ON-LINE ANALYTICAL MINING FOR ADVANCED BUSINESS INTELLIGENCE SOLUTION

Marius GURAN[1], Aref MEHANNA[2], Bilal HUSSEIN[3]

Mineritul de date (DM) se referă la procesul de căutare şi descoperire de paternuri în colecțiile de date pentru a le prezenta într-o formă cât mai utilă şi predictibilă pentru procesele decizionale. Prin DM se poate descoperi cunoaşterea care nu se poate obține prin metode clasice de tip OLAP. În lucrare se propune un model OLAP pentru DM, numit OLAM (OLAP Mining), util în exploatarea unor Baze de Date (DB) şi Depozite de Date (DW) pentru aplicații de tipul BI (Business Intelligence).

Data mining (DM) deals with the proactive process of searching and discovering the patterns in the data and present it in a more predictable and useful format. It discovers the hidden knowledge within the information which is not possible with traditional relational or on-line analytical processing (OLAP) technologies. OLAM (OLAP Mining) model, which is based on the influence domain, is proposed in this paper. An exploratory analysis can be made in large DBs or DWs, based on OLAM, in BI applications.

Keywords: Business Intelligence (BI), Data Base (DB), Data Warehouse (DW), Data Mining (DM), On-Line Analytical Processing (OLAP), On-Line Analytical Mining (OLAM)

1. Introduction

Identifying the patterns in the business data is a required step for the analysts to take right decisions at the right time for business applications. For this reason there is more insight on the need for DM for completing OLAP in BI applications. Relational and OLAP tools are good for finding and reporting information that is already existing and visible within the data. Hidden knowledge that represents patterns and regularities in data can't be easily found by traditional reporting. In Relational and OLAP cases the selection and grouping criteria are known in advance. DM provides a powerful solution when the specific selection and grouping criteria are not known in advance, but are derived from the

[1] Professor, University POLITEHNICA of Bucharest, ROMANIA, e-mail: mguran@mix.mmi.pub.ro
[2] PhD student, University POLITEHNICA of Bucharest, Romania.
[3] PhD student, Université de Valenciennes-LAMIH, France

data values. DM can be highly effective for predicting, identifying, optimizing, targeting, focusing, and preventing data.

OLAP mining or OLAM is a mechanism which integrates OLAP with DM so that mining can be performed in different portions of DB or DW and at different levels of abstraction at user's finger tips. With rapid developments of DW and OLAP technologies in DB industry, it is promising to develop OLAM mechanisms.

OLAM is not only for data characterization but also for other DM functions including association, classification, prediction, clustering, and sequencing. Such integration increases the flexibility of mining and helps users find desired knowledge. In this paper, we introduce the need of OLAM and discuss the concept of how OLAM should be implemented in a DM system.

2. Factors considering DM for OLAP

The motivations for using OLAM are:

- **Data availability in source Systems:** Detailed data is available from source systems, preferably on a near real-time basis. Having detailed data would be a good candidate for accurate and predictable results.
- **Huge data volume:** Large data sets that can be difficult to analyze effectively using other tools lend themselves to DM solutions. Also, the statistical functions in DM require a large sample set in order to produce meaningful results.
- **Complexity to identify trends:** Having multiple factors enter into to forecasting or discovery analysis lends itself to DM, particularly when the appropriate grouping structures are not known in advance.
- **Automating with minimum user interaction:** Because DM is driven by data values. The same solution can be implemented at different customer locations, achieving customized behavior with no changes to the application.

Comparing DM to OLAP we can underline:

- **DM is not a DW:** DW, relational or OLAP can be used for mining process but DM itself is not DW store for storing warehouse objects such as facts, dimensions.
- **DM is not a reporting store:** DM is not a report store. It provides a method for analyzing data and making decisions. It does not provide any reports other than the analyzed data
- **DM is not an OLAP:** Online Analytical processing stores the DW data in multi dimensional store and also does aggregates accordingly. DM does

not require the data to be in multi dimensional or aggregations. It cannot be treated as a replacement of OLAP store.

