OLAP – ONLINE ANALYTICALPROCESSING

PROCESAMIENTO ANALÍTICO EN LÍNEA

Recibido: 25 de marzo 2013- aceptado: 19 de junio 2013

Azoumana Kamagate, MsC.[1]
Universidad Simón Bolívar.

Keywords:
Data, Dimensions
Cubes

Abstract
This analytical process sets out the guidelines for the business world by carrying out a multidimensional analysis of high volume databases in organizations for decision-making. An OLAP system should obey the rules of Dr. Codd: you should have a multidimensional vision of the data: thinking in terms of the dimensions and metrics of business. Not tables and fields.

Palabras claves:
Datos, Dimensiones,
Cubos

Resumen
Este procesamiento analítico le da pautas al mundo empresarial, en permitir realizar análisis multidimensional de las bases de datos de gran volumen en las organizaciones, para realizar toma de decisiones de los mismos (que son el tema de consultas especiales). Un sistema OLAP debe cumplir las reglas del Dr. Codd: Se tiene que tener una visión multidimensional de los datos; Pensar en dimensiones y métricas de Negocio. No en tablas y en campos

1. Profesor de tiempo completo. Universidad Simón Bolívar. Barranquilla, Colombia. Email: akamagate@unisimonbolivar.edu.co
* Este artículo es producto del proyecto: Estudio del desempeño de la Business Intelligence en las empresas Colombianas

I. INTRODUCTION

OLAP (or online analytical processing) has been growing in popularity due to the increase in data volume and the recognition of the value of business analysis. Until the mid- 1990s, the carrying out of OLAP analysis was an extremely costly process mainly restricted to large companies.

Creating the environment to carry out the OLAP analysis also required important investments in time and monetary resources.

This has changed as the leading database providers have begun to incorporate OLAP modules in their database offerings- Microsoft SQL Server 2000 with Analysis Services, Oracle Express with Darwin, and IBM with DB2.

A. Codd's Rules

In 1985 Dr. Edgar Frank Codd published 12 rules to evaluate whether a DBMS (Database Management System) could be considered an RDBMS (Relational Database Management System) or more succinctly, if a database system could be considered relational or non-relational, and more.

Rule #1: Information Rule

All information from the database must be explicitly represented in the logical schema. That is to say, all the data is in the tables.

Rule #2: Guaranteed Access Rule

Each piece of data (atomic values) of a Relational Database (RBD) is guaranteed logically accessible using a combination of table, primary key value and column name.

- We should be able to address any value stored in a RBD uniquely. This means indicating the table, column and row in which it is located (using the primary key).
- Therefore this requires the primary key concept, which is not supported in many implementations. In such cases, to achieve a similar effect we may do the following:

- Make the primary key attributes unable to be null (NOT NULL).
- Create a unique index for the primary key
- Never remove the index.

Rule #3: Systematic Treatment of Null Values

- Null values (which are different from the empty string, blanks, 0,…) are supported in completely relational DBMSs to represent unknown or not applicable information systematically, independently of the type of data.
- We recognize the need for the existence of null values, so that they may be systematically treated.
- There are problems in supporting null values in relational operations, especially in logical operations.
- Tri-valued logic. In one possible solution. Three (not two) values of truth exist: True, False and Unknown (null). Tables of truth are created for logical operations:
- null Y null = null
- True Y null = null
- False Y null = False
- True O null = True
- Etc.

Relational Database System Administrator 2

A disadvantage is that for the user the handling of relational language is then more difficult to understand.

Rule #4: Dynamic Online Dictionary Based on the Relational Model

The description of the database logic is represented in the same manner as normal data, in such a way that authorized users can apply the same relational language to their query, as they apply it to normal data.

- It is a consequence of Rule #1 that stands out for its importance. Metadata is stored using the relational model, with all the consequential actions.

Rule #5: Rule of the entire data sublanguage

A relational system must support several languages and various modes of terminal use (eg. filling in forms, etc.).

However, there must be at least one language whose statements are expressible, using syntax rather defined as character strings and that is comprehensive, supporting:

- Data Definition
- Defining views
- Data manipulation (interactive and by program)
- Integrity Constraints
- Transaction Limitations (Begin, commit, rollback).
- In addition to having friendlier interfaces for querying, etc., there should always be a way to do it totally textually, which means that it may be incorporated into a traditional program.
- SQL language accomplishes this to a great extent.

