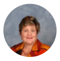
ETL Evolution for Real-Time Data Warehousing

Kamal Kakish
Georgia Gwinnett College
17 PUBLICATIONS **34** CITATIONS



Theresa A Kraft
University of Michigan-Flint
10 PUBLICATIONS **25** CITATIONS

ETL Evolution for Real-Time Data Warehousing

Dr. Kamal Kakish
kkakish@ggc.edu
School of Science and Technology
Georgia Gwinnett College
Lawrenceville, GA 30043 USA

Dr. Theresa A. Kraft
thkraft@umflint.edu
School of Management
University of Michigan – Flint
Flint, MI 48502 USA

Abstract

Informed decision-making is required for competitive success in the new global marketplace, which is fraught with uncertainty and rapid technology changes. Decision makers must adjust operational processes, corporate strategies, and business models at lightning speed and must be able to leverage business intelligence instantly and take immediate action (Oxford Economics, 2011). Sound decisions are based on data that is analyzed according to well-defined criteria. Such data typically resides in a Database Warehouse for purposes of performing statistical and analytical processing efficiently.

Data warehouses (DWH) are typically designed for efficient processing of read only analysis queries over large data, allowing only offline updates at night. The current trends of business globalization and online business activities available 24/7 means DWH must support the increasing demands for the latest versions of the data. Real-Time (or Active) Data Warehousing aims to meet the increasing demands of Business Intelligence for the latest versions of the data (Athanassoulis, et al., 2011).

Achieving Real-Time Data Warehousing is highly dependent on the choice of a process in data warehousing technology known as Extract, Transform, and Load (ETL). This process involves: 1) Extracting data from outside sources; 2) Transforming it to fit operational needs; and 3) Loading it into the end target (database or data warehouse). Not all ETL's are equal when it comes to quality and performance. As such, optimizing the ETL processes for real time decision making is becoming ever increasingly crucial to today's decision-making process. An effective ETL leads to effective business decisions and yields extraordinary decision-making outcomes. This study overviews the theory behind ETL and raises a research vision for its evolution, with the aim of improving the difficult but necessary data management work required for the development of advanced analytics and business intelligence.

Keywords: Extract, Transform, Load, ETL, Data Warehouse Loading, Real-Time, Business Intelligence

1. INTRODUCTION

With the explosive advent of the Internet, broadband communication, mobile computing, and access to cloud computing, the past couple of decades gave a new meaning to the phrase "information overload". Companies need to consider how to adopt and utilize real-time data and information into the fabric of their decision-making or risk falling behind their competitors

(Oxford Economics, 2011). Indeed, this is an era of unprecedented data copiousness and accumulation. The utter variety of new information available on diverse platforms of electronic data sources has changed the way we live, collaborate, conduct business, undertake research, and make decisions; however, the increased reliance upon networked data has introduced unique data quality challenges.

Organizations demand for quick access to new insights has led to predictive analytics for forecasting emerging demands, risks and opportunity. Advanced analytics apply statistical and predictive algorithms to forecasting, correlation, and trend analysis. In contrast, traditional data Warehousing and Business Intelligence have typically been associated with historical analysis and reporting. Advanced statistical and predicative analysis takes advantage of the large data sets (big data) stored within data warehouses to foresee risk, anticipate customer demand, and formulate more successful product and service offerings (Henschen, 2011). The advanced analytics are highly dependent on access to the most recent, real-time business data; hence, data warehouses must have instantaneous real-time access to business transactions. As the advanced analytics requirements get closer to real-time, the software applications must tolerate some amount of data incompleteness or inaccuracy, as it is not financially or technically feasible to provide 100% of the data within such strict time requirements (Schneider, D., 2007).

The rest of this paper is organized as follows: In the next section, we give a brief architecture overview of traditional data warehouses. Then the importance of data quality, industry perspectives, and challenges affecting data management are reviewed in section II. In Section III we explore the Extract, Transformation, Load (ETL) processes and raise a research challenge to the generality and limitations of traditional ETL approaches. The Evolution of ETL and advances in technology for achieving Real-Time ETL and Data warehousing are covered in Sections IV and V. Real-Time Business Intelligence Techniques are reviewed in Section VI. Lastly, we conclude by describing the current industry status and future work.

2. DATA WAREHOUSE ARCHITECTURE

For most organizations, managing data takes on two predominant forms: 1) operational systems to deal with highly capacious transactional processing using real time data, and 2) data warehouses to facilitate information access by providing a centralized database for all enterprise data organized in a manner specifically for querying (Kimball, et al., 1998). Data could be arranged in subject areas important to the organization and provide a static, consistent view of information, traits not found in the operational systems capturing business transactions (Kimball, et al., 1998; Loshin, 2003).

A data warehouse, according to W. H. Inmon, is a "subject-oriented, integrated, time varying, non-volatile collection of data in support of the management's decision-making process" (Inmon, 1996). The corporate data warehouse provides strategic information to support decision-making (Kimball, et al., 1998). High quality data may be the most important factor for data warehouse success (Loshin, 2003), while poor data quality may have unfavorable effects on this decision making (Huang, Lee and Wang 1999, Clikeman 1999).

