OLTP and OLAP Data Integration: A Review of Feasible Implementation Methods and Architectures for Real Time Data Analysis

Samuel S. Conn
Regis University School for Professional Studies
sconn@regis.edu

Abstract

This paper elaborates on the justification and need for integration of OLTP and OLAP data environments for the purpose of real time data analysis, and reviews potential implementation methods and architectures for integration of these disparate environments. A body of relevant literature is reviewed to answer a central question on the efficacy of OLTP and OLAP integrated environments, and the central question is made tractable by its decomposition to the essential questions that must be answered regarding the feasibility of various methods and architectures for integrating data architectures. Additional coverage is given to proposed architectures and methods for real time data warehouses in the context of OLTP and OLAP data integration. This qualitative study uses an Interpretivist epistemology to reach conclusions about the literatures treatment of the topic and the justification and need for data integration to accomplish real time data analysis in the business enterprise.

1. Introduction and Foundation

Database technology is at the center of many types of information systems. Among these information systems are decision support systems and executive information systems. Online Transaction Processing (OLTP) environments use database technology to transact and query against data, and support the daily operational needs of the business enterprise. Online Analytical Processing (OLAP) environments use database technology to support analysis and mining of long data horizons, and to provide decision makers with a platform from which to generate decision making information. Decision support systems and business intelligence tools, as well as data mining analytics, generally utilize a different data structure to do their work. One of the main disadvantages to the OLAP environment is the currency of the data. Because the process by which data is extracted, transformed, and loaded into the OLAP environment can be relatively slow by transactional data standards, the ability to achieve "real-time" data analysis is lost. Kannan [21] contends that the real value of having real-time intelligence is that business processes can be optimized. The importance of generating real-time business intelligence is that it is a building block to achieve better business process management and true business process optimization.

OLTP database structures are characterized by storage of "atomic" values, meaning individual fields cannot store multi-values. They are also transaction oriented as the name implies. Alternatively, OLAP database structures are generally aggregated and summarized. There is an analytical orientation to the nature of the data, and the values represent a historical view of the entity. According to Terr [34], real-time data warehousing combines real-time activity with OLAP concepts. He discusses the problem with latency and differentiates between "real-time" and "near real-time" systems. The business goal of using real-time data analysis is further supported by Terr [34] in his discussion on the compelling reasons for building real-time data analysis systems.

1.1 Significance

The significance of this research is that organizations need real time data analysis in order to improve decision support systems, business intelligence, and business activity monitoring. The migration of the data into the data warehouse using conventional ETL tools and methods, performed on large sets of data, creates an information latency problem because of the time required to perform the ETL function and migrate the data to a separate OLAP platform. Reimers [30] points out that the need to speed up decision-making in business by using real-time analytic tools and platforms is strongly motivated by the fear of falling behind the competition. Database management systems (DBMSs) are generally implemented against data in one of two main structured data environments: the Online Transaction Processing (OLTP) environment and the Online Analytical Processing (OLAP) environment. Conventionally, a separate database from the OLTP environment is used to implement the OLAP environment [20, 7]. The standard design for OLTP schemas originated in Codd's seminal work. Codd [8] published the initial work on designing relational databases in his paper titled "A Relational Model of Data for Large Shared Data Banks". Relational databases are exceptionally well suited for transactional purposes, and have proliferated in the business enterprise as the principle technology to host and maintain operational data. The OLAP environment serves a different business purpose and is much different than the OLTP environment. Data from the transactional OLTP environment is extracted, transformed, and loaded into the OLAP environment where advanced DBMS products can utilize analytics native in the Structured Query Language (SQL) release 3 (1999) standard to provide business intelligence. The OLAP environment was designed to provide for a successful implementation of Decision Support System applications [23].

The extraction, transformation, and loading (ETL) process can be time consuming, and is generally a large determining factor in the frequency of the updates to the OLAP environment. To this extent, the data in the data warehouse can be less than useful in decision support since it is not "real time" data, such as is found in the OLTP environment. The research question is stated as "is it possible to integrate the OLTP and OLAP environments without disruption to the transactional performance so that real-time, or near real-time, data analysis may occur?"