- **DM is not data visualization:** DMX queries (Query language typically used for accessing DM models) are to be issues against the DM models to get the results of algorithms in mining models. There is a limited interface for viewing the DM predictions in BI Development Studio, additional visualization tools must be used to interface with external clients.

Though OLAP is a data store in a multi-dimensional structure that includes aggregations, it cannot replace DM models, as OLAP requires predefined grouping buckets. OLAP data can be used as an alternative to detailed relational data as a source for DM models. The DM models then provide further analysis and additional insight on the patterns and predictions. The following table (Table 1.1) [4, 6, 7] shows what is possible and/or not possible with both OLAP and DM models.

Table 2.1

Comparing essential points between OLAP and DM

OLAP	DM
o Typically focuses on historical facts	o Typically focuses on future outcomes or trends
o Aggregates data using pre-defined groupings	o Requires detail data
o Verification driven/Factual results	o Discovery driven
o Ad hoc queries and reports	o Statistical and machine learning techniques
o Limited ability to include reliability estimates with predictions	o Data models available for predicting, discovering patterns, estimating and producing accurate results for trend analysis and forecasting
o OLAP can be used as a data source for DM models	o DM results can also be used in OLAP applications by incorporating new predictive variables or scores as dimensions or attributes in your OLAP tool

3. DM and OLAP methodology

DM is frequently described as "the process of extracting valid, authentic, and actionable information from large DBs or DW" In other words; DM derives patterns and trends that exist in data. These patterns and trends can be collected together and defined as a mining model.

Building a DM model is part of a larger process that includes everything from defining the basic problem the model will solve, to deploying it into a working environment. This process can be divided into the following steps [4, 6, 7]:

3.1. Defining the problem: this involves defining the expectations from DM including: business analysts' requirements, marketing strategies, portfolio, forecasting, and decision support requirements. For example, a grocery store owner needs to find out the cross-sell of the customers in order to increase the customer base by providing what they need, together. For a grocery store manager, projected sales information may not be as important as the cross-sell requirement. The next steps of the DM act on the data collected for satisfying the business requirement.

3.2. Data Preparation: The Data Extraction details the type of data sources and extraction methods for data collections effectively. Typically the source transaction system consolidates all the discrete channels of data at one place and applies transactions, but it is not necessarily be one source of information. DM involves analyzing the source data at a broader level that includes both internal and external to the system. Data collection is completely based on the business requirements not necessarily extracting all the data. Following are some examples: for internal data sources: company activities, customer records, webs sites, mail campaigns, purchasing transactions, inventory. For external data sources: partners and supplies that contain external credit agencies, market surveyors, customer feedbacks.

DM models can extract data either from relational structures or from OLAP store. DM algorithms implementation work same for both relational and OLAP models. The only difference is the source structure and format.

Relational mining models can be built from data that is stored in relational DB systems. This type of mining model is based on data that is retrieved from any OLE DB data source. A data source may contain both single case and multiple case tables. Case is considered to be a look up or master table in relational transactional systems and as dimension in relational dimensional models. Transactional tables or fact tables where there can be more than one record exist per one case record, treated as Nested for DM models. DM models can extract data from these relational models for modeling and processing. Mined data (that contains the results from models in the form of predictions, patterns) can be extracted using queries (also called DMX) to interface and provide interactions to the analysts.

OLAP Cubes contain complex number of members and dimensions in a multi dimensional structure. Unlike relational DM models, OLAP mining models do not preserve the granularity of the data. This is because the patterns and trends of the data might be lost in the process of creating aggregations. This makes it difficult to manually find the hidden patterns within these members and dimensions. To overcome this difficulty, you can use the OLAP DM models that are based on OLAP data sources. These models allow you to find the complex patterns and apply them to business decisions. Similar to the relational DM models, the case key and case level columns from an OLAP data source need to be specified while designing DM model. While specifying the nested tables, only the measure groups that apply to the dimension are displayed. Select a measure group that contains the foreign key of the case dimension, as nested for DM model.