Rule #6: Rule for updating views

All views that are theoretically updatable can be updated by the system.

- The problem is to identify the theoretically updatable views, as they are not very clear.
- Each system can make some special assumptions about the views that are updatable.

Rule 7: Inserting, Updating and Deleting in High Level

The ability to handle a base or derived relationship as a single operand applies not only to the recovery of data (queries), but also to insert, update and delete data.

- That is, the data management language should also be high level (of sets). Some databases initially could only modify the tuples of the database one at a time (one record for each time).

Rule 8: Physical Data Independence

Application programs and terminal activities remain physically unaltered whenever changes are made to storage representations or access methods.

- The relational model is a logical data model, and hides the characteristics of its physical representation.
- is the ability to modify the internal schema without having to change the conceptual schema (or external ones). For example, it may be necessary to reorganize certain physical files in order to improve the performance of query operations or data updates. Physical independence refers only to the separation of applications and physical storage structures.

The ability to modify the conceptual schema without having to rewrite application programs.

Relational Database System Administrator 3

Rule #9: Logical Independence of Data

Application programs and terminal activities remain logically unaltered whenever changes are made to base tables that preserve information.

- When the logical schema is modified preserving information (for example, without removing an attribute) it is not necessary to modify anything at higher levels.

Examples of changes that preserve information:

- Adding an attribute to a base table.
- Substituting two base tables with the union thereof. Using union views, the previous tables can be recreated…

Rule # 10: Integrity of Independence

The constraints of integrity specific to a particular relational database must be able to be defined in the relational data sublanguage and storable in the catalog, not in application programs.

- The purpose of the database is not only to store data, but also their relationships and to prevent these (constraints) from being encoded in the programs. Hence we should be able to define integrity constraints in an RDB.
- The types of Integrity constraints that can be used in RBDMSs are expanding more and more, although up until recently they were very scarce.
- As part of the constraints inherent in the relational model (as form part of its definition):

- An RBD has entity integrity. That is, every table must have a primary key.
- An RBD has referential integrity. That is, all nonzero foreign keys must exist in the relationship where they are primary.

Rule No. 11: Distribution Independence

An RDB has distribution independence

- The same commands and programs run as well than in one that is centralized DB than in one that is distributed.
- RDBs are easily distributable:
- The tables are divided into fragments that are distributed.
- When needed entire tables are recombined using relational operations with fragments.
- However, the internal management of integrity, etc. is further complicated.
- This rule is responsible for three types of distribution transparency:
- Location Transparency. The user has the impression that he is working with a local DB. (an aspect of the physical Independence rule)
- Fragmentation Transparency. The user realizes that the relationship with which he is working is fragmented (an aspect of the rule of logical data independence).
- Transparency Replication. The user does not realize that there may be copies (replicas) of the same relation in different places.

Relational Databases System Administrator 4

Rule #12: Rule of Non-Subversion

If a relational system has a low level language (one record at a time), this low level cannot be used to bypass (subvert) integrity rules and constraints expressed in higher level relational languages (a relationship (set of records) at a time)

- Some problems cannot be solved directly with high-level language.
- Normally SQL is used embedded in a host language to solve these problems. It uses the cursor concept to individually address tuples in a relation. In any case it should not be possible to bypass the integrity constraints imposed when dealing with tables that level.

II. What is OLAP?

OLAP allows business users to thoroughly examine the data at will. Typically an organization's data is distributed across multiple data sources and each piece of data is not compatible with the other. A simple example: point-of-sale data and sales made through the call center or the Web are stored in different locations and formats. It would be a time consuming process for an executive to obtain OLAP reports such as: - What are the most popular products purchased by customers between the ages of 15-30?

Part of the OLAP implementation process involves the extraction of data from various data repositories and making them compatible. Making the data compatible involves ensuring that the significance of the data in a repository matches all other repositories. An example of incompatible data: client ages can be stored as date of birth of purchases made over the Internet and stored as age categories (ie., between 15 and 30) for sales in stores.

It is not always necessary to create a data warehouse for OLAP analysis. Data stored by the operating systems, like point of sale data, is found in types of databases which are called OLTPs. OLTP, Online Transaction Process, databases are no different from a structural perspective from other databases. The main and only difference is in the way data is stored.