2. The Problem

Business data processing finds that OLAP, and its reliance on the data warehousing environment, are two of the most significant new technology areas [35]. Moreover, the use of relational design and relational database technology are not feasible implementations to support OLAP design because of the complexity of the queries [35]. The business problem is that OLAP queries are not real-time queries because of the refresh cycle of data into the OLAP data repository. Conventional designs call for integration of transactional and operational data into an operational data store where it can be cleansed, transformed, extracted, and then loaded into the OLAP data repository. This is accomplished through the use of (ETL) tools. The ETL process is generally complicated because data must be integrated and transformed for loading into the non-normalized relational schema usually associated with OLAP environments. As such, the process can be complicated and time consuming, and with large amounts of data may only occur at monthly or quarterly time intervals. This creates the problem of not having real-time data in the OLAP repository. Real-time data exists in the OLTP environment where the time horizon of data within the OLTP environment is much shorter because performance decreases can occur with growing amounts of data. This is opposite of the nature and goals of the OLAP environment where data is aggregated and the time horizon of data grows to some large amount as determined by the information life cycle policy of the organization. So the advantage of the OLAP data repository is that is has a long time horizon from which to perform analysis and discover trends and patterns within the business, but the disadvantage is that data may not be (relatively) recent enough to qualify as real-time data for business intelligence purposes.

The OLTP and OLAP environments are separate and distinct because they are designed to accommodate different business purposes. The problem of integrating these two environments is multi-faceted. It includes consideration for the logical database models for the two environments, the integration of the physical data, and the database management system (DBMS) engine that runs against the data. Integration failure with any of these criteria could result in the inability to integrate the OLTP and OLAP environments. Broken down into three subordinate questions, the problem is now tractable and can be articulated as:

a. Is it possible to use one logical database model for OLAP and OLTP?
b. Is it possible to integrate the physical data residing in the OLAP and OLTP repositories?
c. Is it possible to use the same DBMS engine to query the OLAP and OLTP data?

3. Methodology

This research will be conducted as a qualitative study by drawing on relevant literature and other scholarly documentation to investigate any proposed architectures or processes for integrating the OLTP and OLAP environments. The methodology will be compatible with an interpretivist epistemology using action research to acquire critical information followed by active reflection and subsequent changes to the research method. This iterative cycle will be used in the secondary literature searches. The review and notation of the relevant literature will constitute the data collected for analysis in the study, and through a hermeneutic approach the analysis of the findings will attempt to answer the pending research question. This approach is justified because this epistemology and methodology have been proven to be effective in Information Systems research and has been noted to be increasingly used in current publications [24].

Prior Research - The motivating literature for this study centers on four things: OLAP as a decision support system environment, data warehousing as a main architecture for OLAP systems, the SQL language as an analytic tool, and the integration of the OLAP and OLTP environments. Specifically, Jones [19] points out that there are multiple reasons to separate the OLTP and OLAP environments. Chief among them are the operational characteristics of the data environments. OLTP environments serve the operational and transactional needs of the organization, are generally designed on relational principles, and manage shorter time horizons of data. On the other hand, OLAP environments serve the decision support needs of the organization, have denormalized relational designs, and manage much longer time horizons of data. Bontempo and Zagelow [2] point out the distinctions between the OLTP and OLAP environments as being related to application orientation versus subject orientation, limited data integration versus full data integration, volatile data versus non-volatile data, and ad-hoc data retrieval versus predictable data retrieval. Moreover, Gray and Watson [14] compare the various aspects of the data warehouse (or OLAP) topology to the OLTP environment. Included in the comparison is data structures used, the form of data, the flow of data, and the use of OLAP architectures for decision support and knowledge management. Jones [19] notes the problem with using OLAP systems for real-time data analysis is the inherent latency in the ETL process. A second issue is the reformulation of multi-dimensional data cubes for analysis by business intelligence and data mining tools. So the discussion of integration of the OLTP and OLAP environments occurs in two distinct formulations, one being the use of the SQL language in a relational engine, and the other being the redesign of the topology and architecture. According to Eisenberg and Melton [11], the SQL:1999 standard has object extensions and capability, and also includes the analytical functions needed to summarize, group, and aggregate data for decision support applications. Hence, SQL/OLAP was the first amendment to the SQL:1999 standard and allows any relational SQL engine implemented against the standard to function as both a transactional data engine and a data warehousing and data mining engine. According to Riedel, Faloutsos, Ganger, and Nagle [31], it is technically feasible to data mine against an OLTP environment. And alternatively, there is literature such as Martyn [22] that advocates the dissolution of disparate environments to achieve integration of the environments. Based on the work of C. J. Date [9], Martyn [22] suggests that there is no need for multi-dimensional data structures.