Data quality and transformations: In brief, even If your sources are clean, integrated, and validated, they may contain data about the real world that is simply not true. This noise can, for example, be caused by errors in user input or just plain mistakes of customers filling in questionnaires. If it does not occur too often, DM tools are able to ignore the noise and still find the overall patterns that exist in your data

Data preparation and cleaning: is an often neglected but extremely important step in the DM process. It is not required to cleanse the data before the extracted by DM process if the intention is to find out data quality problems using DM models.

DM can handle either numeric or text based data: Here the input columns may have descriptive (text) content, but the prediction columns contain numeric data. Simple operations on numeric data as Greater, Less, Percentage can be applied to deduce more meaningful patterns and forecast. While DM is typically concerned with the detection of patterns in numeric data, very often information that is critical to the business is stored in the form of text. Unlike numeric data, text is often difficult to deal with. Text mining generally consists of the analysis of (multiple) text documents by extracting key phrases, parsing, concepts, etc. and the preparation of the text processed in that manner for further analyses with numeric DM techniques.

3.3. Exploring Data: This step involves checking if the required data containing the expected information. For example, if cross-sell information is to be analyzed as part of DM models then individual customer transactions and products purchased must be captured. The term data reduction in the context of DM usually applies to the goal of aggregating the information from large datasets into manageable information chunks. Data reduction methods include simple tabulation or aggregation, or more sophisticated techniques like clustering and principal components analysis.

3.4. Building the Model: A model typically contains input columns, an identifying column, and a predictable column. Data type for the columns can be defined in a mining structure based on which algorithms process the data. The following basic terms would be useful to understand about the column types.

- **Continuous Column:** This column contains numeric measurements typically the product cost, salary, account balance, shipping date, invoice date having no upper bound.

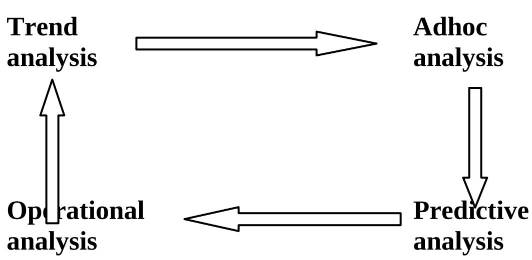
- **Discrete Column:** These are finite or limited unrelated values such as Gender, location, age, telephone area codes. They do not need to be numeric in nature, and typically do not have a fractional component.
- **Discretized Column:** This is a continuous column converted to be discrete. For example: grouping salaries into predefined bands.
- **Key:** The column which uniquely identifies the row, similar to the primary key. This is sometimes called the Case attribute.

Most available algorithms while using DM for BI are:Decision Trees Algorithm, Clustering Algorithm, Bayesian Algorithm, Association Algorithm, Sequence Clustering Algorithm, Time Series Algorithm, Neural Network Algorithm,Logistic Regression Algorithm, Linear Regression Algorithm.

A DM model applies a mining model algorithm to the data that is represented by a mining structure. Model Parameters and boundary values can be defined on the DM algorithms and usage parameters on DM model column. I can define columns to be input columns, key columns, or predictable columns.

3.5. Evaluating and Validating the model: Different models for a given business problem could be used for analyzing various business scenarios, identifying the analytical requirements, tuning the parameters and evaluating the results of the models to make a business decision.Considering various models and choosing the best one based on their predictive performance (i.e., explaining the variability in question and producing stable results across samples). This may sound like a simple operation, but in fact, it sometimes involves a very elaborate process. There are a variety of techniques developed to achieve that goal, typically applying different models to the same data set and then comparing their performance to choose the best.

