ON-LINE ANALYTICAL PROCESSING

Alberto Abelló and Oscar Romero
Universitat Politècnica de Catalunya
http://www.lsi.upc.edu/~{aabello|oromero}

SYNONYMS

OLAP

DEFINITION

On-line Analytical Processing (OLAP) describes an approach to decision support, which aims to extract knowledge from a data warehouse (see Data Warehouse definitional entry), or more specifically, from data marts (see Data Mart definitional entry). Its main idea is providing navigation through data to non-expert users, so that they are able to interactively generate ad-hoc queries without the intervention of IT professionals. This name was introduced in contrast to On-Line Transactional Processing (OLTP), so that it reflected the different requirements and characteristics between both kinds of uses. The concept falls in the area of business intelligence (see Business Intelligence definitional entry).

HISTORICAL BACKGROUND

From the beginning of computerized data management, the possibility of using computers in data analysis has been evident for companies. However, first analysis tools needed the involvement of the IT department to help decision makers to query data. They were not interactive at all and demanded specific knowledge in computer science. By the mid 80's, executive information systems appeared introducing new graphical, keyboard-free interfaces (like touch screens). However, executives were still tied to IT professionals for the definition of ad hoc queries, and prices of software and hardware requirements where prohibitive for small companies. Eventually, cheaper and easy-to-use spreadsheets became very popular among decision makers, but soon it was clear that they were not appropriate for using and sharing huge amounts of data. Thus, it was in 1993 that Codd et al., in [2], coined the term OLAP. In that report, the authors defined 12 rules for a tool to be considered OLAP. These rules caused heated controversy, and they did not succeed as Codd's counterpart for Relational Database Management Systems (RDBMS). Nevertheless, the name OLAP became very popular and broadly used.

Although the name OLAP comes from 1993 and the idea behind them goes back to the 80s, there is not a formal definition for this concept, yet. As proposed by Nigel Pendse in [6], OLAP tools should pass the FASMI (Fast Analysis of Shared Multidimensional Information) test. Thus, they should be fast enough to allow interactive queries; they should help analysis task by providing flexibility in the usage of statistical tools and what-if studies; they should provide security (both in the sense of confidentiality and integrity) mechanisms to allow sharing data; they should provide a multidimensional view so that the data cube metaphor can be used by users; and, finally, they should also be able to manage large volumes of data (gigabytes can be considered a lower bound for volumes of data in decision support) and metadata. However, there are not measures and thresholds for all these characteristics in order to be able to establish whether one of them is fulfilled or not, and therefore it is always arguable that a given tool fulfills them. Nevertheless, it is generally agreed that in order to be considered an OLAP tool, it must offer a multidimensional view of data.

Since their first days, OLAP tools have been losing weight and lowering prices, at the same time that they offered more functionalities, better user interfaces and easier administration. Thus, time has come for small companies to use OLAP. They can afford it and they are willing to use it in their decisional processes. Part of OLAP industry was associated into the OLAP Council (created in January 1995), whose aim was the promotion and

standardization of OLAP terminology and technology. However, some major vendors never became members of this council, so eventually it disappeared (last news date from 1999). Nowadays, there is not such standardization institution specifically devoted to OLAP. Therefore, it seems difficult to have a standard data model and query language in a near future, despite the fact that it is clearly desirable.

SCIENTIFIC FUNDAMENTALS

OLAP environments have completely different requirements, compared to OLTP. Figure 1 summarizes the main differences. Firstly, their usage is different. While OLTP systems are conceived to solve a concrete problem and are used in the daily work of companies, OLAP systems are used in decision support. Thus, in the first case, since the addressed problem can be completely specified, the workload of the system is clearly predefined. Conversely, a decision support system aims to solve new problems every day. Therefore, ad hoc queries are executed. From the kind of access point of view, OLTP systems read as well as write data, while OLAP systems are considered read-only, because decision makers do not directly modify data. Nevertheless, the queries in a decision support system are much more complex, since they usually include big volumes of information processed by joining several tables, grouping data and calculating functions. Queries in OLTP systems do not usually involve volumes of data of the same magnitude, neither as many tables, nor groupings or calculations. The number of records in OLTP operations can be estimated as tens or hundreds at most, while OLAP queries usually involve thousands or even millions of records. Finally, the number of users is also different in both kinds of systems. OLTP systems can have thousands or millions of users (like in the case of cash machines), while OLAP systems have tens or maybe hundreds of users.

