Data Load Distribution by Semi Real Time Data Warehouse

Dr.Muhammad Younus Javed

College of Electrical and Mechanical Engineering, National University of Science and Technology, Rawalpindi,Pakistan.

myjaved@ceme.nust.edu.pk

Asim Nawaz

College of Electrical and Mechanical Engineering, National University of Science and Technology, Rawalpindi,Pakistan.

nawazasim@hotmail.com

Abstract-Today many organizations used data warehouse for strategic decision making. Today's real-time business stresses the potential to process increasingly volumes of data at very high speed in order to stay competitive in market. Data Warehouse is populated by data extraction, transformation and loading from different data sources by software utilities called ETL (Extraction, transformation & loading). ETL process is a time consuming process as it has to process large volume of data. ETL processes must have certain completion time window and ETL process must have to finish within this time window. In this paper we discusses a technique to distribute the volume of data to be extracted, transformed and loaded into data warehouse by merging both conventional and real-time techniques, so ETL process finishes its job within its time window by utilizing ETL idle time.

I.Introduction

Companies have demanding requirement for information in today's appalling ambitious organizatons. In order to keep a competitive benefit organizations has to identify what their clients desire, what are their requirement and the way in which the consumers wants to receive their services or products. The capability to access significant data in a well-timed, professional way by mean of well-known query and analysis tools are essential to grasp competitive advantages. In the same way the sharing and transportation of data throughout an organization, between departments, office and business ally is also very important [24].

The Bill Inmon pioneer of data warehousing in 1990 defined the term data warehousing in the following manner: ***"A warehouse is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of management's decision making process"***.

The key of the architected environment is data warehouse, and is the base of all DSS processing. Data Warehousing is about knowledge and training linked with how to systematically combine data from several distributed data source systems and to make use of that data in appropriate manner in order to support enterprise management and business decision-making.

A. Motivation for Project

Generally data in data warehouses grows asynchronously. ETL is a key methodology to bring data from heterogeneous and asynchronous sources to a homogeneous environment. Data warehouse refreshment can be done either by off-line in a batch mode (conventional way) or in a real time way. Both techniques have its own advantages and disadvantages. Main problem with conventional technique is vastness of data. The Real time technique results in overwhelming the source OLTP systems with the additional workload.

In today's fast growing information based market, companies must be able to integrate large amount of diverse data from different sources in order to support tactical IT plan such as Business Intelligence or Corporate. At the same time, IT companies are under constant stress to get more work done with fewer resources. The best way to overcome these contradictory goals is to implement a cost-effective integration methodology that increases the efficiency of the IT business.

B. Contributions

Now, the principal way for populating data marts and data warehouses is by means of extraction, transformation and loading (ETL) softwares. ETL refresh data warehouse in two ways a) Off-line way b) Real-Time way. Refreshing data warehouse in off-line fashion is time consuming due to vastness of data volume. Refreshing data warehouse in real-time fashion is expensive due to overburdening the source systems. In this paper we present a Semi Real-Time Data Warehouse Architecture for distributing data load. After that describe the technique to find which data to be extracted in real-time manner and in off-line manner. Then the implementation of extraction process of ETL process

978-0-7695-4042-9/10 $26.00 © 2010 IEEE
DOI 10.1109/ICCNT.2010.104

by semi real time way. In our approach we use oracle CDC mechanism to detect the changes occurring in source data.

II. Related Work

ETL processes are responsible for the tasks takes place at the back stage of data warehouse architecture. Data that are different from the previous execution of an ETL process should be extracted from number of different source systems. Extracted data then taken to a special-purpose area of the warehouse, called the Data Staging Area (DSA), where their transformation, homogenization, and cleansing take place. Finally, the data is loaded in the data warehouse (DW). In a conventional data warehouse approach, the data in the data warehouse is periodically refreshed by the ETL process during idle or low load periods of its operation e.g. every night and has a specific time-window to complete. Nowadays, business necessities and demands require near real-time data warehouse refreshment and significant attention is drawn to this kind of technological advancement[1].