3.1 Additional Literature Review

Davenport and Prusak [10] provide a good framework for defining the value proposition between data, information, and knowledge. They define data as a set of discrete, objective facts about events. Migrating upwards in the value proposition, they define information as "data that makes a difference". When investigating the distinctions between OLTP and OLAP systems, Thomas and Datta [35] provide good foundation for identifying the complexities of integrating the two environments. They note the difficulty with using standard relational technology, as found in OLTP systems, to support the OLAP environment. Roussopoulos [32] develops the idea of using materialized views as a derivative concept of relational systems to support the multi-dimensional viewing of data within relational structures. This provides evidence that researchers are looking for ways to integrate the two environments without suffering performance issues in the operational environment. Other approaches noted in the literature include Schallehn, Sattler, and Saake's [33] work on utilizing grouping and aggregation functions to summarize and store data in operational systems for analytical processing. This is yet another proposed method to get around the problem of integration of short and long data horizons as found in OLTP and OLAP systems (respectively). Some researchers [5] have tried to develop a relationship between relational and multi-dimensional query capabilities. Relational Online Analytical Processing (ROLAP) can use conventional SQL query capability, but Multi-dimensional Online Analytical Processing (MOLAP) requires object extension in the SQL to perform queries. These researchers present the idea that ROLAP and MOLAP environments can be mapped together "virtually" using an Object-Oriented DBMS. With yet another approach, Gray and Watson [14] suggest that maybe a solution can be found with integrating real-time data at the data mart level. Although they are careful to point out that this approach can lead back to the problem of silos of information within the organization, they do offer the data mart as a potential future direction for incorporation of volatile data.

Some interesting action research that contributed to this field of study was introduced by Hasan, Hyland, Dodds, and Veeraraghavan [15] and builds on the 1980s work of Brancheau and Wetherbe [3] who developed the concept of a unified logical corporate data model. Hasan, Hyland, Dodds, and Veeraraghavan [15] contend that relational databases and multi-dimensional databases are not competing but complimentary database architectures. This contention provides an interpretation of how ROLAP and MOLAP are legitimate methods of presenting a logical, consolidated view of OLTP and OLAP data stores via a graphical user interface. Although they were not specifically addressing the problem of integration with OLTP and OLAP environments, their work approaches the problem with an integration solution based on the use of multi-dimensional data structures in operational environments. A final approach considered in the literature is the use of XML technology and federated architectures. Hummer, Bauer, and Harde [18] describe the use of the XML standards developed by the Object Management Group (OMG) to store multi-dimensional data in XML format. By using XML as an integration point for both relational and multi-dimensional data, the convergence of historical OLAP data with real-time OLTP data in a federated repository is feasible. Niemi, Nummenmaa, Niinimaki, and Thanisch [25] further this concept by suggesting a method for using distributed XML data (from OLTP and OLAP systems) to construct an "on the fly" OLAP cube for query. More precise federated architectures and query optimization methods are introduced by Pedersen, Riis, and Pedersen [28] with the continued focus on logical, rather than physical, integration of the data. Moreover, Golfarelli, Rizzi, and Vrdoljak [13] contend that XML will become the primary mode of loading into OLAP environments, and as such has the potential to accomplish data structure mapping between the OLTP and OLAP environments for the purpose of query. Also of note are the works presented by Pedersen and Pedersen [28], and Cannataro, Guzzo, and Puliese [6] on using XML for the logical integration of external data in OLAP environments.

