Data Mining, Multidimensional Databases and the Web for a better interpretation of data

Michael Haller, Georg Jenichl, and Josef Küng

Abstract— **With the permanent increase of data stored in databases and Data Warehouses it was necessary to invent new techniques for realizing queries on this data within an acceptable response time. The newly defined data model of multidimensional databases facilitates faster and easier analyses because of special designed structures. The ability to conceive the complex structures of the real world without any unnatural flattening makes the data modeling easier and offers much more performance for complex analysis.**

Moreover Data Mining can be used to extract useful and previously unknown information from large databases and Data Warehouses. With Data Mining tools it is possible to analyse such data and mine interesting knowledge from this data.

Another revolutionary evolution in computer science was the Internet: the global net offers access to a huge amount of data and everybody can analyse the data. An integration of the three technologies, the multidimensional databases and Data Mining on the one hand and the internet on the other hand, is a logical conclusion. In this proceeding the first step is the combination of these technologies. First, the theoretical basis added of these techniques will be explained, then the architectures and the prototypes, which can be used for interactive visualization will be presented. You will see that these technologies help users to find most important information in their Data Warehouses. Finally the tool WebMDDB, which can be used for interactive visualization of multidimensional data over the internet, will be presented.

Keywords— **Data Mining, OLAP, Multidimensional Databases, Internet, Java.**

I. Introduction

DATA Modeling for a Data Warehouse requires new modeling techniques and the creation of different types of schema than those used for traditional operational databases done with **OLTP** (**O**n**L**ine **T**ransaction **P**rocessing). In OLTP data was collected and stored for operations and control purposes. Nowadays, the Data Warehouse contains a database designed and optimized for data analysis and reporting capabilities with aggregated data from the different OLTP-sources. An easy access to the enterprise data must be supported to guarantee that the user will find it easy to access and analyze the data correctly [1]. Therefore, in Data Warehouses the user generally needs a multidimensional view for making decisions. Our visualization tools allow multiple data views and it is easy to make ad hoc decisions and analyses.

In figure 1 you can see the dependencies of Data Mining, Multidimensional Database (= MDDB) and Data Warehouses.

FAW Institute for Applied Knowledge Processing, Johannes Kepler University of Linz, mhaller@faw.uni-linz.ac.at

EURECOM Research Institute, Sophia Antipolis, jenichl@eurecom.fr

FAW Institute for Applied Knowledge Processing, Johannes Kepler University of Linz, jkueng@faw.uni-linz.ac.at

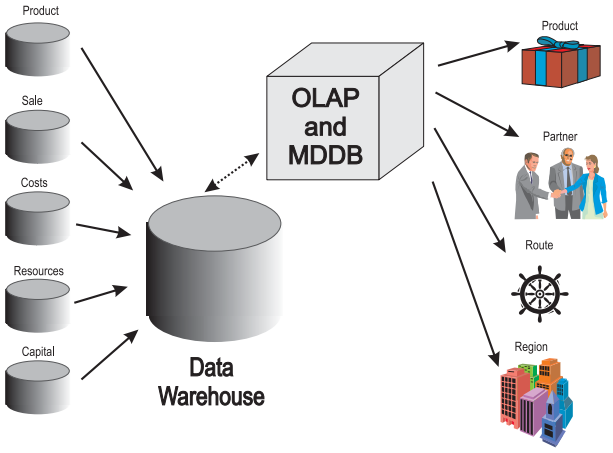

Fig. 1. The process of creating an analytical database

In the next section, we will introduce our OLAP concepts, where aggregations, summarizations and historical data can be calculated and stored in advance. Furthermore, you will see that the main goal of OLAP is a **fast access**, **concurrency**, and **integrity**.

With verification, the business user creates a hypothesis (e.g. business questions) and then tries to confirm his hypothesis by accessing the data in the Data Warehouse. Typical verification mode tools are query tools, reporting systems and multidimensional analysis tools. In the discovery mode, the tool tries to discover characteristics in data, i.e. patterns and associations that are not previously known or suspected by the business user. A typical discover mode tool is a **Data Mining tool** [2].

With **informational processing** we can perform tasks like data and statistical analysis, query and reporting. The data that is accessed and processed may be historical or fairly recent and lightly or heavily summarized. The result is shown by using reports and charts. The scope of this technique is the 2D or 3D analysis of historical data for understanding the past. **Analytical processing** also supports the verification mode, but its goal is to make the data available to the business user in a business user's perspective. With **slice and dice**, **drill-down** and **roll-ups** we can answer difficult questions like e.g. how many Suzuki cars did we sell in Japan in the first quarter of 1998 that had a audio system with a price less than 100.000 Yen?

II. Data Mining

The core components of Data Mining technology have been under development for decades in research such as

databases, machine learning , statistics and artificial intelligence.