Data analysis life cycle represent the maturity model of the analysis as showing in Figs. 2.1.



Figs. 2.1. Data Analysis Life Cycle

- **Operational analysis** is nothing but business transaction Reports (closing bank balances, who was admitted into the hospital today, how many support calls are closed today etc)
- **Trend analysis** understands the growth of the historical data over a period of time.

- **Ad hoc analysis** is business context analysis (Products sales by region) or it can also be used for finding the root cause such as sudden decrease in sales of a product due floods or natural calamity
- **Predictive analysis** is predicting the patterns for the future (also called forecasting)

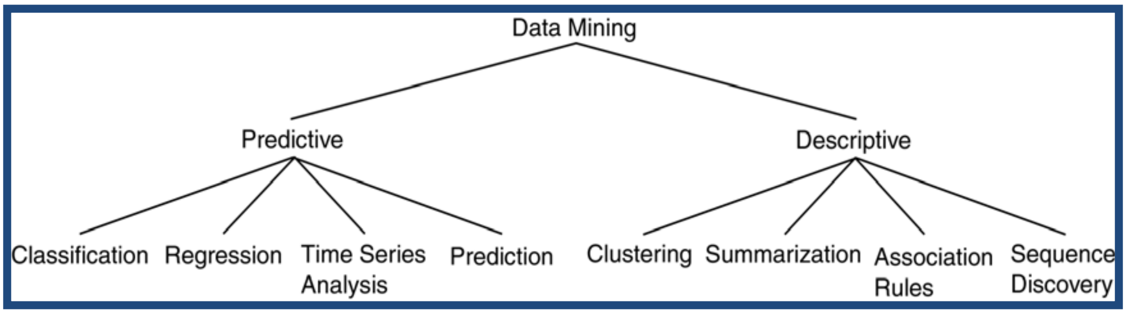
3.6. Deploying and updating the models: Once the right DM model is chosen and trained with the source data, you deploy it on the server for DMX queries and APIs to access. These queries act directly on the deployed models and feed the result to the interface for customer interaction and decision making. The processed models may need to be updated as the business requirement changes. For example, a change in business requirements may require you to consider choosing a different mining model. Alternatively, the input data may change or you may have new values you need to predict. DM models can be trained (processed) and deployed by using tasks and transformations. This is helpful when you want the model to respond to source data changes on a near-real-time basis. Also, mining results can be accessed by using tasks and transformations for feeding OLAP or relational structures. This is useful when end users access and analyze the data using existing OLAP or relational reports.

3.7. Accessing the model: Once the model is built and deployed, the next step is to access the mined information from the front-end interface for further analysis. The query language DMX is typically used for accessing DM models. DMX is similar to the MDX query language for OLAP queries and to the SQL query language for relational queries. Following tasks can be performed using DMX query language: creating mining structures and mining models, processing mining structures and mining models, deleting or dropping mining structures or mining models, copying mining models, browsing mining models, predicting against mining models

4. DM strategies

There are two main kinds of models in DM, they are Predictive and Descriptive. Predictive models can be used to forecast explicit values, based on patterns determined from known results. For example, from a DB of customers who have already responded to a particular offer, a model can be built that predicts which prospects are likeliest to respond to the same offer. Descriptive models describe patterns in existing data, and are generally used to create meaningful subgroups such as demographic clusters.

The following picture [4, 8, 9] represents the hierarchical view of DM models and segregations of algorithms into these two models:



- Prediction is the estimation of future outcomes, such as predicting which customers will be loyal, predicting which customers will respond to a promotion, works on continuous attribute set. Time Series and Decision Trees Algorithms would be better examples to implement these scenarios.
- Segmentation algorithms divide data into groups, or clusters, of items that have similar properties. An example of a segmentation algorithm is the Clustering Algorithm.
- Summarization algorithms are similar to clustering algorithm but instead of grouping the data, it would quantify the members of the group, such as group one has more number of line items available and it has most probability of occurring. Clustering Algorithm would give this information apart from clustering the selected data set.
- Association algorithms find correlations between different attributes in a dataset. The most common application of this kind of algorithm is for creating association rules, which can be used in a market basket analysis. An example of an association algorithm is the Apriory Algorithm.
- Sequence analysis algorithms summarize frequent sequences or episodes in data, such as a Web path flow. An example of a sequence analysis algorithm is the Sequence Clustering Algorithm.