4. Results and Findings

4.1 Is it possible to use one logical database model for OLAP and OLTP?

The logical and physical database design for OLTP and OLAP environments are not compatible. According to Thomas and Datta [35], the separate requirements for the two environments dictate different data models and implementation methods for each type of system. The compromises that must be made are generally not acceptable to users of either environment. The normalized, relational database model used for the OLTP environment is designed for the elimination of insertion, deletion, and update anomalies, and indexed and optimized for speed of queries and transactions. However, the queries that run in a data warehouse can often deplete the resources to an extent that would cripple the response time of the OLTP system. Barbusinski [1] points out that a one size fits all approach to database design never works. Howard [16] tempers this view by dividing the issue of a single database and a single database model to support both environments. According to Howard [16], a single database engine could handle the parallel processing of both environments, but a single database model to support both environments is not feasible. Even so, he goes on to point out that it is not "recommended" because the workloads in the two environments are so different. The notion of the workloads being extremely different is supported by Oates [26] who believes that two separate databases and database servers are always needed to separate the transaction and analytical systems. Rehm [29] presents an interesting point in that even if one data model and one database were implemented to support both environments, in the real business world the constant demand for integration of new data from business acquisitions and mergers would present another layer of design complexity that may not be accommodated in the design.

4.2 Is it possible to integrate the physical data residing in the OLAP and OLTP repositories?

Howson [17] points out the differences between ROLAP (Relational Online Analytical Processing) and MOLAP (Multi-dimensional Online Analytical Processing) and the impact each has on the physical data structures needed to support them. Accordingly, the physical structure of data in the OLAP environment is not in 3rd normal form because of the need to reduce the number of join conditions in query processing. ROLAP allows the data to remain in 3rd normal form as in an OLTP system because it uses relational queries. However, the slow performance of ROLAP is a major drawback. Alternatively, MOLAP can be used to speed up the OLAP query performance because the data is constructed in a persistent dimensional data cube that is separate from the transactional data. According to Howson [17], with the query speed increase of MOLAP comes two main problems. The first is that the physical data must reside apart from the transactional data because the cube is created and stored as dimensional data. The second is the lack of flexibility if any dimension of the data were to be changed. This requires the entire MOLAP cube to be reconstructed. An alternative, according to Howson [17], is the implementation of DOLAP (Dynamic Online Analytical Processing). In this case the smaller cubes of data that support departmental level needs can be dynamically constructed as needed, and do not require a separate data store for housing.

4.3 Is it possible to use the same DBMS engine to query the OLAP and OLTP data?

The SQL3 standard introduced the ability to work with multidimensional objects and DBMS vendors also advanced the ability to work with materialized views. So even though ROLAP was possible using a relational SQL engine, the MOLAP applications were not available until DBMS vendors began to implement against the SQL3 standard. So it is currently possible to utilize the same DBMS engine for both OLTP and OLAP applications, but the use of a single engine is not really the road block to successful design of a real-time integrated data warehouse environment.

Further findings are that the concept of the real-time data warehouse (RTDW) is giving birth to a third generation of data warehouse design. The four potential architectures are as follows:

Simulated real-time and trickle feed architectures – these architectures effectively work to speed up the ETL process. According to Gadodia [12], in the simulated real-time architecture the updates to the OLAP environment are batch written at more frequent intervals than usual. This increased frequency of updates has the net effect of integrating OLTP real-time data with historic OLAP data. Thus it has a "simulated" real-time effect. Gadodia [12] also notes that the "trickle feed" architecture opens a persistent connection (pipe) between the two environments. Data can then be streamed through the pipe in order to replicate the OLTP data to the OLAP repository in a near real-time mode. Some type of message queuing technology is used to feed the OLAP data repository. Both of these methods populate the data warehouse (OLAP) environment on faster cycles than a normal ETL process, and produce the effect of real-time data in the OLAP repository. Terr [34] points out that ETL tools have advanced to a state of maturity and that current reporting tools enable data analysis without latency problems or cross platform issues. The concept of data "freshness" is discussed by Brobst [4]. Freshness, or currency, of data is an imperative in new OLAP system deployments.

Active disks – the "active disk" concept takes advantage of free blocks that are available to be read during the "seeks" required for OLTP transaction processing. According to Riedel, Faloutsos, Ganger, and Nagle [31], the active disk concept works well with SAN environments where there is physical separation of the data from the server hosting the DBMS. More efficient management of bandwidth and I/O is the key to the active disk concept.