Conceptually, a data warehouse (DWH) collocates data from diverse heterogeneous sources organized into a centralized repository for reporting and analysis purposes. The data stored in the DWH are uploaded from the transactional and operational systems (such as marketplace, sales etc., shown in figure 1). The data may pass through an operational data store for additional operations before they are used in the Data Warehouse for reporting (Conn, S., 2005).

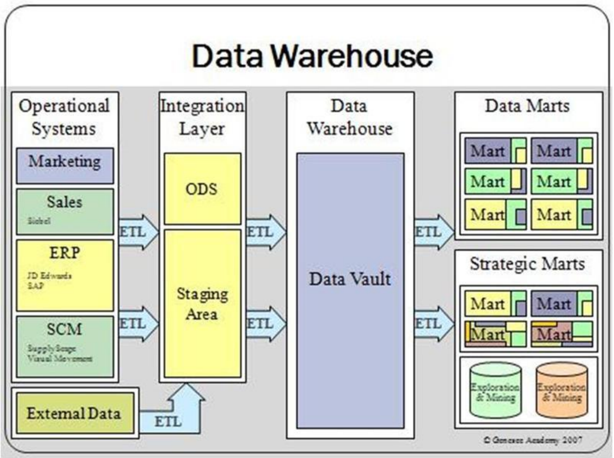


Figure 1: Data Warehouse Architecture
(http://en.wikipedia.org/wiki/Data_warehouse)

ISSN: 2167-1508
v5 n2214

3. DATA QUALITY ISSUES

A research program conducted in May 2011 by Bloomberg BusinessWeek Research Services of 930 business professionals identified data quality, integrity, and consistency as the largest challenge facing companies in their adoption of business intelligence and analytics (See Figure 2).

An InformationWeek survey of over 300 BI and Information managers identified access to relevant, timely and reliable data as the highest major impediment to information management success as shown in Figure 3. Data cleansing and consistency was ranked as the second biggest problem (Henschen, D., 2010).

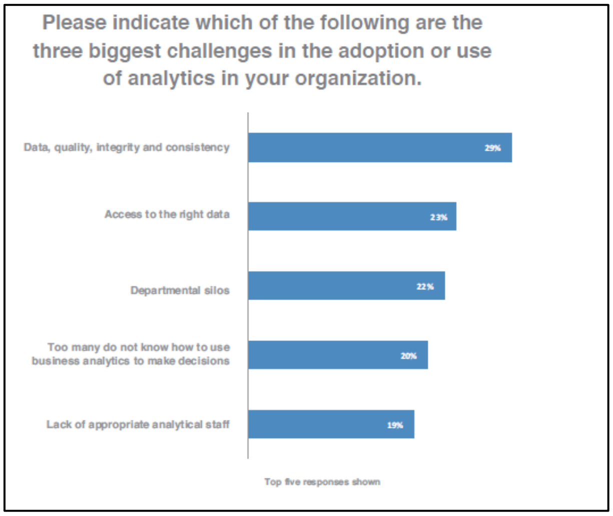
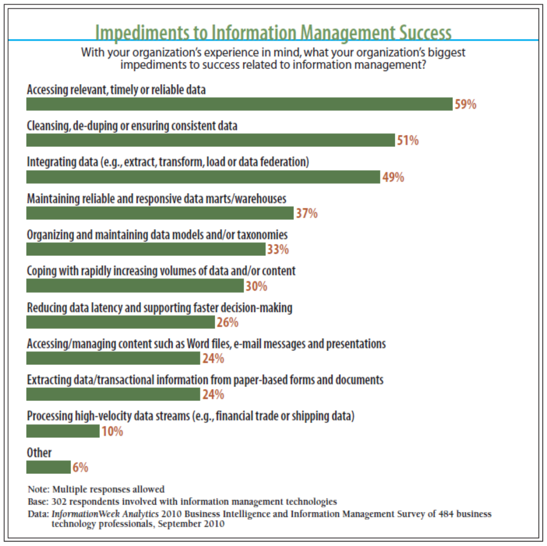


Figure 2: Business Intelligence / Analytics Roadblocks
(Bloomberg BusinessWeek Research Services, 2011)

Data quality issues range the gamut and include complexities of data completeness, consistency; validity/correctness, conformity, accuracy, and Integrity/trustworthiness (See Figure 4).

The definition for each Data Quality Dimension is as follows:

1. Completeness –The completeness of data shows or ensures the extent to which the expected attributes of data are provided and all required information is available. It is imperative to note that although the data may not be available, it could still be considered completed. This happens when the data meets the expectations of the user.