Databases are critical to everyday commerce, and computers process and record massive amount of transactions. The fundamental concept of databases is the query model (e.g. ask a question to the database and the database is replying), whereas Data Mining uses another query form. A typical query is: What will happen with my product in the future or what are implications of a new business strategy? Unfortunately the database systems of today offer little functionality to support such 'mining' applications. At the same, machine learning techniques perform poorly when applied to such large datasets. This is one main reason that this huge amount of data is still unexplored.

With statistical analysis we can detect unusual patterns of data. These patterns are explained with statistical and mathematical models. Typical techniques are linear and nonlinear analysis, regression analysis, univariate, multivariate and time series analysis. With statistical tools we can successfully reduce the analysis time and free scarce resources for other analysis activities. To use these tools a business user must select and extract the right data from the Data Warehouse or datamart. The key features for extraction are knowledge discovery algorithms for pattern and relationship recognition, which are derived from the artificial intelligence sector.

III. OLAP databases

OLAP involves representing data as the user defines dimensions. Dimensions are almost related in hierarchies and a multidimensional database can have multiple hierarchies. Multidimensional database servers are designed to optimize the storage of dimensional data and provide flexible query and computational operations on dimensions with performance that dramatically exceeds what is possible with SQL and today's relational database implementations. Analysis requirements span a spectrum from statistics to simulation. The two forms of analysis most relevant to mainstream business users are commonly known as 'slice and dice' and 'drill-down' or 'roll-up'.

A. Slice and Dice

OLAP enables end users to slice and dice consolidated information in order to view data from many different perspectives. With this technique we can look at multidimensional correlations to uncover business behaviour and business rules. Users can 'cut' and 'rotate' particular pieces from the hypercube (the aggregated data) along any dimensions. The ease and speed with which a rotation can be performed is another example of the inherent advantages of manipulating data in a multidimensional array. Each rotation yields a different slice or a two dimensional table of data. Therefore the rotation is very often called **'data slice'**. The number of possible views increases exponentially with the number of dimensions. In a three dimensional cube with three dimensions like `model`, `color` and `dealership` has six different views:

- model by color (with dealership in the background)
- color by model (with dealership in the background)
- color by dealership (with model in the background)
- dealership by color (with model in the background)
- dealership by model (with color in the background)
- model by dealership (with color in the background)

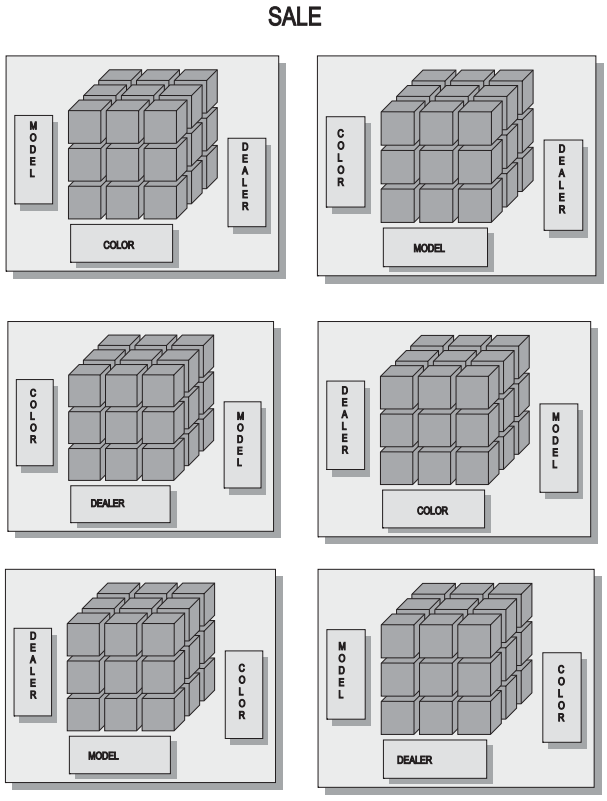

Fig. 2. A 3-D data-cube with six different views

B. Drill-Down and Roll-Up

OLAP allows users to **'drill'** or navigate through information to get more detail. Using the drill-down feature, users can view finer levels of detail within a dataset. In addition to drill-down, **'roll-up'** allows to look at a closer view of dataset.