Materialized views – materialized views offer potential as a relational specification technique to perform many "materializations" of data for the purpose of OLAP applications. Roussopoulos [32] offers some good insight into the use of materialized views for OLAP applications. He maintains that materialized views are multi-faceted, meaning they can be used to create snapshots of data, data indices, and data summaries and aggregations. Roussopoulos [32] contends that materialized views are underutilized in their potential to support OLAP. He advocates the "re-use" of views by the DBMS to save processing cost and time.

eXtensible or eXtended Markup Language (XML) technology - Hummer, Bauer, and Harde [18] provide an interesting look at the use of XML to form the data warehouse structures. As a form of semi-structured data, XML can be used as a merge point for both structured and un-structured data. This offers enormous potential to the enterprise that wants to aggregate the two forms of data in the design of knowledge management systems. What it means for the integration of OLTP and OLAP data is that it can be used as a transport mechanism and data storage structure. Pedersen, Riis, and Pedersen [27] propose the use of XML to create OLTP "federations" where federated structures are used to integrate OLTP data and OLAP data via XML. A continuation of this logical design is presented by Niemi, Nummenmaa, Niinimaki, and Thanisch [25], who develop the idea of using distributed XML sources to construct multi-dimensional OLAP cubes for query. These constructs, however, take the technology outside of the relational theory bounds of using SQL as a primary query language into the use of XML query languages. As a part of the next iteration of the ANSI SQL standard, XML query is still a proprietary technology that is generally not supported by SQL3 relational engines. However, the logical design for using semi-structured data as a point of OLTP and OLAP data integration is supported by Golfarelli, Rizzi, and Vrdoljak [13] in their work on designing data warehouses from XML sources. Their conceptual design serves as a blueprint for the use of XML technology in developing integration plans for OLTP and OLAP data.

5. Conclusion

The contributions of this research are discussion and analysis of the three related questions about the efficacy of OLTP and OLAP integration, the identification of the major research discussions available in the relevant literature, and the articulation of four major approaches to the design and development of real-time, or near real-time, data analysis. The conclusion to the first question is that one data model cannot serve the needs of both an OLTP and an OLAP environment. This author concludes that the answer to the second question is that it is becoming increasingly feasible to integrate OLTP and OLAP data either through storage area network zones, or through logical constructs such as materialized views. And the conclusion to the third question is that with the object extensions of the SQL3 standard, it is possible to use the same database engine for both OLTP and OLAP applications. This paper further contributes by identifying four major approaches to the concept of real time data warehousing (RTDW). The limitation of this study is the availability of performance metrics on an implemented system. To date, there is little literature published on performance testing against standard metrics for query of OLAP systems using any of the four design concepts. As the importance of real-time data warehousing and analysis increases, more research and investigation into OLTP and OLAP integration design should produce additional data. Key database vendor initiatives, such as the Teradata University, the Oracle Technology Network, and the IBM Scholars Program are beginning to offer published materials on this topic, but have yet to produce definitive work in this area.

6.0 References

[1] Barbusinski, L. (2002, May). Can You Use One Database Model for OLAP and OLTP? *DM Review*.

[2] Bontempo, C., & Zagelow, G. (1998). The IBM Data Warehouse Architecture. *Communication of the ACM, 41*(9).

[3] Brancheau, J. C., & Wetherbe, J. C. (1986). Information Architectures: Method and Practice. *Information Processing and Management, 22*(6), 453-464.

[4] Brobst, S. (2002). Delivery of Extreme Data Freshness with Active Data Warehousing. *Journal of Data Warehousing, 7*(2).

[5] Buzydlowski, J. W., Song, I.-Y., & Hassell, L. (1998). *A Framework for Object-Oriented On-Line Analytic Processing.* Paper presented at the DOLAP '98, Washington DC.

[6] Cannataro, M., Guzzo, A., & Pugliese, A. (2002). Knowledge Management and XML: Derivation of Synthetic Views Over Semi-Structured Data. *ACM SIGAPP Applied Computing Review, 10*(1), 33-36.

[7] Chaudhuri, S., & Dayal, U. (1997). An Overview of Data Warehousing and OLAP Technology. *SIGMOD Record, 26*(1), 65-74.

[8] Codd, E. F. (1970). A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM, 13*(6), 377-387.

[9] Date, C. J. (2000). *An Introduction to Database Systems* (7th ed.). Boston, MA: Addison-Wesley.

[10] Davenport, T. H., & Prusak, L. (1998). *Working Knowledge: How Organizations Manage What They Know*. Boston, MA: Harvard Business School Press.

[11] Eisenberg, A., & Melton, J. (2000). SQL Standardization: The Next Steps. *SIGMOD Record, 29*(1), 63-67.

[12] Gadodia, R. (2004). Right in Time. *Intelligent Enterprise, 7,* 26-44.