There are different conceptual and logical design models that are used for data warehousing. In order to implement sales and shipping data warehouse systems, the comparison is made to select the most excellent conceptual and logical data models. According to author's comparison Object Oriented Multidimensional (OOMD) Model is best for conceptual modelling and Star Schema is best for Logical modelling [2].

In [4] the authors presents an overview of tasks involved in Extraction , Transformation & Loading processes. Its challenges and lists some ETL software tools exists in the market. This paper explains one of the listed software "informatica" that is used to perform ETL process.

A. EFFORTS RELATED TO CONVENTIONAL ETL

Two decades ago this issue has been already identified by the industrial requirements. Now at the start of new century research focus on demanding dilemma of ETL process. The modelling and design problems of ETL process has mostly focused in the begining of research works concerning conventional ETL processes. For ETL process conceptual design number of different techniques have been proposed that utilize diverse design techniques and formalisms e.g. UML [19, 21, 22]. Furthermore ETL processes logical models have been presented also e.g. [15].

Later on, during the last five years, research work has figure out the issues other then modelling of ETL processes. A little research efforts are also carried out for the ETL processes optimization [16, 8], together with works associated with distinct operators like the DataMapper [17]. Also research work in the field of data fusion [9,13], schema mapping and data cleaning [20,23] that is linked with ETL processes.

Currently there are many ETL tools available in market that are the result of research work that has been carried out in the field of data warehousing. Every leading database company provide ETL solutions and along with their database software they also ships ETL tools e.g. Oracle Warehouse Builder. There are many independent ETL tools vendors that provide their own ETL tools, such as Informatica.

B. EFFORTS RELATED TO SEMI REAL TIME ETL

In [11] the author discusses the state of the art of the Zero Latency Data Warehouse and for its implementations presents two tentative techniques. (a) The first one is the Grid-based Zero-Latency Data Stream Warehouse (GZLDSWH) that overcome the resource limitation problems by processing data stream without utitlizing approximation methodologies. (b) The secound one is Sense & Response Service Architecture (SARESA) that aware, define, calculate, automate and respond to business processes with the help of a complete Business Intelligence process.

In [3] author discussed the technique in order to implement real-time data warehouse. Based on Service Oriented Architecture (SOA) author presented practical real-time data warehouse architecture. The changed data from various source database systems are capture by means of data capture web service. The real time data warehouse is updating with the help of update technique based on web service and xml. In addition real-time data storage is attained by mean of the multi-level real-time data cache. We can easily implement real-time data warehouse by utilizing this type of data warehouse architecture.

In [5] authors discuss another technique for data warehouse refreshment. This technique differs from conventional data warehouse refresh technology in which data warehouse is updated in an off-line fashion after 24 hours. Both conventional and near real time ETL techniques are state of the art reviewed in this paper, the authors pinpoint the technical problems, the architecture and the background that take place in the field of near real time ETL, and for future work figure out motivating research issues.

In [10] author present a technique that is used to overcome the technical problem of workload management of the near real time data warehouse. At the data warehouse side the main dilemma is that with user queries data modification reach concurrently that results in causing overburdening the computational resources. The workload management mechanism presented by authors deals with the scheduling of query execution and the transactions are updated according to users preferences.

In [14], the architecture for the implementation of near real time data warehouse is discussed with the aim of: (i) minor modification in the software configuration of the source, (ii) minimum burden over the source because of the "active" method of data movement, (iii) the option of efficiently manage the general configuration of the environment in a right manner. This paper focus on spliting the extraction, the transformation and cleaning and the loading jobs. In order to carry out that particular job, each ETL activities examine its queue to glimpse whether data waited in the queue has to be processed or not. After that, it choose a chunk of records, carry out the processing and move them to the next phase. The performance and tuning of the overall refreshment process is calculated with the help of Queue theory.