	OLTP	OLAP
Usage	Application specific	Decision support
Workload	Predefined	Unforeseeable
Access	Read/Write	Read-only
Query structure	Simple	Complex
Records per operation	Tens/Hundreds	Thousands/Millions
Number of users	Thousands/Millions	Tens/Hundreds

Figure 1: Comparing OLTP vs OLAP

The main characteristic of OLAP is Multidimensionality. The data cube metaphor is used to make user interaction easier and closer to decision makers' way of thinking, who would probably find SQL or any other text-based query language hard to understand and error prone. Thus, it is much easier for them to think in terms of the multidimensional model, where a Fact (see Fact definitional entry) is a subject of analysis and its Dimensions (see Dimension definitional entry) are the different points of view that analysts could use to study the Fact. In this way, the instances of a Fact are shown in an n-dimensional space usually called Cube or Hypercube.

Market (billion US$)	ROLAP tools		MOLAP tools	
	Europe	USA	Europe	USA
2005	1	2	0.75	0.25
2006	1.5	2.5	1	0.5

Figure 2: Example of Cross-tab or Statistical Table representation of a $2 \times 2 \times 2$ data cube

In order to show n-dimensional Cubes in 2-dimensional interfaces, Cross-tabs or Statistical Tables such as the one in figure 2 (its data is entirely fictitious) are used. While in Relational Tables it is found that fixed columns and different instances are shown in each row, in Cross-tabs both, columns as well as rows, are fixed and interchangeable. In this example, you see three dimensions (i.e. Product, Place and Year) that show the different points of view to analyze the OLAP tools market.

Multidimensionality is based on this fact-dimension dichotomy (see Multidimensional Modeling definitional entry). A Dimension is considered to contain a hierarchy of aggregation levels (see Hierarchy definitional entry)

representing different granularities (or levels of detail) to study data, and an aggregation level to contain descriptive attributes. On the other hand, a Fact contains quantitative attributes that are called measures (see Measure definitional entry). Dimensions of analysis arrange the multidimensional space where the Fact of study is depicted. Each instance of data is identified (i.e. placed in the multidimensional space) by a point in each of its analysis dimensions. Two different instances of data cannot be spotted in the same point of the multidimensional space. Therefore, given a point in each of the analysis dimensions they only determine one, and just one, instance of factual data. Moreover, it is also worth to say that data summarization (see Summarizability definitional entry) performed must be correct, i.e. aggregated categories must be a partition (complementary and disjoint) and the kind of measure, aggregation function, and the dimension along which data is aggregated must be compatible (for example, stock, sum and time are not compatible, since stock measures cannot be added along temporal dimensions).

Operations

Unfortunately, there is no consensus on the set of multidimensional operations and how to name them. However, in [10] you find a comparison of algebraic proposals in the academic literature, besides a set of operations subsuming all of them. A sequence of these operations is known as an OLAP session. An OLAP session allows to transform a starting query into a new query. Figure 3 draws the transitions generated by each one of these operations (circles and triangles represent different attributes for Fact instances):