Examples of OLTPs can include ERP, CRM, SCM, point of sale applications, and Call Centers.

OLTPs are designed for operation at optimal speeds. When a consumer makes a purchase online, they expect the transaction to occur instantaneously. With a database design, calling for data modeling, optimized for transactions ‘consumer's name, address, telephone number, order number, name of Order, price, payment method’ the log is created quickly in the database and the results can be revoked by administrators just as quickly if necessary.

Investig.Innov. Ing. 1 (2) pp. 42-48. Julio – Diciembre 2013 .Universidad Simón Bolívar. Barranquilla, Colombia. ISSN: 2344-8652

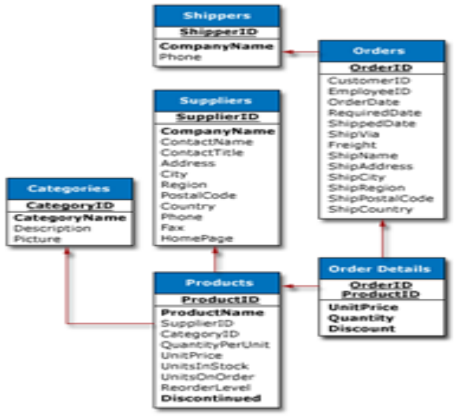


Figura 1. Modelo de datos para OLTP

Data typically is not stored for a prolonged period in OLTPs for reasons related to storage costs and transaction speed.

OLAPs have a different task from OLTPs. OLAPs are designed to provide a comprehensive analysis of what happened. Therefore, data storage (ie, data modeling) must be configured differently. The most common model is the star design.

Figura 2. Estrella Modelo de datos para OLAP

The central table in an OLAP model data home is called the fact table. The surrounding tables are called dimensions. Using the above data model, it is possible to create reports that answer questions such as:

• The supervisor who gave the most discounts.

• The amount sent on a specific date, month, year or quarter.

• In what zip code did they sell more of product A?

To obtain responses, as above, OLAP cubes are created from a data model. OLAP cubes are not strictly cuboid – it is simply the name given to the process of linking data of different dimensions. The cubes can be developed along business units such as sales or marketing in this instance. Or a giant cube can be formed with all the dimensions.

OLAP can be a valuable and rewarding business tool. In addition to the creation of reports, OLAP analysis can help an organization evaluate its scorecard objectives.

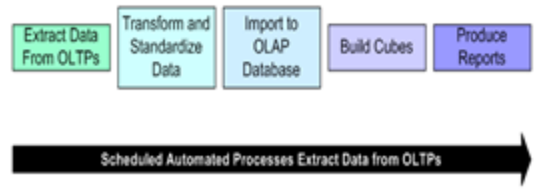


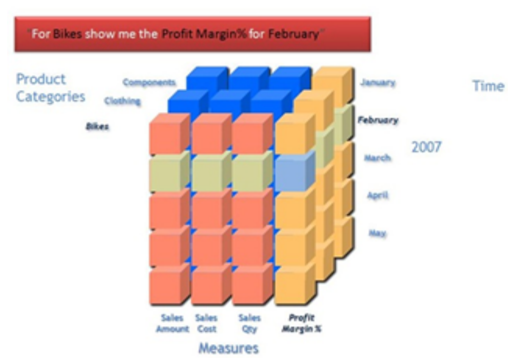
Figura 4. Pasos en el Proceso de Creación de OLAP

II. OLAP Analysis

A. Analytical Processing:

OLAP, Online Analytical Processing, is being used aggressively by organizations to discover valuable business trends of data marts and data warehouses.

OLAP provides a historical view of data, although it is useful when used by itself, OLAP analysis becomes really powerful when combined with predictive analysis of data mining.