III. The Semi Real Time Data Warehouse Architecture

The architecture proposed for semi real-time data warehouse is presented in figure 1. It has following components.

1. Source OLTP Systems.
2. CDC (Change Data Capture) Mechanism.
3. Transformation and Loading through Oracle Warehouse Builder (OWB).
4. Data Warehouse (Implemented by Star Schema).
5. Data Marts.
6. End User Workstations.

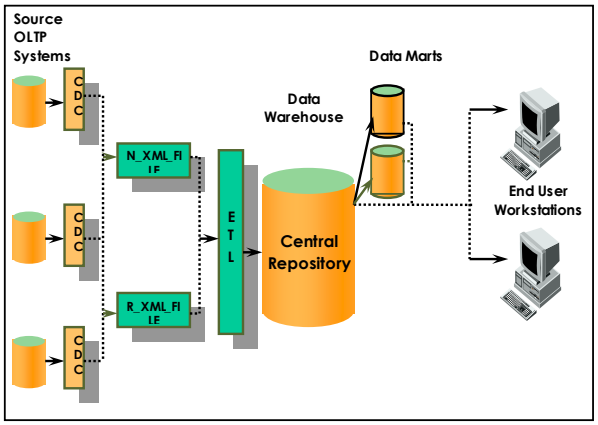

Figure 1

A. Methodology

ETL are key processes to extract, transform and load data in data warehouse. A challenge with ETL is that it is a time consuming process and each step is dependent of each other. If extraction takes lot of time to extract OLTP source data then transformation and loading also become time consuming processes. Our focus of work is to distribute the load of OLTP source data that has to be extracted that results in minimizing the time required by extraction, transformation and loading processes. The first step is to extract the OLTP source data. There are different mechanisms for extraction of OLTP source data. Our extraction technique differs from traditional extraction technique as we have to distribute OLTP source data load. Therefore we have to adopt a mechanism that divides the OLTP source data to be extracted, transformed and loaded into data warehouse. Thus resulting in shrinking ETL time window.

For OLTP source data extraction we implemented CDC mechanism. In our proposed architecture some data has to processed in real-time fashion and remaining in off-line fashion (convention way). Oracle CDC captures change data in synchronous and asynchronous manner. CDC mechanism copy change data in change tables by appending timestamp with change data.

We also maintain information about tables whose data is to be loaded in real-time and off-line manner with priority. We also write the utility in java that generates two files 1) N_XML_FILE / N_CSV_FILE and 2) R_XML_FILE / R_CSV_FILE. N_XML_FILE / N_CSV_FILE contain data to be loaded in off-line way and R_XML_FILE / R_CSV_FILE contains data to be loaded in real-time way. In this way we distribute OLTP source data load to be extracted, transformed and loaded in data warehouse.

The next step is to transform extracted data by applying different transformation techniques. Data transformation is a basic process in application scenarios concerning integration of data, migration of legacy data, data cleansing or data scrubbing, and extraction-transformation-loading procedures [17]. Transformation and loading are done with the help of oracle warehouse builder. N_CSV_FILE and R_CSV_FILE result as source input for oracle warehouse builder. External tables are created with the help of these CSV files. Using Oracle Warehouse Builder we create maps that links matching external tables and dimensions.

B. Motivation Example

We introduce an example in order to motivate our discussion. A database with schema S having set of n number of tables T = {t1, t2, t3,…,tn}. As we know that ETL process runs after 24 hours or a week or after month or after specified period of time. Let we consider that ETL performs it jobs after 24 hours.

Let Te, Tt and Tl are time consumed by extraction, transformation and loading processes after 24 hours. Total time consumed by whole ETL process is given by

$$T = Te + Tt + Tl;$$

T is total time consumed by ETL process if complete OLTP source data has to extracted, transformed and loaded into data warehouse after 24 hours.