5. Right DM algorithms for a better OLAM

Choosing the right algorithm to use for a specific business task can be a challenge. While you can use different algorithms to perform the same business task, each algorithm produces a different result, and some algorithms can produce more than one type of result. For example, you can use the Decision Trees algorithm not only for prediction, but also as a way to reduce the number of columns in a dataset, because the decision tree can identify columns that do not affect the final mining model.

You can use different algorithms to perform the same business task and each algorithm produces a different result. Lift charts would be useful to check the accuracy of the DM models once built on the input data.

Use more than one algorithm to produce results and analyze the results for choosing the right one. Different algorithms produce different results. The choosing of the algorithms is based on the accuracy and on the business need

Use algorithms together, use some algorithms to explore data, and then use other algorithms to predict a specific outcome based on that data. For example, you can use a clustering algorithm, which recognizes patterns, to break data into groups that are more or less homogeneous, and then use the results to create a better decision tree model.

Use multiple algorithms within one solution to perform separate tasks. For example, regression tree algorithm can be used to obtain financial forecasting information, and a rule-based algorithm to perform a market basket analysis.

Drill down analysis on processed mining models would be useful to check the granular content behavior over the time to denote the interactive exploration of data, in particular of large DBs. The process of drill-down analyses begins by considering some simple break-downs of the data by a few variables of interest (e.g., Gender, geographic region, etc.).

If the individual attributes you have are transaction amounts, you should model them as continuous rather than discrete. Below lists (Table 4.1) [6, 8, 9] out some of the popular business scenarios for which DM models are sought and right model chosen Real Customer Scenarios:

Table 4.1

Right Algorithm(s) for Some Business Scenarios

PROBLEM	SOLUTION
The marketing department of a car company needs to identify the characteristics of existing customers to determine whether they are likely to buy a product in the future	By using the Decision Trees Algorithm, the marketing department can predict whether a particular customer will purchase a product. The Decision Trees algorithm can make a prediction based on the customer information, such as demographics or past buying patterns.
The marketing department of a car company needs to predict monthly car sales for the coming year. It also needs to identify whether the sales report of one model can be used to predict the sales of another model	By using the time Series Algorithm on the past three year historical data, a DM model that forecasts future car sales can be produced. Using this model, you can also make cross predictions to determine the relationship between sales trends of individual car models
The car company is redesigning its web site to favor the sale of products	By using the Association Rules Algorithm on the company records of each sale in the transactional DB, the company can identify the various car models and accessories that tend to be purchased together. The company can then predict additional items that a customer might be interested in.
Predicting when the customer balance	Making use of Decision Trees and Neural

would become zero, a typical banking requirement so that alerting the customer or automatic moving the funds from another account	Networks. Check the recent transaction of the customers when it hit zero, come out with input attributes such as premium repay, summer bike renting, gender, and age then prepare the model. This can also be predicted with average monthly balance, average weekly balance of all the customer and predict when current week balance is less than average weekly balance and could reach zero if continues.
In price comparison system which matches the best prices for a purchase. There could be a number of major items in an order and each major item could have multiple related sub items. The other variables that affect the price include trade-ins if any, sales going on at the time of order, number of units etc.	Making use of decision trees, neural nets, or logistic regression would be a better option, consider the parameters such as product type, weight, cost, location, date of the year as input parameters and predict the cost of the product across different variables

Also the table below lists (Table 4.2) [6, 7, 8] the right algorithms to use for some decisional mining tasks like predictions and groupings:

Table 4.2

Right Algorithm(s) to Use in Predictions and Groupings

TASK	ALGORITHMS TO USE
o Predicting a discrete attribute. For example, to predict whether the recipient of a targeted mailing campaign will buy a product.	o Decision Trees Algorithm o Naive Bayes Algorithm o Clustering Algorithm o Neural Network Algorithm
o Predicting a continuous attribute. For example, to forecast next year's sales.	o Decision Trees Algorithm o Time Series Algorithm
o Predicting a sequence. For example, to perform a click stream analysis of a company's Web site.	o Sequence Clustering Algorithm
o Finding groups of common items in transactions. For example, to use market basket analysis to suggest additional products to a customer for purchase.	o Association Algorithm o Decision Trees Algorithm
o Finding groups of similar items. For example, to segment demographic data into groups to better understand the relationships between attributes.	o Clustering Algorithm o Sequence Clustering Algorithm

6. DM guidelines for OLAM

The list of suggestions [2, 4, 6] related to implementation for the DM models and algorithms are:

- Considering updating DM model when volume of data increases
- Consider slicing the Source Cube if OLAP is source or Filter out if Relational storage is used as Mining source
- Cleanse the incoming data is modeling is not for data quality identification
- Consider DM models built from same data source
- DM models sourcing data from granular stores
- Use lift chart for finding the accuracy and deciding the modal
- Considering splitting the source data into a training set and a testing set
- DM models predict but decision is yours
- Consider having a statistical analyst for studying the models and aligning with business requirements
- Consider denormalized tables/views (star schema) as a source for DM models
- Consider using integration services for modeling, maintaining and reprocessing mining models
- Consider aggregating to the required level before building time series model
- Consider the detailed transaction table as nested and its master table as a case table

7. OLAM implementation architecture

Using previous ideas, we are to developing a DM system, which the OLAM Architecture describes.

Our OLAM architecture (Figure 6.1) [2, 3] facilitates constraint-based, multidimensional mining of large DBs and DWs. The architecture consists of four layers:

1. The **Lowest Layer** is the data repository layer, which consists to the supporting of DBs and DWs.
2. On top of it is the **Multidimensional DB Layer** (MDDB), which provides a multidimensional view (Cube) of data for OLAP and DM.
3. The essential layer for DM is the **OLAP/OLAM Layer**, which consists of two engines, one for OLAP and one for DM.
4. Finally, on top of the OLAP/OLAM layer lies the **User Interface Layer**, which provides easy-to-use interfaces. These interfaces let users construct DWs, create multidimensional DBs, select the desired sets of data, perform constraint-based interactive OLAP and DM, and visualize and explore the results.

The **OLAM architecture** provides a modularized and systematic design for a DM system and provides **several benefits** [1, 2, 3]:

- First, it takes advantage of widely available, comprehensive information processing infrastructure. These systems have been systematically constructed around relational DB management systems and DWs, which can store huge amounts of data. Data cleaning, integration, and consolidation have been largely performed in the construction of DWs. An efficient OLAM architecture should use existing infrastructure in this way rather than constructing everything from scratch.
- Another benefit of the OLAM architecture is that it provides an OLAP-based exploratory data analysis environment. The integration of DB and DW at one end, OLAP and DM at the other facilitates two things: First, it becomes possible to mine different subsets of data and at different levels of abstraction by drilling, pivoting, filtering, slicing, and dicing a multidimensional DB and the intermediate DM results. Secondly, it facilitates online, interactive selection of DM functions and interestingness thresholds. Performing these functions interactively and viewing the results with data/knowledge visualization tools will greatly enhance the power and flexibility of exploratory DM.