Note: Multiple responses allowed
Base: 302 respondents involved with information management technologies
Data: *InformationWeek Analytics* 2010 Business Intelligence and Information Management Survey of 484 business technology professionals, September 2010

Figure 3: Impediments to Information Management Success
(Henschen, D., 2010)

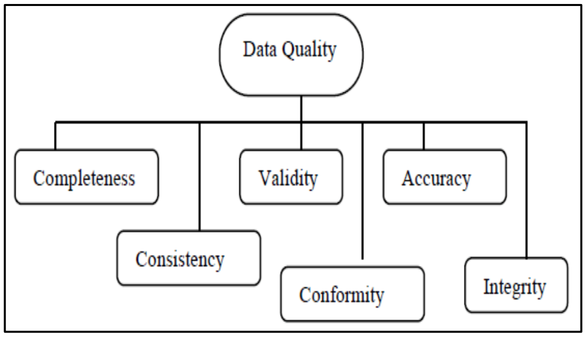


Figure 4: Data Quality Dimensions
(Singh, R., Singh, K., 2010)

2. Consistency – In order for data to be consistent, data across the enterprise should be harmonious with each other with values not having any conflicts across all data sets.

3. Validity – This refers to the correctness and reasonableness of data.

4. Conformity – Data conformity means that data values are consistent across specific formats. Maintaining conformance to specific formats is important.

5. Accuracy – Data is said to be accurate if the data correctly reflects the real world object or an event being described. For example, incorrect spellings of product or person

ISSN: 2167-1508
v5 n2214

names, addresses, and even untimely or not current data can affect operational and analytical applications.

6. Integrity – The integrity of data refers to the trustworthiness of the data. If data is missing important relationship linkages and is unable to link related records together, then it may actually introduce duplication across all systems. (Singh, R., Singh, K., 2010)

Additional factors concerning data quality include confidentiality, availability, and security. Confidentiality ensures that data is safe from unauthorized use and availability is providing the information requested or required by the authorized users in a timely manner (Jesan, J., P., 2006). Data and information in its various forms is arguably the most important asset of any organization, and data security policies must protect these information assets (Gerber, M., Von Solms, R., 2008)

An extensive research study by Byron Evans examines three data management techniques that can improve data quality. These techniques include data cleansing, information stewardship, and metadata management, which is the administration of the names, definitions, logical and physical data structures, and other data about the organization's data resources (Evans, B. 2005).

4. THE ETL PROCESS

Within an enterprise there are large quantities of diverse applications and data sources which have to be integrated. On-line transaction processing (OLTP) and data warehouses cannot coexist efficiently in the same database environment since the OLTP databases maintain current data in great detail whereas data warehouses deal with lightly aggregated and often globally reconciled historical data. OLTP emphasizes efficiency of short update transactions each covering a small part of the database, whereas data warehousing requires long queries surveying a large part of the database (Jarke, M., Vassilliou, V., 1997).

The ETL process extracts the data from source systems, transforms the data according to business rules, and loads the results into the target data warehouse as shown in Figure 5. Detailed ETL activities include (a) the identification of relevant information at the source side; (b) the extraction of this information; (c) the customization and integration of the information coming from multiple sources into a common format; (d) the cleaning of the resulting data set on the basis of database and business rules; and (e) the propagation of the data to the data warehouse and/or data marts (Bergamaschi, S., et al. 2011).

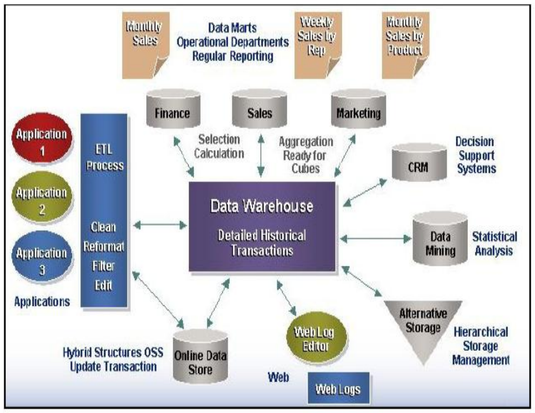


Figure 5: Technical Architecture of ETL
(Tank, D. M., et al. 2010)

The quality of data and the effectiveness of a data warehouse are directly dependent on the efficiency of the ETL process. Hence, a quality ETL process begets quality decision-making power. Research has found that seventy percent (70%) of the software implementation and maintenance effort of data warehousing is spent on the ETL system (Behrend, A., Jörg, T., 2010). The underlying traditional approaches to ETL are based are the propagation of true data updates and not incremental loads.

Extract:

Taking out the data from a variety of disparate source systems correctly is often the most challenging aspect of ETL, as it sets the stage for how subsequent processes will go. In general, the goal of the extraction phase is to convert the data into a single format which is appropriate for transformation processing. Each separate system may also use a different data organization/format. Common data source formats are relational databases and flat files, but may include non-relational database structures such as Information Management System (IMS) or other data structures such as Virtual Storage Access Method (VSAM) or Indexed Sequential Access Method (ISAM), or even fetching from outside sources such as through web-spidering or screen-scraping (http://en.wikipedia.org/wiki/Extract,_transform_,_load).