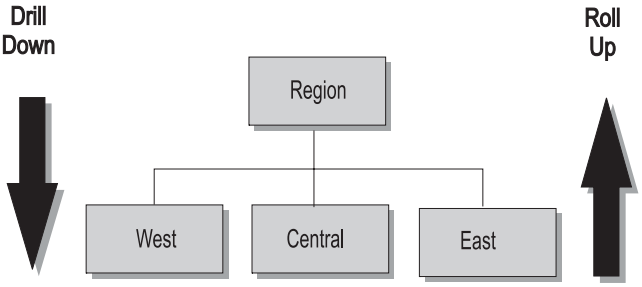

Fig. 3. The Drill-Down- and Roll-Up-function

C. MSQL - Multidimensional Structured Query Language

MSQL is a **M**ultidimensional **S**tructured **Q**uery **L**anguage, which provides a flexible access to multidimensional database systems. It is based on the relational SQL

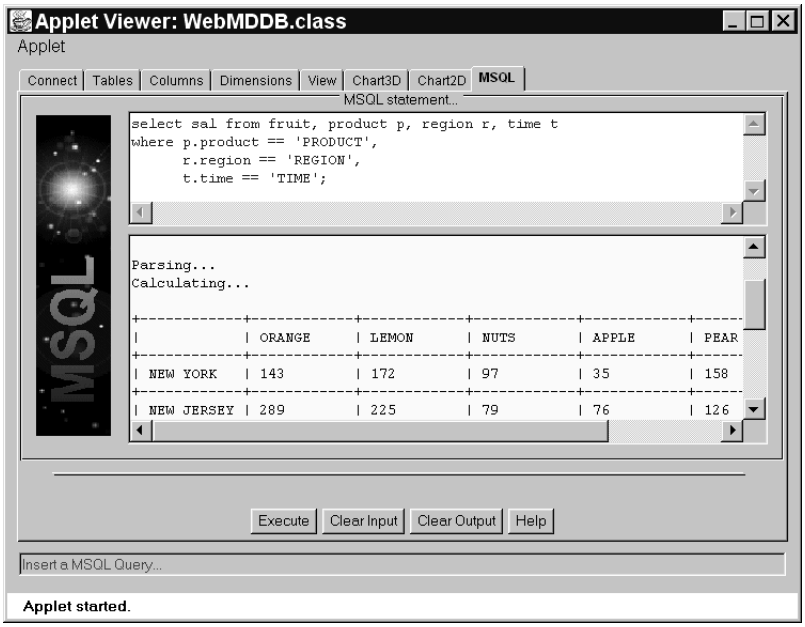

Fig. 4. Graphical User Interface for the Multidimensional Structured Query Language

and resembles CQL (Cube Query Language), which has its origin at the University of Erlangen [3]. Our OLAP tool reformulates a query from a graphical user interface. At the moment there is no standard for a multidimensional query language to formulate textual complex decision support queries. Therefore, our tool **WebMDDB** has integrated an own query language, where the (Extended Bacus Naur Form) EBNF-grammar [4] has the following structure:

```
MSQL           = "SELECT" cell
                 "FROM" cube
                  "," dimension dimVariable
                  {"," dimension dimVariable}
                 [WhereStat] ";".

WhereStat      = "WHERE" Condition
                 {"," Condition}.

Condition      = LeftParameter Operator
                 RightParameter
                 {"AND"
                  LeftParameter Operator
                  RightParameter}.

LeftParameter  = dimVariable "." position.

Operator       = ("="|"!")"=".

RightParameter = "'" name "'".
```

Fig. 5. Input used to produce this paper.

One of the primary goals of MSQL was to define a short and easy readable grammar for multidimensional queries, so that the user can immediately formulate statements without any problems.

To get a valid result, the analyst must use at least two dimensions. With the third dimension and the additional `where`-clause the drill-down and roll-up can be realized. Using the operators `'=='` and `'!='` the user has the possibility to limitate dimensions and the `and`-combination allows the creation of more complex queries.

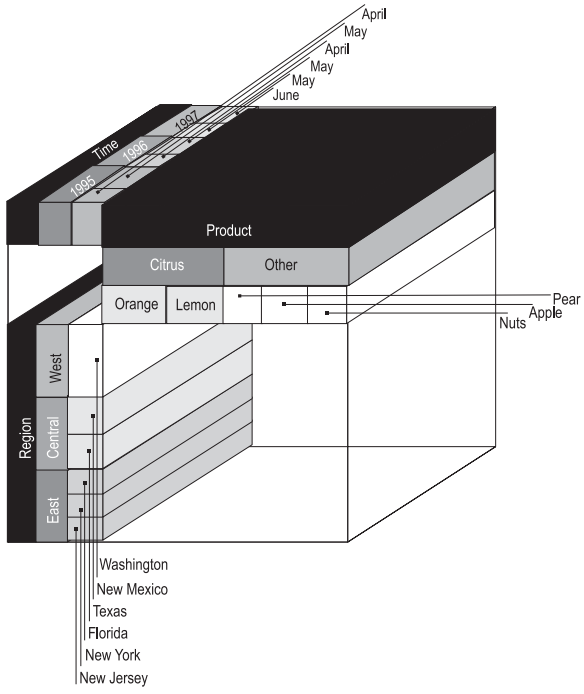

Fig. 6. The `'Fruit'`-data-cube

• If you select all products of all regions from the whole period from the data-cube 'Fruit', the MSQL-statement has the following structure:

```
select sal from fruit,
product p, region r, time t
where p.product == 'PRODUCT',
       r.region == 'REGION',
       t.time == 'TIME';
```