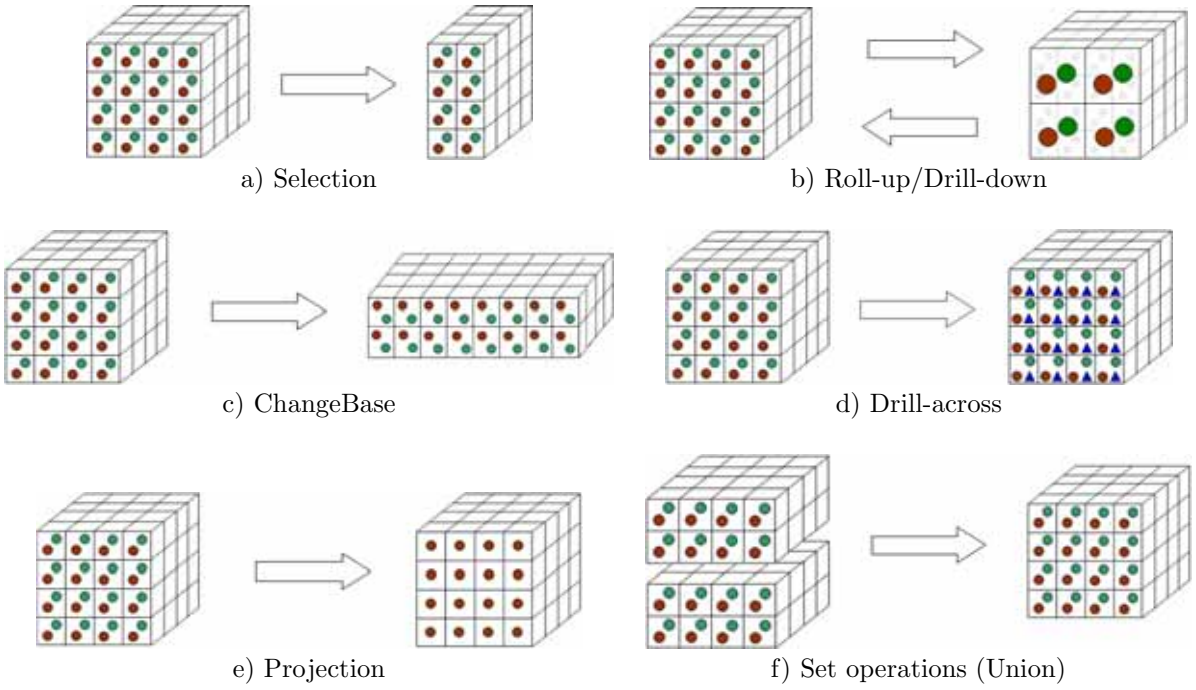

Figure 3: Schema of operations on Cubes

Selection or Dice: By means of a logic predicate over the dimension attributes, this operation allows users to choose the subset of points of interest out of the whole n-dimensional space (figure 3.a).

Roll-up: Also called "Drill-up", it groups cells in a Cube based on an aggregation hierarchy. This operation modifies the granularity of data by means of a many-to-one relationship which relates instances of two aggregation levels in the same Dimension, corresponding to a part-whole relationship (figure 3.b from left to right). For example, you could roll-up monthly sales into yearly sales moving from "Month" to "Year" aggregation level along the temporal dimension.

Drill-down: This is the counterpart of Roll-up. Thus, it removes the effect of that operation by going down through an aggregation hierarchy, and showing more detailed data (figure 3.b from right to left).

ChangeBase: This operation reallocates exactly the same instances of a Cube into a new n-dimensional space with exactly the same number of points (figure 3.c). Actually, it allows two different kinds of changes in the space: you can just rearrange the multidimensional space by reordering the Dimensions interchanging rows and columns in the Cross-tab (this is also known as Pivoting), or it could add/remove dimensions to/from the space.

Drill-across: This operation changes the subject of analysis of the Cube, by showing measures regarding a new Fact. The n-dimensional space remains exactly the same, only the data placed in it change so that new measures can be analyzed (figure 3.d). For example, if your Cube contains data about sales, you could use this operation to analyze data regarding production using the same Dimensions.

Projection: It selects a subset of measures from those available in the Cube (figure 3.e).