Most of the time the data in source system is very complex, thus determining which data is relevant is very difficult. Designing and creating extraction processes is a very time consuming programming effort. To keep the data up to date in data warehouse, data has to be extracted several times in a periodic manner. There are two logical methods for extraction: *Full extraction* and *Incremental extraction.*

1. *Full extraction*: In this type of extraction data from the source system is completely extracted. As full extraction extracts all the data from source systems there is no need to keep track of changes made in the source system with respect to previous extraction.

2. *Incremental extraction*: In this type of extraction only the changes made to the source systems will be extracted with respect to the previous extraction. Change data capture (CDC) is mechanism that uses incremental extraction. Most ETL tools do not use this CDC mechanism, instead they extract the entire tables from source systems in the staging area and compare these tables with the data or tables extracted from previous extraction to identify the changes. This comparison results in a large performance impact on data warehouse ELT processes (Jörg, T., Deßloch, S., 2008).

There are two physical methods of extraction: *Online extraction* and *Offline extraction*

1. *Online extraction:* The extraction process of ETL connects to source system to extract the source tables or store them in a pre-configured format in intermediary systems e.g., log tables.

2. *Offline extraction:* The data extracted is staged outside the source systems (ETL Data Extraction Methods - Part Two).

The process of physically moving data from one system to another system is also part of the extraction process, which is referred to as transportation and includes:

1. Moving data from source system to data warehouse
2. Moving data from staging database to data warehouse
3. Moving data from data warehouse to data mart

Transportation could be performed using flat files mechanism, distributed operations mechanism, or using transportable table spaces.

Transform:

The transform stage applies a series of rules or functions to the extracted data from the source to derive the data for loading into the end target. Some data sources will require very little or even no manipulation of data. Other cases require one or more of the following transformation types:

1. Selecting only certain columns to load ,
2. Translating coded values (e.g., if the source system stores 1 for male and 2 for female, but the warehouse stores M for male and F for female);
3. Encoding free-form values (e.g., mapping "Male" to "1");
4. Deriving a new calculated value (e.g., sale amount = qty * unit price);
5. Sorting;
6. Joining data from multiple sources (e.g., lookup, merge) and duplicating the data;
7. Aggregation;
8. Generating surrogate-key values;
9. Transposing or pivoting (turning multiple columns into multiple rows or vice versa);
10. Splitting a column into multiple columns;
11. Disaggregation of repeating columns into a separate detail table;
12. Lookup and validate the relevant data from tables or referential files for slowly changing dimensions; and
13. Applying any form of simple or complex data validation. (http://en.wikipedia.org/wiki/Extract,_transform,_load).

Load:

This phase loads the data into the end target, usually the data warehouse. Depending on the requirements of the organization, this process varies widely. Some data warehouses may overwrite existing information with cumulative information; frequently updates with extracted data are performed on hourly, daily, weekly, or monthly basis. The timing and scope to replace or append updates are strategic design choices dependent on the time available and the business needs.

As the load phase interacts with a database, the constraints defined in the database schema — as well as the triggers activated upon with the data load (for example, uniqueness, referential integrity, mandatory fields), will contribute to the overall data quality performance of the ETL process.

Mechanisms to load a DWH include:

1. SQL loader: generally used to load flat files into data warehouse.
2. External tables: this mechanism enables external data to be used as a virtual table which can be queried and joined before loading into the target system.
3. Oracle Call Interface (OCI) and direct path Application Programming Interface (API): are methods used when the transformation process is done outside the database.
4. Export/Import: this mechanism is used if there are no complex transformations and data can be loaded into target data warehouse system as it is (Oracle, 2005).

The typical ETL-based data warehouse uses staging, integration, and access layers to house its key functions. The staging layer or staging database stores raw data extracted from each of the disparate source data systems. The integration layer integrates the disparate data sets by transforming the data from the staging layer often storing this transformed data in an operational data store (ODS) database. The integrated data is then moved to yet another database, often called the data warehouse database, where the data is arranged into hierarchal groups often called dimensions and into facts and aggregate facts.

ETL plays an important role in data warehousing architecture since these ETL processes move the data from transactional or sources systems to data staging areas and from staging areas into the data warehouse. Demands for real-time data warehousing result from the recent trends of business globalization, 24x7 operations, ever increasing data volumes, competitive pressures, demanding customers and increased demand by decision makers for real-time data. (Ankorion, I., 2005). These current trends require business to have access to the most updated data for analysis and statistical purposes, which necessitates a requirement for building real-time data warehousing and ETL.

Techniques to achieve real-time data warehousing include the Change Data Capture (CDC) technique and the integration of change data capture with existing ETL processes to maximize the performance of ETL and achieve real time ETL (Jörg, T., Deßloch, S., 2008). The CDC integration with existing ETL tools provides an integrated approach to reduce the amount of information transferred while minimizing resource requirements and maximizing speed and efficiency (Tank, D., M. et al., 2010). In contrast, migrating the data into data warehouse using conventional ETL tools has a latency problem with the large volumes of data sets because ETL processes consume substantial CPU resources and time for large data sets.