• In the next example the sale of products should be limitated to all eastern regions of 1998. Therefore, you have to change the `Where`-clause for the dimension `region` (ranging-operation) as well the dimension `time` (drill-down-operation):

```
select sal from fruit,
product p, region r, time t
where  p.product == 'PRODUCT',
       r.region == 'EAST',
       t.time == 'T1998';
```

• In the third example the sale should be limitated to all citrus-fruits and to the apples for all eastern regions produced in the 1998s. The `Where`-clause has to be modified including an `AND`-statement: Therefore, the where clause was modified to the following statement:

```
select sal from fruit,
product p, region r, time t
where p.product == 'CITRUS' AND
      p.other == 'APPLE',
      r.region == 'EAST',
      t.time   == 'T1998';
```

D. Performance Advantages

In contrast to relational databases the multidimensional structure has much **'knowledge'** about where a particular piece of data lies. The multidimensional system has to search only along the dimensions to find the matching 'record' or cell. Using a 10x10x10 array as an example, a relational system requires a search through all 1000 record in the worst case. The multidimensional system however has to search along three dimensions of 10 positions to find the matching cell. This is a maximum of 30 position searches for the array versus a maximum of 1000 records searches for the table. Another feature is that the values are stored in arrays and updates don't impact the index. Therefore we obtain fast responses to complex queries.

In the last sections we have illustrated some basic Data Mining and OLAP features. In the next section we will show how to embed these features in a distributed environment and how to realize a distributed Data Mining system via the World Wide Web.

IV. OLAP, Data Mining and the Web

In the article 'Data Warehouse And The Web' Neil Raden points out, that in the past 12 months the two most pervasive themes in computer science have been the Internet and Data Warehousing [5]. The integration of these two

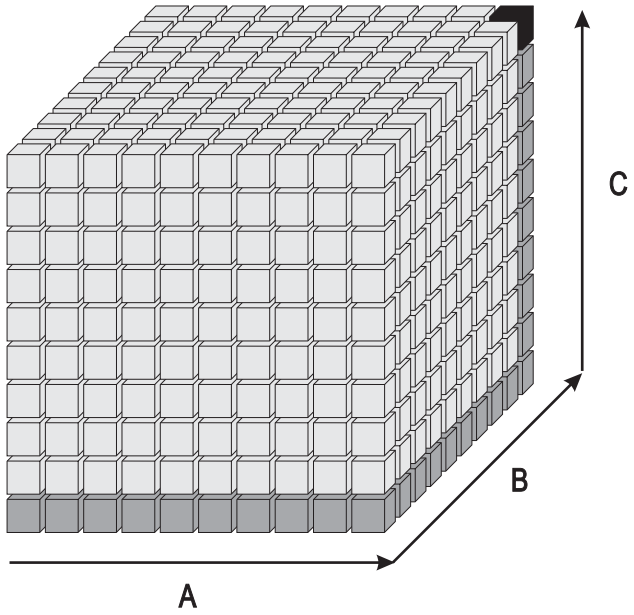

Fig. 7. In the worst case you need to compare 30 cells in a 3-D cube with 10 positions

giant technologies would be a logical conclusion and many database vendors are gearing up to offer these technologies through the World Wide Web. The Data Warehouse, with its underlying support for **ROLAP** (**R**elational **OLAP**) and **MOLAP** (**M**ultidimensional **OLAP**) technology enables users to Slice and Dice through data to crystallize information meaningful for making decisions. The Web is the technology that allows widely distributed users affordable, reliable and uncomplicated access to information all over the world. Together, they create the underlying technological framework to extend astonishing analytical power to a world-wide electronic audience of millions. For several reasons, the logical platform for broader deployment of OLAP and Data Mining is the Internet/Intranet WWW in combination with **CORBA** (**C**ommon **O**bject **R**equest **B**roker **A**rchitecture). Our basic approach was to implement a subset of functions for a distributed application using the JAVA language, a JAVA enabled Web browser, JAVA Database Connectivity (JDBC) and a CORBA product. CORBA offers a complete distributed object platform and with these distributed objects we can innovate the Internet because the objects encapsulate the inner workings and present a well-defined interface. This means that an object's implementation does not affect other objects or applications because the object's interface remains the same.