Set operations: These operations allow users to operate two Cubes defined over the same n-dimensional space. Usually, Union (figure 3.f), Difference and Intersection are considered.

This set of algebraic operations is minimal in the sense that none of the operations can be expressed in terms of others, nor can any operation be dropped without affecting its functionality (some tools consider that the set of measures of a Fact conform an artificial analysis dimension, as well; if so, Projection should be removed from the set of operations in order to be considered minimal, since it would be done by Selection over this artificial Dimension). Thus, other operations can be derived by sequences of these. It is the case of Slice (which reduces the dimensionality of the original Cube by fixing a point in a Dimension) by means of Selection and ChangeBase operations. It is also common that OLAP implementations use the term Slice&Dice to refer to the selection of fact instances, and some also introduce Drill-through to refer to directly accessing the data sources in order to lower the aggregation level below that in the OLAP repository or data mart.

Declarative languages

There are some research proposals of declarative query languages for OLAP. [1] proposes a graphical query language, while [3] proposes a calculus. From the industry point of view, MDX (standing for Multidimensional Expressions [5]) is the de facto standard. It was introduced in 1997, and in spite of the specification being owned by Microsoft it has been widely adopted. Its syntax resembles that of SQL.

```
[ WITH <MeasureDefinition>+ ]
SELECT <DimensionSpecification>+
FROM <CubeName>
[WHERE <SlicerClause> ]
```

However, its semantics are completely different. Roughly speaking, an MDX query gets the instances of a given Cube stated in the FROM clause and places them in the space defined by the SELECT clause. Moreover, complex calculations can be defined in the WITH clause, and the dimensions not used in the SELECT clause can be sliced in the WHERE clause (if not explicitly sliced, it is assumed that dimensions that do not appear in the SELECT are sliced at the higher aggregation level: All).

```
WITH MEMBER [Measures].[pending] AS '[Measures].[Units Ordered]-[Measures].[Units Shipped]'
SELECT
    {[Time].[2006].children} ON COLUMNS,
    {[Warehouse].[Warehouse Name].members} ON ROWS
FROM Inventory
WHERE ([Measures].[pending],[Trademark].[Acme]);
```

In the previous MDX query, an ad-hoc measure "pending" is firstly defined as the difference between units ordered and shipped. Then, the children of the instance representing year 2006 (i.e. the twelve months of that year) is placed on columns, and the different members of the aggregation level "Warehouse Name" on rows. Now, this matrix is filled with the data in "Inventory" cube, showing the previously defined measure "pending" and slicing "Acme" trademark.

KEY APPLICATIONS

Managers are usually not trained to query databases by means of SQL. Moreover, if the query is relatively complex (several joins and subqueries, grouping, and functions) and the database schema is not small (with maybe hundreds of tables), using interactive SQL could be a nightmare even for SQL experts. Thus, OLAP is used to ease the tasks of these managers in extracting knowledge from the data warehouse by means of Drag&Drop, instead of typing SQL queries by hand.

OLAP market is estimated around 6 billion US$ in 2006, which is mainly devoted to decision making. However, this paradigm can also be used in any other field with non-expert users, where schemas and queries are relatively complex. For example, its usage is under investigation in bioinformatics [8], and the semantic web [9].

FUTURE DIRECTIONS

OLAP is used to extract knowledge from the data warehouse. Another kind of tool used with this purpose are data mining tools (see Data Mining definitional entry). Till now, both research communities have been evolving separately. The former must be interactive, while the latter presents computational complexity problems. However, it seems promising to integrate both kinds of tools so that ones can benefit from the others. In fact, it was already suggested in [4], and some tools like Microsoft Analysis Services already integrate them in some way. Nevertheless, there is much work to do in this field, yet.

