cross rate foreign exchange course of March 28th 2005. in period from 10-16h.



godina	mesec	dan	sat	Open	Open	Close	Close	Raspon	Raspon	Priraštaj	Priraštaj	Total Open	Total Close	Total Raspon	Total Priraštaj
				EURUSD	GBPUSD	EURUSD	GBPUSD	EURUSD	GBPUSD	EURUSD	GBPUSD				
2005	3	28	10	1.2913	1.8625	1.2922	1.8636	0.0018	0.0023	0.0009	0.0011	1.5769	1.5779	0.0041	0.0020
2005	3	28	11	1.2922	1.8636	1.2921	1.8639	0.0017	0.0019	0.0001	0.0003	1.5779	1.5780	0.0036	0.0002
2005	3	28	12	1.2921	1.8629	1.2925	1.8635	0.0014	0.0023	0.0004	0.0006	1.5775	1.5780	0.0037	0.0010
2005	3	28	13	1.2926	1.8641	1.2931	1.8649	0.0014	0.0020	0.0006	0.0008	1.5783	1.5790	0.0034	0.0014
2005	3	28	14	1.2929	1.8647	1.2873	1.8609	0.0069	0.0061	0.0056	0.0038	1.5788	1.5741	0.0130	-0.0094
2005	3	28	15	1.2874	1.8610	1.2874	1.8657	0.0025	0.0096	0.0000	0.0047	1.5742	1.5766	0.0121	0.0047
2005	3	28	16	1.2874	1.8657	1.2865	1.8636	0.0035	0.0045	0.0009	0.0021	1.5766	1.5751	0.0080	-0.0030
Total				1.2913	1.8625	1.2865	1.8636	0.0081	0.0105	0.0047	0.0014	1.5769	1.5751	0.0186	-0.0033

(	Vreme-godina	=	2005		AND
(	CrossRate-Ulcrosrate	=	EURUSD		OR
	CrossRate-Ulcrosrate	=	GBPUSD	)	AND
	Vreme-mesec	=	3		AND
	Vreme-dan	=	28		AND
	Vreme-sat	>	9		AND
	Vreme-sat	<	17	)	AND

Figure 6. Example of report with functions FIRST and LAST and complex filter

Some of the previous examples show simplicity and flexibility of DW modeler and DW analyst. It is necessary to mention that the DW for Foreign Exchange was implemented also on the Microsoft platform which is described in details in [9] and it is much more complicated versus to the given solution.

4. COMPARISON

In this paper we represented DW modeler and DW analyst application. Tables 4, 5 and

6 show comparative characteristics of our solution compared to the Microsoft solution.

Elements of technology	***Microsoft solution***	***Our solution***
Simple cubes, More cubes in one base, virtual cubes	SUPPORT	SUPPORT
Simple and calculated measurements	SUPPORT	SUPPORT
Participation of the cube	SUPPORT	NOT SUPPORTED
Processing of the cube	SUPPORT	SUPPORT
Aggregations based upon first and last functions	NOT SUPPORTED	SUPPORT
Automatic improving of the model	NOT SUPPORTED	SUPPORT
Observation of substantial changes in meta data to avoid processing of MD cube	NOT SUPPORTED	SUPPORT
OLAP operations	SUPPORT	SUPPORT
Dimensional modeling	SUPPORT	SUPPORT
Meta data	SUPPORT	SUPPORT
ROLAP	SUPPORT	SUPPORT
HOLAP	SUPPORT	NOT SUPPORTED
MOLAP	SUPPORT	NOT SUPPORTED

Table 4 Comparison of the main elements of Microsoft solution and our solution

Characteristics	***Microsoft solution***	***Our solution***	***DW based upon materialized queries***
Algorithm for picking of the aggregate	Modification of BUPS and BPS algorithms	Algorithm based upon query history	Aggregates all queries from users application
Dimensional modeling	Strict dimensional hierarchies	Flexible dimensional hierarchies	Don't use dimensional modeling
Security	Procedural and declaration	Based on virtual cubes and filtering of meta data	Same as OLTP application
Introduction of new operators and functions for aggregating of data	Unknown	Practicable	Not applicable
Meta data	Binary	XML format	Do not use
Observation of substantial changes in meta data to avoid processing of MD cube	Can not be performed	Helped with XML meta data	Do not use
Taking restricted action so it would not be create illogical queries	It is not secured	Built-in model, by using concept of analysis constraint	It is not secured