With an **ORB** (**O**bject **R**equest **B**roker)-enabled browser a user can access objects from multiple servers and hosts [6] without regard to client and server operating systems and programming languages. The client object requests the services of other objects with the client ORB. Using an ORB, the client connects to server objects it wishes to use. It lets objects transparently make requests to other objects that are located locally or remotely.With the **IIOP** (**I**nternet **I**nter **O**RB **P**rotocol) the client ORB

then looks for ORBs on other systems (Data Mining server, OLAP server) and these servers provide objects that can support the requested services. Each object has its unique name, according to the CORBA naming service, which identifies it and an interface defined with **IDL** (**I**nterface **D**efinition **L**anguage). Once the ORB has found the requested object, the two objects can communicate by using IIOP. Suppose a business user wants to analyze his Data Warehouse. From his ORB-enabled Web browser he enters a URL of a valid Web server, which contains a HTML page with an embedded JAVA applet. The Web server sends the Web page via **HTTP** (**H**yper **T**ext **T**ransfer **P**rotocol) to the client. On the client side the user enters and selects interesting analysis tasks and with the client ORB, the IIOP messages are sent across the network via IIOP. The server ORB selects the corresponding server object and these objects can perform multiple tasks. One task could be to obtain the calculated Data Mining rules from the rule induction server object or the retrieval of huge data from the database for further analysis. The server object routes the information with IIOP back to the client applet, which displays the obtained data.

Our basic approach was to design distributed applications in which all of the user-side client software is implemented as JAVA applets, which use CORBA for remote operations with the rest of the application's software components. The user can transparently download the applets when they are needed, thus this removes the need for manually distributing and installing any application specific software. With these applets it was possible to create better user interfaces and to be platform independent. The next section will give a better overview of these tools.

At the moment the most common techniques are based on the **CGI** (**C**ommon **G**ateway **I**nterface): it allows a thin client to request the services of a decision support system via the Web server. After receiving a request for information via CGI, output from the decision support system is simply returned to the browser in HTML format. The disadvantage of this non JAVA solution is the data visualization, which is one of the most important key element for a business user. The advantage is that it was the first 3-tier client/server solution over the Internet and therefore used for many industry products. With JAVA it is also possible to transfer data with similar methods from decision support systems to thin clients (with Microsoft's ODBC protocol and the JDBC protocol defined by JavaSoft). Support of these two prominent delivery mechanism is a key factor in the acceptance of a decision support system in a Web-enabled environment, where JDBC and ODBC can be utilized to reach data from a Data Warehouse with direct access and more efficiency.

V. Visualization Tools

Visualization tools are critical to OLAP and Data Mining. Therefore, many innovative techniques must be developed to visualize relationships, that let the user easily and quickly view information. Our implemented visualization tools provide precise, visual summaries of data from a Data Warehouse and can be viewed graphically. We provide two different tools: a visual Data Mining tool and a visual OLAP database tool, which we will explain in this section. First the user should understand what is happening in the business. Then he must understand the behavior of customers and markets. Finally, there should be clear what can be done. While our multidimensional data analysis tool focuses mainly on **what** is happening [7], the Data Mining tool is focusing the **why** part. It focuses the need to discover the why and then to forecast possible actions.

Our tools support two modes for this task: **verification** and **discovery**. With verification, the business user creates a hypothesis and then tries to confirm his hypothesis by accessing the data in the Data Warehouse. In the discovery mode, the tool tries to discover characteristics in the data, i.e. patterns and associations that are not previously known or suspected by the business user.

A. Visual Data Mining Tool

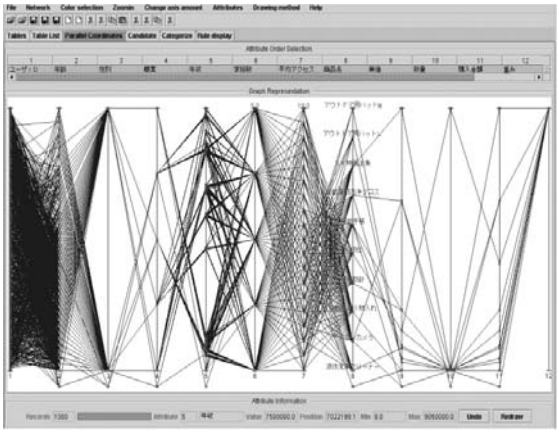

Fig. 9. JAVA powered parallel coordinate tool

There are a lot of techniques used for Data Mining: **neural networks**, **decision trees**, **rule induction** and **data visualization** [8]. These techniques scan through data stored in a Data Warehouse to detect hidden patterns. However, human pattern recognition skills are in many situations better than automated data mining algorithms with rule induction, neural netwporks, decision trees because it is easier for the analyst to gain a deeper, intuitive understanding of the data by representing a graphical image, which presents a large amount of information in a precise manner. With the construction of a visual interface with parallel coordinates we can detect many patterns. In this section we will present a JAVA powered visualization tool with parallel coordinates, as illustrated in figure 9. The construction of a parallel coordinate display is fairly simple. A single horizontal line is drawn and a series of vertical axes, each representing a separate variable, are placed with distances along the line. One record consists of n-1 lines which connect the n attributes of a record across the n axes.