[13] Golfarelli, M., Rizzi, S., & Vrdoljak, B. (2001). *Data Warehouse Design from XML Sources*. Paper presented at the DOLAP '01, Atlanta, GA.

[14] Gray, P., & Watson, H. J. (1998). Present and Future Directions in Data Warehousing. *The DATA BASE for Advances in Information Systems, 29*(3), 83-90.

[15] Hasan, H., Hyland, P., Dodds, D., & Veeraraghavan, R. (2000). Approaches to the Development of Multi-Dimensional Databases: Lessons from Four Case Studies. *The DATA BASE for Advances in Information Systems, 31*(3).

[16] Howard, S. (2002, May). Can You Use One Database Model for OLAP and OLTP? *DM Review*.

[17] Howson, C. (2004, May 15). BI Scorecard: OLAP. *Intelligent Enterprise, 7,* 28-38.

[18[Hummer, W., Bauer, A., & Harde, G. (2003). *XCube - XML for Data Warehouses*. Paper presented at the DOLAP '03, New Orleans, LA.

[19] Jones, K. (1998). An Introduction to Data Warehousing: What Are the Implications for the Network? *International Journal of Network Management, 8*, 42-56.

[20] Kalnis, P., & Papadias, D. (2001). Proxy-Server Architectures for OLAP. *ACM SIGMOD 2001*.

[21] Kannan, N. (2004). *Real-Time Business Intelligence – Building Block for Business Process Optimization*. DM Review. Retrieved July 9, 2004, 2004, from the World Wide Web: http://www.dmreview.com/editorial/newsletter_article.cfm?nl=dmdirect&articleId=1002228&issue=20013

[22] Martyn, T. (2004). Reconsidering Multi-Dimensional Schemas. *SIGMOD Record, 33*(1), 83-88.

[23] McCulley, M. B. (1997). *Effective Virtual Design of Multi-Dimensional Data Models and Interfaces.* Paper presented at the Workshop on New Paradigms in Information Visualization and Manipulation, Las Vegas, NV.

[24] Myers, M. D. (1997). Qualitative Research in Information Systems. *MIS Quarterly, 21*(2), 241-242.

[25] Niemi, T., Nummenmaa, J., Niinimaki, M., & Thanisch, P. (2002). *Constructing an OLAP Cube from Distributed XML Data.* Paper presented at the DOLAP '02, McLean, VA.

[26] Oates, J. (2002, May). Can You Use One Database Model for OLTP and OLAP? *DM Review*.

[27] Pedersen, D., Riis, K., & Pedersen, T. B. (2002). *Query Optimization for OLA-XML Federations.* Paper presented at the DOLAP '02, McLean, VA.

[28] Pedersen, D., & Pedersen, T. B. (2003). *Achieving Adaptivity for OLAP-XML Federations.* Paper presented at the DOLAP '03, New Orleans, LA.

[29] Rehm, C. (2002, May). Can You Use One Database Model for OLTP and OLAP? *DM Review*.

[30] Reimers, B. D. (2003). *Real-Time Data: Too Much of a Good Thing?* ComputerWorld. Retrieved July 6, 2004, 2004, from the World Wide Web: http://www.computerworld.com/databasetopics/data/story/0,10801,80203,00.html

[31] Riedel, E., Faloutsos, C., Ganger, G. R., & Nagle, D. F. (2000, May 14-19). *Data Mining on an OLTP System (Nearly) for Free.* Paper presented at the International Conference on Management of Data, Dallas, Texas.

[32] Roussolpoulos, N. (1998). Materialized Views and Data Warehouses. *ACM SIGMOD 1998, 27*(1).

[33] Schallehn, E., Sattler, K.-U., & Saake, G. (2001). *Advanced Grouping and Aggregation for Data Integration.* Paper presented at the CIKM 2001, Atlanta, GA.

[34] Terr, S. (2003). *Real-Time Data Warehousing: Real-Time Data Warehousing 101*. DM Review Online. Retrieved July 5, 2004, 2004, from the World Wide Web: http://www.dmreview.com/article_sub.cfm?articleId=7524

[35] Thomas, H., & Datta, A. (2001). A Conceptual Model and Algebra for On-Line Analytical Processing in Decision Support Databases. *Information Systems Research, 12*(1), 83-102.