We find that ETL process remains idle for almost 24 hours. After 24 hours lot of data is collected in OLTP source system that has to be extracted, transformed and loaded into data warehouse. By utilizing our technique now we calculate the total time taken by ETL process to finish its job. In our technique we have to first analyse which source table data has to be treated in real-time manner and which source table data has to be treated in conventional manner. Out of n source tables we consider that n/m table should be extracted , transformed and loaded in real-time manner where m is positive integer values from set {1,2,3,4,5,…}and remaining (n-n/m) source tables by conventional manner.

By utilizing our technique let we assume **Trm** is time consumed by ETL process to complete its job in real-time manner on n/m source tables and **Tcm** is time consumed by ETL process to complete its job in conventional manner on remaining (n-n/m) source tables. The percentage time taken by ETL process to finish its job in real-time fashion is given by the following equation.

$$\% \text{ Time Consumed } = (Trm \times 100) / T ;$$

and % time taken by ETL process to finish its job in conventional manner is given by the following equation.

$$\% \text{ Time Consumed } = (Tcm \times 100) / T ;$$

Percentage decrease in time for ETL process after 24 hours is given by the following equation.

$$\% \text{ Reduction Time } = [(T - Trm) \times 100] / T ;$$

In this way we have distribute the OLTP source data load of ETL process. Suppose by utilizing ETL idle time we process 33% of OLTP source data in real-time way. Then after 24 hours 67% OLTP source data is available for ETL process. Hence we have reduced the data load of OLTP source table to approximately 33% by processing 33% of OLTP source data in real-time fashion. So the reduction in OLTP source data load is directly proportional to data we have processed in real-time manner.

IV. Experimental Result

In this section, we report the major findings from our experimental study. Figure 2 dipicts comparison between data load without data load distribution and data load with data load distribution. Volume of data without data load distriubution is elevated as compared with data volume with data load distribution. Data to be extracted, transformed and loaded without data load distribution takes lot of time because of vastness of data. We can distribute data load to extracted,transformed and loaded by processing some data load in real-time manner and remaining in normal-way.

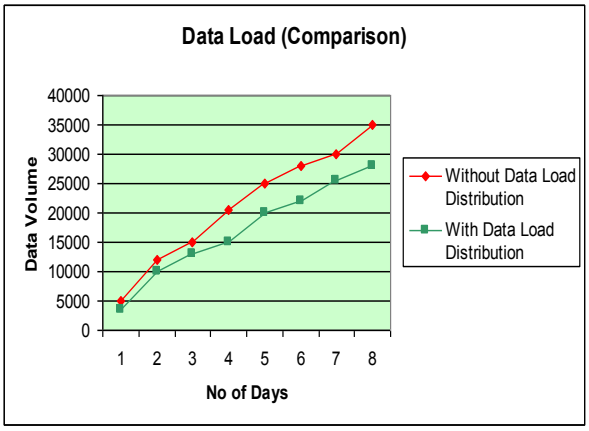

Figure 2

V. Conclusions and Future Research Work

Extraction, Transformation and Loading (ETL) are time consuming process. In our work we try to minimize time consumed by extraction process in ETL by extracting some data in a real-time manner and remaining in conventional way. We try to utilize ETL idle time to distribute load of OLTP source data. As in ETL extraction, transformation and loading are dependent processes. Data load distribution of extraction process also results in increasing the performance of transformation and loading process. In this paper we only implemented the extraction module for semi-real time data warehouse. So lot of future work can be done in this regards. Our next focus is to write a module that transform and load data both in real-time and normal manner. We also try to improve the extraction methodology discuss in this paper by implementing without the support of flat files. Another important issue is managing relations between OLTP source tables.

References

[1] EXTRACTION, TRANSFORMATION, AND LOADING , Panos Vassiliadis , Alkis Simitsis , Springer, 2009.

[2] A Casestudy of Data Models in Data Warehousing , Deepti Mishra, Ali Yazici, Beril Pinar Başaran , IEEE 2008.

[3] Data Updating and Query in Real-time Data Warehouse System , Youchan Zhu , Lei An , Shuangxi Liu , 2008 International Conference on Computer Science and Software Engineering.