With this kind of visualization you can drag and drop axes, change axes, view the minimum and maximum of

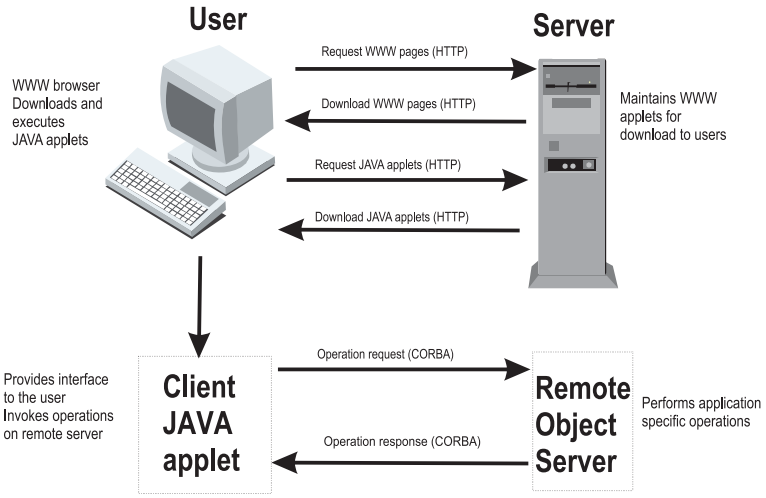

Fig. 8. CORBA Architecture

every attribute, see the categorization level and perform a useful zoom-in.

If we have a huge amount of records we will encounter that the display swamps beyond recognition. But even if it is completely overcrowded, we can use color highlighting operations to detect correlations across many variables. With this visualization the business analyst can display the relationships between many variables simultaneously rather than the perpendicular axes of traditional visualization tools. This helps us to manage large datasets and transform multivariate relations into well defined 2D patterns.

A.1 Advantages:

The parallel coordinate display enables us to see patterns and relationships that would be extremly difficult to determine by data ming algorithms, no matter what the computaional capabilities of the system are. Another advantage of parallel coordinates lies in the ability to visualize multidimensional relationships.

B. Visual OLAP database tool

WebMDDB (= **Web** Multi**d**imensional **D**ata**b**ase) is a Web-based OLAP-tool developed at the Johannes Kepler University of Linz [9]. The primary goal was to implement a tool, which allows the end user to have an interactive access to a huge amount of data via a Web browser. A good visual representation of the data is one of the most striking features of the tool. It is one of the first tools that offers a complete end-to-end JAVA solution for full-featured, browser-based OLAP decision support. By using JAVA it was possible to realize a good interactive visualization, which allows making queries and analysis from a web browser. The following subsection summarizes the characteristics of WebMDDB:

1. 100% end-to-end thin-client JAVA solution.
2. Fully interactive real-time interaction between the user and the data.
3. Works with any standard Web server and browser.
4. Easy to use (realization of drill-down, roll-up, ranging and rotation).
5. Interactive 2D and 3D presentation of data.
6. Integration of a multidimensional query language (MSQL), which is easy to use.

B.1 WebMDDB Benefits:

WebMDDB is a full-featured OLAP query builder. Using the simple, point-and-click OLAP Query Wizard, analysts and users can select their business information. After downloading the applet, you connect to the OLAP-database via JDBC. It is also easy to integrate other OLAP-databases by performing some modifications. Once you are connected to the database, you have the possibility to select dimensions, measures, levels, and much more. The following questions can be solved by using the WebMDDB-tool:

1. Which data-cubes are disposal in the database?
2. How do the dimensions of the data-cube look like and how many positions are available in the hierarchy-tree?
3. Which correlation does exist between the data and the dimensions?

B.2 Analysis:

The analysis tool enables real-time, interactive data visualization with 2D and 3D JAVA based charts for more intuitive recognition of patterns and trends. Users can dynamically select data, resize and rotate charts. Using the 2D visualization, the user can realize powerful tabular analysis. This allows him to **slice and dice** via pivoting and drill-down in a spreadsheet-like environment. Another feature of WebMDDB is the visualization of the distribution and the correlation between the data: each cell of the spreadsheet corresponds to a cube cell with its value. If there is a striking value compared to the others, the cell has a deeper color. The last analyzing feature is the integration of a multidimensional text based query tool (MSQL).