Many **unsolved problems** include:

- How to integrate constraint based mining methods based on different measures of interest; how to systematically design and implement a constraint-based **DM query language** (DMQL); and how to perform incremental mining by relaxing certain constraints. More research is needed to develop an integrated, constraint-based association-mining environment.
- We would also like to know how to apply constraint-based, multidimensional mining to other types of knowledge, such as characterization, classification, clustering, and anomaly analysis. These are unexplored but very promising areas for future research.

8. OLAM implementation challenges

A lot of challenges particularly related to the incoming data and its quality. An OLAM system may work perfect for consistent data and perform significant worse when a little noise exists to the incoming data. What we mean are the most prominent problems and challenges [7, 9] of data mining systems today.

- **Noisy Data:** In a large database, many of the attribute values will be inconsistent and/or incorrect. This may be due to erroneous instruments, human error during data entry, migration problems, missing values or incorrect transactions at the source system.

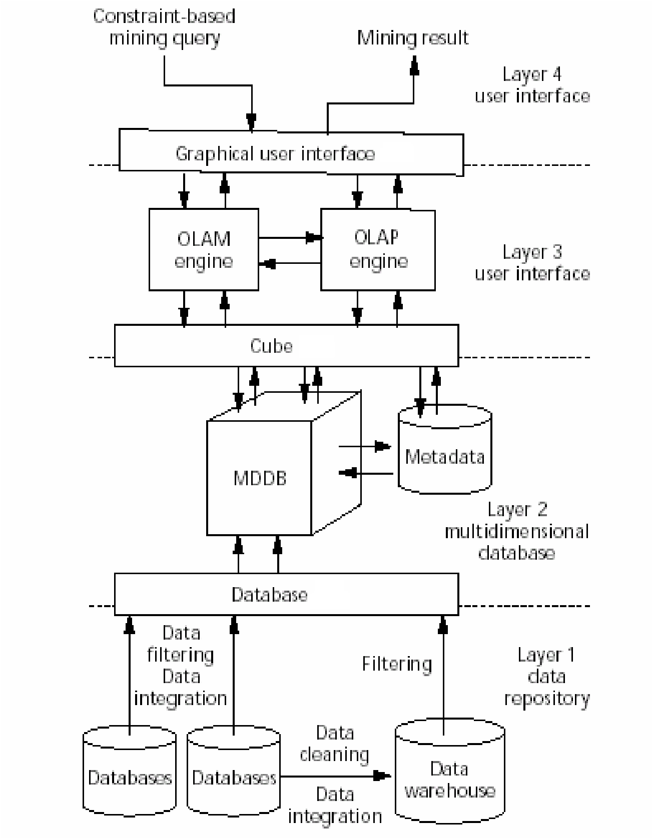


Fig. 8.1 Online Analytical Mining (OLAM) Architecture

- **Databases may be Huge:** Most DM algorithms have been created for handling very large data sets but what happens if you need to perform data mining on terabytes of source data? On the one hand, a huge input data source is advantageous, since that will often result in more accurate modeling, but the number of possible attribute values within a large dataset can be enormous. This means that processing the models will take enormous time and query performance can be too slow. One solution is to utilize less complex mining algorithms to minimize processing time. Other options include randomly select data from the source, or possibly summarizing to the minimum level of detail needed for the mining.

- **Complex incoming data:** Sometimes the incoming data is complex enough that data modelers cannot uniquely define the requirements:

    - **Non-Representative Data:** If the data in the training set is not representative of overall patterns, data mining will not produce accurate results. For example, if you are creating a model for employee wages, and the training set includes only salaried people, the model will not accurately represent hourly employees. Even when using large training data sets, you must be careful that the training data is representative and that it belongs to the same domain as the target data set.

    - **No Boundary Cases:** To find the real differences between two classes, boundary or unusual cases should be present. For example, if a DM system is classifying animals, and non-flying birds such as penguins are not included in the training set, the model may define a bird as an animal that has wings and can fly.

    - **DBs are Dynamic:** If the source DB structure changes frequently, it is a challenge for DM, as this requires changing and re-training the models. Ideally, the trained model should reflect the content of the source data at all times, in order to make the best possible classification. An important challenge for OLAM systems is to accomplish this, by changing its rules as the source data changes.