5. EVOLUTION OF ETL

With the evolution of Business intelligence, ETL tools have undergone advances and there are three distinct generations of ETL tools.

The First-generation ETL tools were written in the native code of the operating system platform and would only execute on the native operating system. The most commonly generated code was COBOL code because the first generation data was stored on mainframes. These tools made the data integration process easy since the native code performance was good but there was a maintenance problem.

Second generation ETL tools have proprietary ETL engines to execute the transformation processes. Second generation tools have simplified the job of developers because they only need to know only one programming language i.e. ETL programming. Data coming from different heterogeneous sources should pass through the ETL engine row by row and be stored on the target system. This was a slow process and this generation of ETL programs suffered from a high performance overload.

Third Generation ETL tools have a distributed architecture with the ability to generate native SQL. This eliminates the hub server between the source and the target systems. The distributed architecture of third generation tools reduces the network traffic to improve the performance, distributes the load among database engines to improve the scalability, and supports all types of data sources.

Third Generation ETL uses relational DBMS for data transformations. In this generation the transformation phase does processing of data rather than row by row as in second generation ETL tools. "In the ETL architecture, all database engines can potentially participate in a transformation—thus running each part of the process where it is the most optimized. Any RDBMS can be an engine, and it may make sense to distribute the SQL code among different sources and targets to achieve the best performance. For example, a join between two large tables may be done on the source" (De Montcheuil, Y., 2005). RDBMS have power for data integration; ETL tools are taking the advantage of this feature of the RDBMS to improve their performance.

6. REAL-TIME DATA WAREHOUSING AND ETL TECHNIQUES

Increasingly there is a need to support and make business decisions in near real-time based on the operational data itself. Typical ETL architectures are batch update oriented and cause a significant lag in the currency of information at the data warehouse.

We question the performance effectiveness of typical batch ETL architectures and near real-time based updates on operational data and raise the questions to address instant-time research in order to address the timely business-decision making process. We raise the issue that the current ETL process needs to move away from periodic refreshes to continuous updates; however online updating of data warehouses gives rise to challenges of view synchronization and resource allocations. "To cope with real-time requirements, the data warehouse must be able to enable continuous data integration, in order to deal with the most recent business data" (Santos, R., J., Bernardino, J., 2009).

View synchronization problems arise when views are composed of data derived from multiple data sources being updated indiscriminately. Resource challenges result when there are conflicting resource demands of long-term analysis queries in the presence of concurrent updates.

In traditional ETL tools, loading is done periodically during the downtime and during this time no one can access the data in data warehouse. The separation between querying and updating clearly simplifies several aspects of the data warehouse implementation, but has a major disadvantage that the data warehouse is not continuously updated. (Polyzotis, N., et al. 2007). Traditional ETL tools are not capable enough to handle such continuous inserts or updates with no data warehouse down time.

In real time data warehousing loading is done continuously as opposed to a periodic basis in traditional approaches. One approach to the general architecture of a near real time data warehouse consisting of the following elements: (a) Data Sources hosting the data production systems that populate the data warehouse, (b) an intermediate Data Processing Area (DPA) where the cleaning and transformation of the data takes place and (c) the Data Warehouse (Vassiliadis, P., Simitsis A., 2008). The architecture is illustrated in Figure 6.

Each data source hosts a Source Flow Regulator (SFlowR) module that is responsible for the identification of relevant changes and propagates them towards the DW. Then the Data Processing Flow Regulator (DPFlowR) module is responsible of deciding which source is ready to transmit data.

We propose that further research is necessary to improve continuous data integration, but suggest that it is possible to achieve with emerging broadband technology.

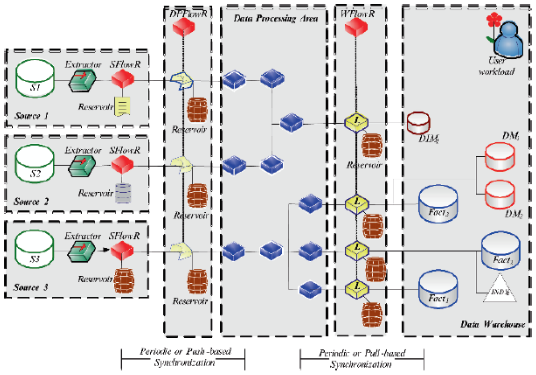


Figure 6: Architecture Near Real-Time DW
(Vassiliadis, P., Simitsis A., 2008)

The role of the data processing area (DPA) is to: a) cleanse and transform the data in the format required by the DW; b) act as the regulator for the data warehouse (in case the warehouse cannot handle the online traffic generated by the source); and c) perform various tasks such as check pointing, summary preparation, and quality of service management.

"A Warehouse Flow Regulator (WFlowR) orchestrates the propagation of data from the DPA to the warehouse based on the current workload from end users posing queries and the requirements for data freshness, ETL throughput and query response time." (Vassiliadis, P., Simitsis A., 2008).