A fast access to data is a very important requirement in the Internet. Response time of more than 15 seconds is not

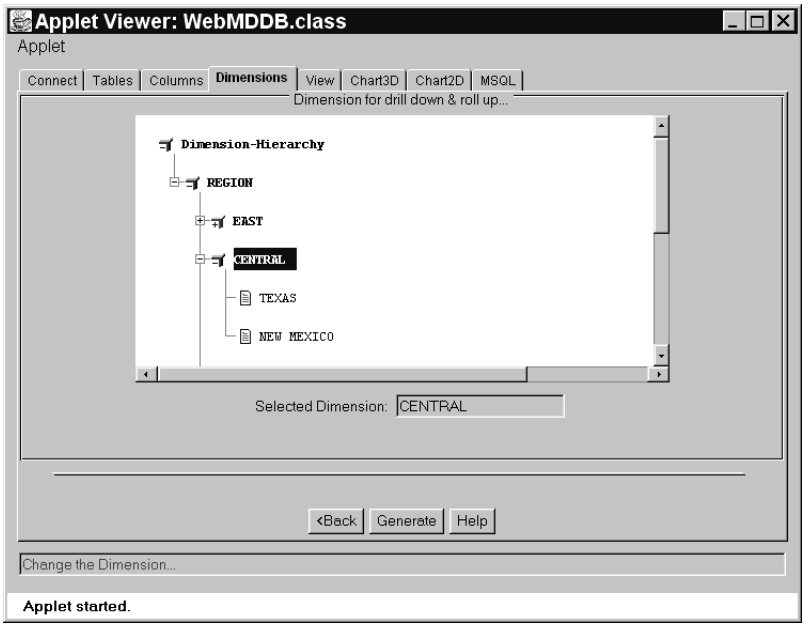

Fig. 10. The hierarchical representation for drill-down- and roll-up-operations

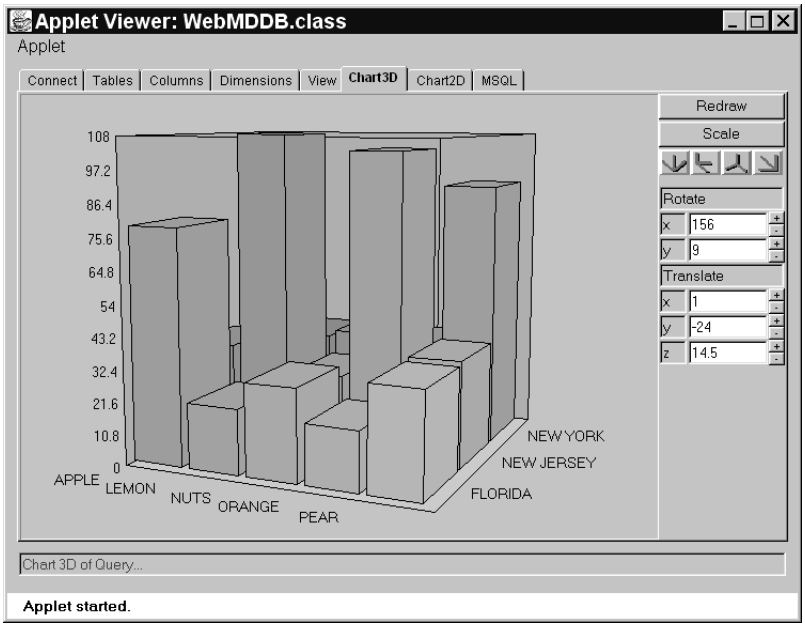

Fig. 11. The 3D visualization of WebMDDB

acceptable for the Internet-user. Therefore, the application has to be loaded fast and the user shouldn't wait too long for making data analysis.

B.3 Architecture:

Establishing an end-to-end JAVA environment, WebMDDB maintains persistence and state between the client and the server with its connection to the database. Furthermore, no CGI is used and no unnecessary processes are spawned on the server side. Putting all together, WebMDDB's JAVA architecture provides quick and easy data access and supports analysis capabilities for many users in a point-and-click environment.

VI. Further work

In the next years a variety of 3D information visualization techniques and research systems will be explored, which aid the human comprehension of large information systems. At the moment these techniques range from the familiar data presentation of surface plots and 3D bar charts through to the creation of abstract data spaces. The most popular visualization techniques at the moment are:

- Perspective walls
- Cone trees

- Rooms
- Fish-eye views

Fig. 12. Sphere visualisation produced using VizNet

In figure 12 you can see a similar visualization to the perspective wall technique. All objects are mapped onto the surface of a sphere. Unrelated and not important objects are displayed further from the object of interest and thus become less visible as they move round to the opposite side of the sphere. This provides a natural fisheye view which emphasises objects of interests [10].

The 3D-Rooms metaphor has been developed at XeroX Parc. It allows another way for users to structure and organise their work by allocating certain tasks to certain rooms and moving between rooms as needed.

The information cube (see figure 13) is a technique developed by Rekimoto and Green to visualize hierarchical information using nested translucent cubes: the available display is partitioned into a number of rectangular bounding boxes which represent the tree structure. Parent items are subdivided into boxes which represent their children, and so on [11].