B. OLAP Solution Design:

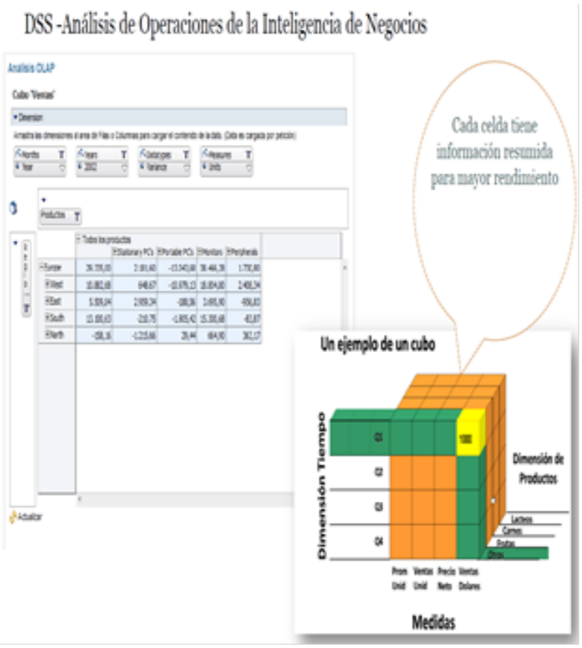
MOLAP, ROLAP, HOLAP and other acronyms.

C. Selection of an OLAP application

- Minimizing risks in the Organization's product selection process.

D. Characteristics to add to an OLAP from a practical standpoint

- It should be quick. Not too much time should pass between the need for information and the result.
- It must use functional and business language
- It must be easy to use, with



Wizards and templates

- It must be able to integrate API.
- It must have strong graphics capabilities
- It must use maps habitually
- Capability for storage and sharing reports and calculations created by users.
- Users must lead the administration,
- The implementation time (project) must be very short
- It must generate measurable responses for decision-making.
- We have to be able to obtain ROI with OLAP applications.
- Comment

III. SOME OLAP TOOLS

OLAP	Vendedor	Versión
Oracle Enterprise BI Server	11.1.1.6	Oracle
Oracle HyperionSystem	Oracle	
Microsoft BI & OLAP tools	Microsoft	2012/'10
IBM Cognos Series 10	IBM	10.1.1
QlikView	QlikTech	11
Board Management IntelligenceToolkit	Board International	7.4
SAP BusinessObjects Enterprise XI	SAP	4.0/FP3
SAP NetWeaverBI	SAP	7.3
Microstrategy	Microstrategy	9.2
JasperSoft (open source)	JasperSoft	4.7
Mondrian OLAP server	Open Source	Pentaho
Palo	Open Source	3.1.1

IV. CONCLUSION

Today there is a wide range of OLAP tools which give you the best opportunity to have more easily understandable data. So, why waste time searching for regular occurrences in your data? OLAP tools can do it for you. It really does not matter if you are a professional data analyst, or a marketing manager, or simply a stamp collector. The rule is that if you have a database it's possible that you may want to use it to respond to questions and gain useful knowledge.

V. REFERENCES

[1] E. Codd, a Relational Model of Data for Large Shared Databanks, 1970.

[2] http://www.baquia.com/posts/que-es-olap, Emilio, 2006.

[3] http://www.dwreview.com/OLAP/index.html, Hari Mailvaganam, 2007.

[4] http://www.trinus.com/

[5] http://www.Definición de OLAP – Tecnología OLAP - Significado y definición de OLAP – Tecnología OLAP.htm, Analía Lanzillotta, 2009.

[6] http://www.OLAP y el diseño de cubos GestioPolis.htm, Alfredo Martínez Orol.2008.

[7] http://social.technet.microsoft.com/wiki/contents/articles/7611.problem-processing-mining-structure-

when-datasource-is-olap-cube.aspx, Karan Gulati, 20 de febrero 2012 08:02 AM.

[8] http://www.dashboardinsight.com/articles/new-concepts-in-business-intelligence/so-just-what-is-olap.aspx, Robert Hunter, Dash board Insight, Lunes, 03 de mayo 2010.

[9] http://bakery.cakephp.org/articles/eimermusic/2009/06/04/starschema-olap-setup-for-reporting, eimermusic publicada el 18/02/10 11:01:47 AM.

[10] http://www.oracle.com/technetwork/middleware/bi-foundation/large-scale-data-warehousing-essbas-128179.pdf, Hyperion, 2000.

[11] http://thephantomwriters.com/free_content/db/p/boost-productivity.shtml, 2007-2012 John Parsons, Copyright (c).

[12] http://www.sswug.org/articles/viewarticle.aspx?id=15497, Alexzander Nepomnjashiy

[13] http://codeissue.com/articles/a04edc3096d9c7d/introduction-to-mondrian-olap-schema, Yogaraj Khanal el 04 de diciembre 2011.