Third generation ETL tools are using techniques to achieve real time data warehousing without causing downtime. Some of the real time ETL techniques are found in the research of J. Langseth (Langseth, J., 2008) and include:

1) Near real time ETL: The cost effective solution for applications that do not have a high demand for real time data is to just increase the frequency of loading, for ex: from daily to twice a day.

2) Direct Trickle feed: In this approach true real time data can be achieved by continuously moving the changed data from the source systems by inserting or updating them to the fact tables. There is a scalability problem with this approach because complex queries don't perform well with continuous updates. Constant updates on tables, which are being queried by reporting or OLAP tools leads to degrading the query performance of the data warehouse.

3) Trickle and flip: In this approach, data is inserted or updated into staging tables which are in the same format as target tables. The real time data is stored in staging tables, which have same format as historical data in target tables. The data warehouse can access fresh data instantly by getting a copy from the staging tables into the fact tables, the time window for refreshing the data warehouse can vary from hours to minutes.

4) External real time data cache: In this approach real time data is stored outside data warehouse in an external real time data cache (RTDC). The function of RTDC is to load the real time data into database from source systems. It resolves the query contention and scalability problem by directing the queries to RTDC which access real time data. With this approach, there is no additional load on the data warehouse as the real time data lies on separate cache data base. It provides up-to-the-second data and users don't wait for queries to get executed because they are so quick (Langseth, J., 2008).

5) Integrating OLAP and OLTP - Integrating OLAP and OLTP raises three questions:
   a) "Can we use one logical database for OLAP and OLTP?
   b) Can we integrate the physical data residing in the OLAP and OLTP repositories?
   c) Can we use DBMS engine to query the OLAP and OLTP data?" (Conn, S., 2005).

The physical data residing in OLAP is in its de-normalized form for query processing while relational online analytical processing (ROLAP) needs data to be in 3rd Normal form because it uses the relational queries for processing the data. Multidimensional analytical processing (MOLAP) can be used because data is built from a data cube, which is separate from transactional data.

7. REAL-TIME DATA BUSINESS INTELLIGENCE TECHNIQUES

One proposal for real-time business intelligence architecture requires that the data delivery from the operational data stores to the data warehouse must occur in real-time in the format referred to data streams of events (D. Agrawal, 2009). This proposal is illustrated in Figure 7. The usage of real-time data event streams eliminates the reliance on batched or offline updating of the data warehouse.

This architecture also introduces the middleware technology component, referred to as the stream analysis engine. This stream analysis engine performs a detailed analysis of the incoming data before it can be integrated into the data warehouse to identify possible outliers and interesting patterns. The goal of the stream analysis process is to "extract crucial information in real-time and then have it delivered to appropriate action points which could be either tactical or strategic" (D. Agrawal, 2009).

Complex Event Processing Engines (CEP engines) such as Stream-base enable business users to specify the patterns or temporal trends that they wish to detect over streaming operational data known as events. Decision

ISSN: 2167-1508
v5 n2214

makers can then take appropriate actions when specific patterns occur.

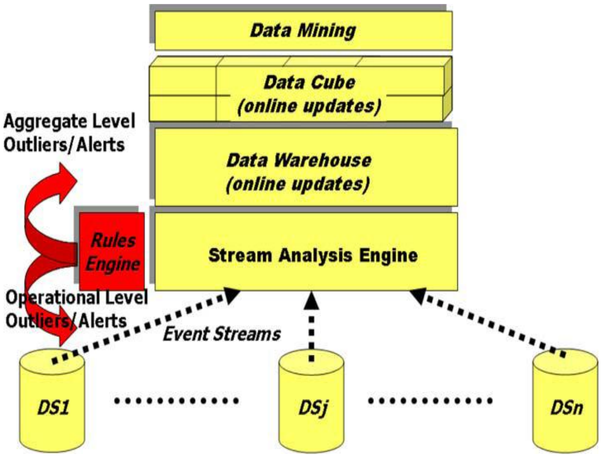

Data Mining
Data Cube (online updates)
Data Warehouse (online updates)
Stream Analysis Engine
Rules Engine
Aggregate Level Outliers/Alerts
Operational Level Outliers/Alerts
Event Streams
DS1
DSj
DSn

Figure 7: Architecture for a Real-time Business Intelligence System
(D. Agrawal, 2009).

The origins or CEP engines were in the financial domain where they were applied to algorithmic stock trading. More recently they are being applied to make decisions in real-time such as the click stream analysis of manufacturing process monitoring. The difference from traditional BI is shown in Figure 8, where operational data is no longer needed to be loaded into the Data Warehouse. Applications define declarative queries that can be performing on the incoming streams of data such as filtering, windowing, aggregations and joins (S. Chaudhuri, et. al 2011).

The arrival of events from the input streams trigger the query processing, and the queries are performed continuously as long as events arrive in the input stream. One major technical challenge is that the continuous running queries may reference data in the data base and impact near real-time requirements. A major challenge is that algorithms which require multiple passes over the data are no longer feasible for streaming data.