Table 5 Comparative analysis of Microsoft and our solution

Element	***Microsoft***	***Our Solution***
Dimensions	More time dimensions(Strict hierarchy)	One time dimension(Flexible hierarchy)
Measures	For every dimension level in	One set of measures FIRST

	time dimensions, one set of redundant measures OPEN and CLOSE (because solution don't have FIRST and LAST function)	and LAST(implemented in model)
Possibility of meaningless queries	Very emphasized	Not possible(Analysis constraint concept)
Database size	Larger(redundant measures and unnecessary aggregations)	Smaller(don't use redundant measures and unnecessary aggregations)
Improvement of model	Not applicable	Included in model

Table 6 Comparative analysis of DW solution for FOREX in Microsoft technology and in our modeler

According to Table 4, 5 and 6 it can be concluded that suggested solution has numerous advantages than Microsoft's has, first of all in the access to the development of the DW, processing algorithm, guiding of users through the data analysis procedure, usage of XML meta data, and many other advantages. The greatest disadvantages of the suggested solution is the lack of cube partitioning and limitation only to ROLAP solutions.

5. CONCLUSION

This paper represents MD data model and suitable MD object model, and DW modeler and DW analyst tools. Certain advantages have been achieved by using special layer with meta data. Special importance is attributed to introduction of the approach which is based on analysis constraint with dimension levels and measurements that prevent user to create meaningless queries, that was disregarded in other methodologies.

Applying of our approach, six DWs have been realized, by now. Comparing the characteristics and performances of realized DW against the developed ones in Microsoft platform, it can be concluded that represented solution shows many advantages (shown in tables 4, 5 and 6).

The possibilities of improvement-represented application are significant and they can be performed in many different directions, like:

1) Development of data mining algorithm and it's implementation in MD model
2) Implementation of the dynamic security concept, for which the directions are given
3) Creating of the ETL tool which will unable the forming solution process to be more independent against some other tools and technologies
4) Improving the algorithm based on query history
5) Integration with the external systems
6) Other

In last decade DWs have become one of the most important types of (DSS) for enterprises. But, there are still numerous problems in realization, such as first of all: problems based on data quality, defection of relation models, different systems integration problems, new function introduction, data types which can be used as measures, etc. This paper shows some solutions, which can simplify the development of DW, and it also points out to a different edge of DW development. We are certain that DW will still have intensive development, and that will soon appear a new generation of DW, which will be significantly integrated with data mining techniques, expert systems and other methodologies for knowledge detecting.

BIBLIOGRAPHY

[1] Barry.D. (1997) "Data warehouse from architecture to implementation", Addison - Wesley, 1997;

[2] Craig R.S., Vivona J. A. i Bercovitch D, (1999), “Microsoft data warehousing“, Wiley;

[3] Devedžić V., (2000), “Inteligent information systems“, Digit;

[4] Group of authors, (1998), “Microsoft SQL Server 7.0 data warehousing training kit“, Microsoft Press;

[5] Inmon, W.H., (1996), „Building the data warehouse“, New York, John Wiley & Sons;

[6] Kimball R., (1996), „The data warehouse toolkit: practical techniques for building dimensional data warehouse“, John Willy & Sons;

[7] Krulj D., Čupić M, Martić M i Suknović M., (2003), “Design and implementation of data warehouse for windows logs analysis”, INFOFEST, Budva, Serbia and Montenegro, 2003;

[8] Krulj D., Čupić M., Suknović M. i Martić M., (2005), “Data warehouse based on materialized queries“, YUINFO, Kopaonik, Serbia and Montenegro, 2005;

[9] Krulj D., Čupić M., Suknović M. i Martić M., (2005), “Data warehouse for foreign exchange market“, JISA, Herceg Novi, Serbia And Montenegro 2005;

[10] Krulj D (2005) “Design of decision support systems based od data warehouse“, Phd Disertation, Faculty of Organizational Sciences, Belgrade, Serbia and Montenegro;

[11] Krulj D., Suknović M., Čupić M., Martić M. i Vujnović T., (2002), “Design and implementation of olap system for student data service of faculty of organizational sciences“, INFOFEST, Budva, Serbia and Montenegro;

[12] Krulj D., Cupic M., and Suknovic M. (2005), “Managing projects of data warehouse development”, JUPMA, Zlatibor, Serbia and Montenegro;

[13] Suknović M., Čupić M., Martić M. i Krulj D., (2005), “Data warehouse and data mining – a case study“, YUJOR, No. 15 Vol. I, pg. 125-145.