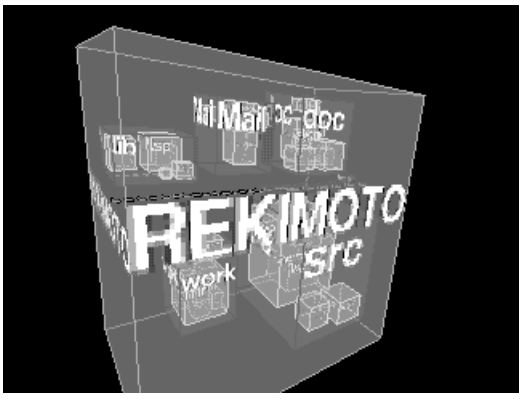

Fig. 13. An information cube visualisation

All of them are researching projects with the aim to get a better visualization of huge data and they are collected in the article "Three Dimensional Information Visualization" (see [12]).

VII. Conclusion

We have seen that one of the most important features for Data Warehouses is a well defined visualization for a huge amount of data stored in Data Warehouses. Otherwise the end users loose the overview and are not able to make a good decision. The advantage of our approach is the possibility that data from databases can be distributed in a convenient manner that allows the user to browse and analyze the down-loaded information inside a web-browser. Furthermore, we are platform-independent due to the JAVA portability and can perform this analysis no matter which operating system we are using. Therefore, the user interface makes it possible to gain a deeper, intuitive understanding of data.

In the future, we should direct our attention to a well defined visualization on the data. One of imaginable possibilities could be realized with Virtual Reality and **VRML** (**V**irtual **R**eality **M**odeling **L**anguage): the user could navigate in his virtual environment and be involved directly in his data [13].

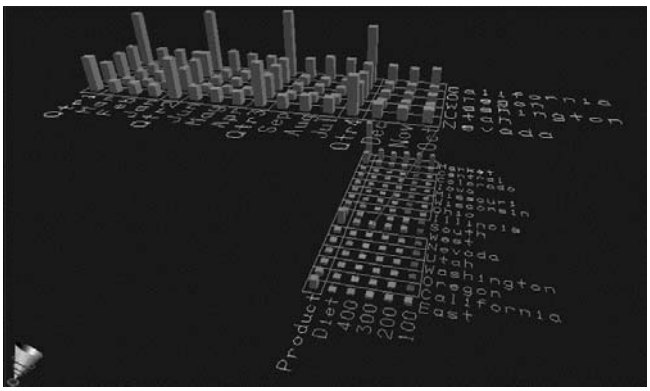

Fig. 14. VRML visualization from Arbor Essbase

Acknowledgments

To Katharina and Chicago...

References

[1] D. R. McClanahan, "Data Modeling for OLAP," *Database Advisor*, , no. 66–70, March 1997.

[2] F. Kwakkel M. Kersten, "Research and Business Challenges in Data Mining Technology," *Datenbanksysteme in Büro, Technik und Wissenschaft*, vol. 1, pp. 1–16, March 1997.

[3] W. Lehner A. Bauer, "The Cube-Query-Language (CQL) For Multidimensional Statistical And Scientific Database Systems," in *5th International Conference on Database*, DASFAA'97, Ed., Melbourne, Australia, April 1997.

[4] N. Wirth, *Compiler construction*, vol. 1 of *International computer science series*, Addison-Wesley, 1 edition, 1996.

[5] N. Raden, "Warehouses And The Web," *TechWeb*, May 1996.

[6] Netscape, "Corba: Catching the next wave," *Netscape White Paper*, 1997.

[7] J. Küng M. Haller, G. Jenichl, "Data Mining and Multidimensional Databases are the key to Data Warehouses and WWW," in *6th Interdisciplinary Information Management Talks*, IDIMT'98, Ed., Zadov, Czech Republic/Europe, October 1998.

[8] G. Jenichl, *Implementation of an Intranet Data Mining System and Visualization for Data Gold*, Sophia Antipolis - France, 1 edition, July 1998.

[9] M. Haller, *Multidimensional Databases and a Web-access realized with JAVA*, University of Linz - Austria, 1 edition, December 1997.

[10] K.M. Fairchild, *Virtual Reality: Applications and Explorations*, Academic Press Processional, Camebridge, MA, alan wexelblat edition, 1993.

[11] *The Information Cube: Using Transparencty in 3D Information Visualization*, 1993.

[12] Peter Young, "Three dimensional information visualization," Tech. Rep. 12, Department of Computer Science, University of Durham, November 1996.

[13] Arbor, "VRML: Visible Decision VRML Visualization," http://www.webgate.arborsoft.com/vdi/vrml.html, Juli 1997.

[14] G. Schaufler M. Haller, G. Jenichl, *Einführung in Virtual Reality*, GUP (Graphische und Parallele Datenverarbeitung), University of Linz - Austria, 1 edition, June 1996.

[15] Jakob Nielsen, "Interface Design for Sun's WWW Site," http://www.sun.com:80/sun-on-net/uidesign, November 1996, Stand vom Oktober 1997.