8. CONCLUSION AND FUTURE WORK

As the role of enterprises becomes increasing real-time such as E-commerce sites, real-time BI will be increasing important to such companies.
In the traditional ETL approach, the most current information is not available. With the increase demands by businesses for real-time Business Intelligence and Predictive Analytics, there is a need to build ETL tools, which provide real-time data into data warehouses**.**

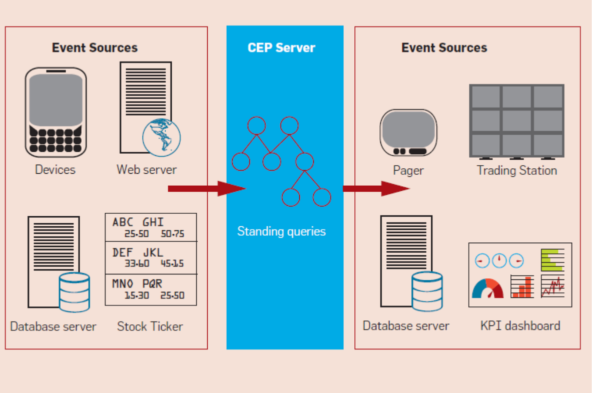


Figure 8: Complex Event Processing Server Architecture
(Chaudhuri, S., et al. 2011).

Not every analysis task warrants real-time analysis. The trade-off between the overhead of providing real-time business intelligence and data warehousing, and the intrinsic need for such an analysis calls for serious research and consideration. Otherwise, or the resulting system may have prohibited costs associated with it (D. Agrawal, 2009).

The underlying technology components and custom solutions are prohibitively expensive. The importance, complexity and criticality of such an environment make real-time BI and DW a significant topic of research and practice; therefore, these issues need to be addressed in the future by both the industry and the academia (Vassiliadis, P., Simitsis A., 2008).

9. REFERENCES

Agrawal, D., (2009), The Reality of Real-Time Business Intelligence, *Proceedings of the 2nd International Workshop on Business Intelligence for the Real Time Enterprise* (BIRTE 2008), Editors: M. Castellanos, U. Dayal, and T. Sellis, Springer, LNBIP 27, 75-88.

Ankorion, I. (2005). Change data capture: Efficient ETL for Real-Time BI. *Information Management, 15*(1), 36-36. Retrieved May 29, 2012 from http://search.proquest.com/docview/214690875?accountid=14584

Athanassoulis, M., Chen, M., S., Ailamaki, A., Gibbons, P. B., and Stoica R., (2011), MaSM: Efficient Online Updates in Data Warehouses, *In Proceedings of the 2011 International Conference on Management of Data (SIGMOD '11,* ACM, New York, NY, USA, 865-876. DOI=10.1145/1989323.1989414 Retrieved June 14, 2012 from http://doi.acm.org/10.1145/1989323.1989414

Behrend, A., Jörg, T., (2010), Optimized incremental ETL Jobs for Maintaining Data Warehouses. In *Proceedings of the Fourteenth International Database Engineering & Applications Symposium* (IDEAS '10). ACM, New York, NY, USA, 216-224. DOI=10.1145/1866480.1866511 Retrieved May 29, 2012 from http://doi.acm.org/10.1145/1866480.1866511

Bergamaschi, S., Guerra, F., Orsini, M., Sartori, C., Vincini, M., (2011), A Semantic Approach to ETL Technologies, *Data & Knowledge Engineering,* 70(8), 717-731.

Bloomberg BusinessWeek Research Services, (2011), The Current State of Business Analytics: Where Do We Go From Here? A white paper produced in collaboration with SAS.

Chaudhuri, S., Dayal, U., Narasayya, V., (2011) An overview of Business Intelligence Technology, *Communications of the ACM,* 54(8), 88-98. DOI= 10.1145/1978542.1978562 Retrieved June 29, 2012 from http://doi.acm.org/10.1145/1978542.1978562

Clikeman, P. M. (1999). Improving information quality. Internal Auditor, 56(3), 32-33.

Conn, S., S., (2005), OLTP and OLAP Data Integration: a Review of Feasible Implementation Methods and Architectures for Real Time Data Analysis, Southeast Con, Proceedings IEEE , 515- 520. DOI = 10.1109/SECON.2005.1423297 Retrieved May 14, 2012 from URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1423297&isnumber=30732

De Montcheuil, Y., (2005) Lesson – Third Generation ETL: Delivering the Best Performance, *What Works: Best Practices In Data Warehousing and Business Intelligence*, 20, p. 48.

Evans, B., (2005), Improving the data warehouse with selected data quality techniques: Metadata management, data cleansing and information stewardship. Capstone Report, University of Oregon, Applied Information Management Program. Retrieved June 14, 2012 from URL: http://hdl.handle.net/1794/7814

ETL Data Extraction Methods - Part Two Retrieved June 14, 2012 from http://www.best-business-intelligence.com/2011/09/etl-continued-data-extraction-methods.html

Gerber, M., Von Solms, R., (2008) Information security requirements – Interpreting the Legal Aspects, *Computers and Security,* 27, 124 - 135.