9. Aim and scope of OLAM

Nowadays, it is widely recognized that OLAP technology provides powerful analysis tools for extracting useful knowledge from large amounts of data stored in different and highly-heterogeneous formats, and very often distributed across networked settings ranging from conventional wired environments to innovative wireless and P2P networks. Several advantages confirm the benefits coming from such an analysis model:

1. The amenity of naturally representing real-life data sets that are multi-level, multidimensional, and highly-correlated in nature;
2. The amenity of analyzing multidimensional data according to a multi-resolution vision;
3. The rich availability of a wide class of powerful OLAP operators (such as roll-up, drill-down, slice-&-dice etc) and queries (e.g., range-, top-k, iceberg and gradient queries);

4. The integration of OLAP with more complex analysis tools coming from statistics, time series analysis, and DM algorithms.

An Elegant and Successful Solution in this line of research consists in coupling OLAP and DM tools and algorithms, which is the basis of OLAM model, proposed by *Jiawei Han*. [1] Basically, this proposal consists in meaningfully combining the powerful of OLAP with the effectiveness of DM tools and algorithms capable of discovering interesting knowledge from large amounts of data (e.g., the data cell set of a given OLAP data cube) by means of clustering, classification, association rule discovery, frequent item set mining, and so forth.

10. Conclusion

During the last decade, researchers have devoted their attention on the issue of meaningfully coupling OLAP and DM tools and algorithms, leading to the term **"OLAP Intelligence"**, which can be reasonable considered as one of the emerging research topics of next years in the context of knowledge discovery methodologies. This great interest is essentially due to both exciting theoretical perspectives, such as complexity issues of executing time-consuming DM routines over very large OLAP data cubes, and relevant application issues, which have a great impact in a excess of real-life scenarios ranging from conventional distributed DB management systems and cooperative information systems to innovative data stream management systems and sensor network data analysis tools.Despite these efforts, many aspects need to be further investigated in order to achieve a reliable **Convergence between OLAP and DM (OLAM)**, thus making this technology a reference for next-generation data-intensive analysis tools.

R E F E R E N C E S

[1] *Jiawei Han*, Towards on-line analytical mining in large databases, Intelligent Database System Research Lab, School of Computing Science, Simon Fraser University, Canada

[2] *Jiawei Han*, *H.S. Sonny, Chee, Jenny Y. Chiang*, Issues for On-Line Analytical Mining of Data Warehouses, Intelligent Database Systems Research Laboratory, School of Computing Science, Simon Fraser University, British Columbia, Canada V5A 1S6

[3] *Jiawei Han,* Constraint-Based, Multidimensional Data Mining, Simon Fraser University, Laks V.S. Lakshmanan, Concordia University, Raymond T.Ng, University of British Colombia,

[4] *Marius Guran, Aref Mehanna, Sebastian Roşu,* Knowledge and Data Management in Business Intelligence, CSCS16, Universitatea POLITEHNICA Bucureşti, Facultatea: Automatica si Calculatoare

[5] *Paul Sanders,* SQL Server 2005, Real-time Business Intelligence using Analysis Services, 2005

[6] * * * Business Intelligence / Microsoft Corporation 2006

[7] *Mike Biere*, Business Intelligence for the Enterprise, Publisher: Prentice Hall PTR, Pub Date: June 04, 2003, ISBN: 0-13-141303-1
[8] *René leFébure, Gilles Venturi,* Data Mining, Eyrolls, deuxieme édition 2001.
[9] Margaret H. Dunham, Data Mining, Introductory and Advanced Topics, Prentice Hall, 2003 by Pearson Education Inc, Upper Saddle River, New Jersey 07458
[10] * * * OLAP Intelligence, International Journal of Business Intelligence and Data Mining, InderScience Publishers