[4] Extraction Transformation Loading – A Road to Data warehouse, Dr. Rajender Singh Chhillar , BarjeshKochar , 2nd National Conference Mathematical Techniques: Emerging Paradigms for Electronics and IT Industries 2008.

[5] Near Real Time ETL , Panos Vassiliadis , Alkis Simitsis , Springer 2008.

[6] A New Generation of Middleware Solutions for a Near-Real-Time Data Warehousing Architecture , Ronnie Abrahiem , IEEE EIT 2007 Proceedings.

[7] What-if Analysis for Data Warehouse Evolution , George Papastefanatos, Panos Vassiliadis, Alkis Simitsis, and Yannis Vassiliou , Springer 2007.

[8] Deciding the Physical Implementation of ETL Workflows, V. Tziovara, P. Vassiliadis, and A. Simitsis In DOLAP, pages 49–56, 2007.

[9] FuSem - Exploring Different Semantics of Data Fusion, J. Bleiholder, K. Draba, and F. Naumann, In VLDB, pages 1350–1353, 2007.

[10]Partition-based Workload Scheduling in Living Data Warehouse Environments, M. Thiele, U. Fischer, and W. Lehner , In DOLAP, pages 57–64, 2007.
[11]Zero-Latency Data Warehousing (ZLDWH): the State-of-the-art and experimental implementation approaches, Tho Manh Nguyen, and A Min Tjoa, Research, Innovation and Vision for the Future, 2006 International Conference on Volume , Issue , Feb. 12-16, 2006 Page(s): 167 - 176
[12]Role of Metadata in in the Data Warehousing Enviornment , Karanti Kumar Parankusham , Ravinder Reddy Madupu , 2006.
[13]Data Fusion in Three Steps: Resolving Schema, Tuple, and Value Inconsistencies, F. Naumann, A. Bilke, J. Bleiholder, and M Weis, IEEE Data Eng. Bull., 29(2):21–31, 2006.
[14]ETL Queues for Active Data Warehousing , Alexandros Karakasidis , Panos Vassiliadis , Evaggelia Pitoura , ACM 2005.
[15]A generic and customizable framework for the design of ETL scenarios, Panos Vassiliadis, Alkis Simitsis, Panos Georgantas, Manolis Terrovitis, Spiros Skiadopoulos, Elsevier Science Ltd 2005.
[16]Optimizing ETL processes in data warehouses, P Vassiliadis, A Simitsis, T Sellis, Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on Volume , Issue , 5-8 April 2005 Page(s): 564 – 575.
[17]Data Mapper: An Operator for Expressing One-to-Many Data Transformations, P. J. F. Carreira, H. Galhardas, J. Pereira, and A. Lopes, In DaWaK, pages 136–145, 2005.
[18]Modeling and Optimization of Extraction-Transformation-Loading (ETL) Processes in Data Warehouse Environments , ALKIS SIMITSIS , 2004.
[19]Data Mapping Diagrams for Data Warehouse Design with UML, S. Luján-Mora, P. Vassiliadis, and J. Trujillo, In ER, pages 191–204, 2004.
[20]Problems, Methods, and Challenges in Comprehensive Data Cleansing , Heiko Müller, Johann-Christoph Freytag , 2003.
[21]A UML Based Approach for Modeling ETL Processes in Data Warehouses, J. Trujillo and S. Luján-Mora, In ER, pages 307–320, 2003.
[22]Conceptual modeling for ETL processes, P. Vassiliadis, A. Simitsis, and S. Skiadopoulos, In DOLAP, pages 14–21, 2002.
[23]Data Cleaning: Problems and Current Approaches, Erhard Rahm, Hong Hai Do, IEEE Techn. Bulletin on Data Engineering, Dec. 2000.
[24]John Vandermay. Considerations for Building a Real-time Data Warehouse. Data Mirror Corporation. < http://www.grcdi.nl/considerations.pdf>.