Huang, K., T., Lee, Y., W., Wang, R., Y., (1999), Quality Information and Knowledge, NJ: Prentice-Hall.

Huang, D. , L., Luen, P., Rau, P., Salvendy, G., (2010), Perception of Information Security, Behavior & Information Technology, 29(3), 221 - 232.

Henschen, D., (2010), Agile Business: 2010 BI and Information Management Survey, *Information Week*, Report ID: R1921110, Retrieved June 29, 2012 from http://analytics.informationweek.com

Henschen, D., (2011), 2012 BI and Information Management Trends, *Information Week*, Report ID: R335111, Retrieved June 29, 2012 http://reports.informationweek.com

Inmon, W. H., (1996), *Building the Data Warehouse,* 1st edition, Indiana: Wiley Publishing Inc.

Jarke, M., Vassiliou, V., (1997) Data Warehouse Quality: A Review of the DWQ Project. Retrieved May 14, 2012 from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.36.4346

Jesan, J., P., (2006) Information security. *Ubiquity,* Article 3 (January 2006), 1 pages. DOI=10.1145/1117693.1117695 Retrieved May 14, 2012 from http://doi.acm.org/10.1145/1117693.1117695

Jörg, T., Deßloch, S., (2008), Towards Generating ETL Processes for Incremental Loading. In *Proceedings of the 2008 international symposium on Database engineering & applications* (IDEAS '08). ACM, 101-110. DOI=10.1145/1451940.1451956 Retrieved June 29, 2012
http://doi.acm.org/10.1145/1451940.145196

Kimball, R. Reeves, L., Ross. M., Thornthwaite, (1998), *The Data Warehouse Lifecycle Toolkit: Export Methods for Designing, Developing and Developing and Deploying Data Warehouses*, 1st edition, Indiana: Wiley Publishing Inc.

Langseth, J., (2008), Real-Time Data Warehousing: Challenges and Solutions, DSSResources.COM.

Loshin, D., (2003), Business Intelligence. San Francisco: Morgan Kaufmann Publishers.

Loshin, D., (2003), Data Quality ROI in the Absence of Profits, *Information Management*, 13(9), 22-22.

Polyzotis, N., Skiadopoulos, S., Vassiliadis, P., Simitsis, A., Frantzell, N., (2007), Supporting Streaming Updates in An Active Data Warehouse: *In Proceedings of the IEEE 23rd International Conference on Data Engineering: ICDE 2007, 476-485. doi:10.1109/ICDE.2007.367893 Retrieved June 29, 2012 from URL:* http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4221696&isnumber=4221635

Oracle (2005) Oracle Database Data Warehousing Guide, 10g Release 2 (10.2)

Oxford Economics, (2011), Real-Time Business: Playing to Win in the New Global Marketplace, A white paper produced in collaboration with SAP.

Santos, R. J., and Bernardino J., (2009), Optimizing Data Warehouse Loading Procedures for Enabling Useful-Time Data Warehousing. *In Proceedings of the 2009 International Database Engineering & Applications Symposium (IDEAS '09)*. ACM, New York, NY, USA, 292-299. DOI=10.1145/1620432.1620464 Retrieved June 29, 2012 from http://doi.acm.org/10.1145/1620432.1620464

Shaker, H., El-Sappagh, A., Abdeltawab, M., Hendawi, A., Hamed, A., Bastawissy, E., (2011), A Proposed Model for Data Warehouse ETL Processes, *Journal of King Saud University - Computer and Information Sciences*, 23(2), 91-104. Retrieved May 14, 2012 http://www.sciencedirect.com/science/article/pii/S131915781100019X

Schneider, D., A., (2007), Practical Considerations for Real-Time Business Intelligence, Berlin: Springer

Simitsis, A., Skoutas, D., Castellanos, M., (2010) Representation of Conceptual ETL Designs in Natural Language Using Semantic Web Technology, *Data & Knowledge Engineering,* 69(1), 96-115.

Singh, R., Singh, K., (2010), A descriptive classification of causes of data quality problems in data warehousing. *IJCSI International Journal of Computer Science Issues*, 7(3).

Tank, D., M., Ganatra, A., Kosta, Y.,P., Bhensdadia, C.,K., Speeding ETL Processing in Data Warehouses Using High-Performance Joins for Changed Data Capture (CDC), Advances in Recent Technologies in Communication and Computing (ARTCom), 2010 International Conference on, pp.365-368. Retrieved June 29, 2012 from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5656810&isnumber=5655294

Third Generation ETL: Delivering the Best Performance Part 1 Retrieved May 14, 2012 from http://businessintelligence.com/third-generation-etl-delivering-the-best-performance-part-1/ May 2012

Vassiliadis, P., Simitsis A., (2008), Real-Time ETL: Annals of Information Systems: *New*

Trends in Data Warehousing and Data Analysis, 3, New York: Springer Publishing Company

Zubcoff, J., Pardillo, J., Trujillo, J., (2009), A UML Profile for the Conceptual Modeling of Data-Mining with Time-Series in Data Warehouses, Information and Software Technology, 51(6), 977-992.