On the other hand, security is usually a flaw in data warehousing projects. [7] contains a survey of OLAP security problems. In the past, OLAP tools used to have just a few users and all of them had high responsibilities in the company, so this was not really a concern in the sense of confidentiality. Nowadays, with the increase in potential users of OLAP systems inside as well as outside the company, security has appeared as a priority in these projects (see Security in DWs definitional entry). Moreover, personal data (like those of customers) are usually analyzed in almost all companies. Thus, inference control mechanisms need to be studied in data mining as well as OLAP tools.

Other research directions in OLAP can be the improvement of user interaction and flexibility in the calculation of statistics (see Visual OLAP definitional entry), and the integration of what-if analysis (see What-if Analysis definitional entry).

URL TO CODE

Some OLAP vendors:

- •Microsoft Analysis Services:
  `http://www.microsoft.com/sql/technologies/analysis/default.mspx`
- •Hyperion Solutions:
  `http://www.hyperion.com`
- •Cognos PowerPlay:
  `http://www.cognos.com/products/business_intelligence/analysis/index.html`
- •Business Objects:
  `http://www.businessobjects.com/products/queryanalysis/olapaccess/businessobjects.asp`
- •MicroStrategy:
  `http://www.microstrategy.com/Solutions/5Styles/olap_analysis.asp`

Some open source OLAP tools:

- •Mondrian:
  `http://mondrian.pentaho.org`
- •Palo:
  `http://www.palo.net`

CROSS REFERENCE

Business Intelligence (BI)
Cube implementations
Data mart
Data Warehouse
Dimension
Fact
Hierarchy
Measure
Multidimensional modeling
Relational Database Management Systems (RDBMS)
Security in DWs
Star schema
Summarizability

RECOMMENDED READING

Between 3 and 15 citations to important literature, e.g., in journals, conference proceedings, and websites.

[1] Luca Cabibbo and Riccardo Torlone. From a Procedural to a Visual Query Language for OLAP. In *Proceedings of 10th International Conference on Scientific and Statistical Database Management (SSDBM'98)*, pages 74–83. IEEE Computer Society Press, 1998.

[2] Edgar F. Codd, S. B. Codd, and C. T. Salley. Providing OLAP to user-analysts: An IT mandate. Technical report, E. F. Codd & Associates, 1993.

[3] Marc Gyssens and Laks V. S. Lakshmanan. A Foundation for Multi-dimensional Databases. In *Proceedings of 23rd International Conference on Very Large Data Bases (VLDB'97)*, pages 106–115. Morgan Kaufmann, 1997.

[4] Jiawei Han. OLAP Mining: Integration of OLAP with Data Mining. In *IFIP TC2/WG2.6 Seventh Conference on Database Semantics (DS-7)*, volume 124 of *IFIP Conference Proceedings*, pages 3–20. Chapman & Hall, 1997.

[5] Microsoft. Multidimensional Expressions (MDX) Reference. Available at `http://msdn2.microsoft.com/en-us/library/ms145506.aspx`, 2007. SQL Server books online.

[6] Nigel Pendse. The OLAP Report - What is OLAP? Available at `http://www.olapreport.com/fasmi.html`, 2007. Business Application Research Center.

[7] Torsten Priebe and Günther Pernul. Towards OLAP Security Design - Survey and Research Issues. In *Third ACM International Workshop on Data Warehousing and OLAP (DOLAP 2000)*, pages 33–40. ACM, 2000.

[8] Erhard Rahm, Toralf Kirsten, and Jorg Lange. The GeWare data warehouse platform for the analysis of molecular-biological and clinical data. *Journal of Integrative Bioinformatics*, 1(4):47, 2007.

[9] Oscar Romero and Alberto Abelló. Automating Multidimensional Design from Ontologies. In *Proceedings of the ACM International Workshop on Data Warehousing and OLAP (DOLAP'07)*, pages 1–8. ACM, 2007.

[10] Oscar Romero and Alberto Abelló. On the Need of a Reference Algebra for OLAP. In *Proceedings of the International Conference on Data Warehousing and Knowledge Discovery (DaWaK'07)*, volume 4654, pages 99–110. Springer (LNCS), 2007.