Current ETL Techniques used in Data Stream Warehousing

Mandar Vaidya
School of Computing
National College of Ireland
Dublin, Ireland
x17153409@student.ncirl.ie

***Abstract*—Data management technology created to concurrently handle big and fast data is known as data stream warehousing. Data that is obtained from various data sources needs more processing, managing, analyzing and monitoring. Data warehouse stabilizes data coming from different data sources. Thus, real-time ETL techniques should process the data to extract value of it by monitoring the issues related to the data streams characteristics. This article will analyze the potential of ETL techniques that handle data streams. This article will further evaluate whether ETL is an effective method for today's challenges related to big data.**

Keywords—ETL (extract, transform and load), data streams, data warehouse, big data, real-time.

I. INTRODUCTION

In early stages of 1990, on-line analytical processing (OLAP) and Data Warehousing started expanding the outlook of Decision Support System (DSS). According to Bill Innon, who is often called the Father of Data Warehousing, " A Data Warehouse is a subject-oriented, integrated, time-variant, non-volatile collection of Data." Managing the data to its original form was a difficult task because of its redundancy and un-structured form. However, all these issues were solved by developing techniques to extract, transform and load (ETL) data into specific databases. However, the amount of data generated has increased tremendously as the cost of storage and computing has become negligible, which led to data warehousing techniques becoming unfeasible. Additionally, this also lead to the invention of Big Data paradigm. However, the demand of Big Data processing has increased so much that it rose to many challenges, which has its characteristics. Total there were 5 main characteristics; Volume, Velocity, Variety, Value, and Veracity, which are often known as the 5 V's of Big Data. Volume refers to the amount of data to be processed. Velocity refers to speed at which data is accessible. Variety refers to the arrival of data from different sources with different formats. Value refers to getting the value from that data so that it benefits your organization. Veracity refers to accuracy and authentication of data. However, the biggest obstacle that arrives while processing such big data is to extract the true information out of addressing the big 5 V's. To ease the process of querying, the processed data is stored in Data warehouse. The data warehouses often support analytical data processing. Data Streams, which are processed by Real-time ETL arrive from various sources and formats and are characterized by huge velocity and volume.

Large companies develop big-data systems using Hadoop, Sparkle or MapReduce based ideas for data mining while Data Stream Management Systems (DSMSs) are used to streamline data stream with light weight processing. The method as to where the gap between two large systems should be divided is widely known and hence it is possible to build two separate systems. So, for interpreting new big-data, data stream warehouses have been proposed to cut down the gap. However, building a data stream warehouse is challenging because storing and querying large amount of historical data as well as processing, handling, consistency and real-time response processing issues.

Many of the current ETL techniques process the data, which is found from databases which that is highly structured and then populate the data warehouse for further decision making with the required format. However, some of the ETL techniques attempt to handle the processing of data streams. However, many of the data streams are characterized by 5 V's which that needs to be addressed while processing the data. Thus, from the above perspective, the current available real time ETL techniques are inadequate to address all the characteristics of data streams in warehouse. Therefore, there is a need to rectify the real time ETL process that can address the issues related to data streams warehouse. [1]

II. Main Body

The main body of the paper consists of: related work in this area, tools available and their drawbacks in real time ETL process and analyzing the capability of current ETL techniques to handle maximum amount of data.

A. Extracting useful data from data streams by applying a continuous query onto the arriving data

Different sources produce data streams are not capable of storing the data by themselves. Therefore, there is a need to extract and process the data in real time so that the users can query the data and then make appropriate changes. However, the arriving data may be permanently if it is not captured properly. The data streams are characterized by velocity and volume which, refers to capturing of data. Therefore, the current techniques use continuous query methods to extract the data, which is only important with respect to a specific context while rest of data is deleted.

B. Transforming the extracted data into the form required by the data warehouse

Many of the current techniques of converting highly structured data into dimension and fact tables are required by traditional data warehouse techniques. However, data streams are characterized by disparities. Therefore, there is a need to solve the challenges that arise due to disparities of the data. Some of the current techniques to address the above issue is creating a semantic model by ontologies and RDFs.

C. Integration of real time and traditional data warehouse architectures

Integration of real time and traditional data warehouse architectures. Many of the current techniques enhances the traditional data warehouse architecture to handle real time data by adding a real time data storage component to the existing architecture. After being processed, data streams are loaded into the real time data storage component from which the users can query for the fresh data and after some time duration the data from the real time data storage area can be moved to the historical storage area for later use.

D. Synchronization of data updates and querying the data warehouse

While handling data streams, continuous loading and updating of processed data into the data warehouse is one of the most important process. However, it has disadvantages too. It affects the performance of fetching the query results from the warehouse. This leads to a major problem which that is known as query contention. Therefore, synchronization is needed in query questions and data updation. This issue is solved by introducing buffering techniques to capture the processed real-time data and updating the warehouse once all the conditions were trigged are satisfied.

III. Techniques to Handle Data Streams:

Authors Xiaofong Li and Yingchi Mao suggested an approach to handle data streams using real-time ETL. The method separated out real-time ETL from historical ETL and brings out a dynamic storage area to locate the problem of batch updates organizing the data aggregation operation with real-time and to advancing the freshness of the query results. However, this method does not solve the issues regarding unstructured data with huge velocity and volume. [2]

Authors of [3-4], put together semantic technology into the ETL process to solve the semantic heterogeneity of the data. The approach in [3] builds a semantic data model by mapping the data sets onto the ontologies and then loading the RDFs to the data warehouse. It solved the problem of integrating heterogeneous data source into a standard format and thus solving the issue of a variety characteristics and provided a way to run the data through RDF, links which in turn increases the efficiency of results. However, this method does not consider the volume & velocity of data, as it is challenging to make ontologies manually. Author of [4], urged a method with semantic path that use linked RDFs to show the data sets. Semantic filtering is used for organizing large volume of heterogeneous streams on the fly, then after filtering by repeated queries. The method sums up to discard some data when it exceeds a limit that can be handled. Thus, the semantics of original data is lost when it comes up with huge volume & high velocity.

Author of [5], initiated an approach to handle data streams which, combines a data stream processor and an operational data store (ODS). The method applies continuous query on arriving data streams so that the unwanted data can be filtered out and the data stream can shorten to a feasible size, then also it crosses to a specific threshold level which can be stored in the memory, further then it is spliced into 2 parts and samples are collected and the remaining data is neglected. The velocity of data stream but if the volume increases, the structure fails while processing the data. And, while processing, the semantics of original data is mostly lost.

Author of [6], initiated the problem of synchronizing data aggregation operation and querying the real-time data warehouse by introducing an algorithm called Integration Based Scheduling Approach (IBSA). It is branched in two parts: 1) Triggering the ETL process when data sets are coming from different sources and 2) Algorithm decides to invoke the thread that queries the real-time data warehouse or

to invoke the thread that updates the data warehouse. This idea, solves the issue of fair allocation of updating and query operations. However, the approach fails to explain when the input data while updating is characterized by high volume & velocity and when the query queue is full.

Author of [7], suggested a structure to support both real-time data warehouse to handle data streams and historical data warehouse to handle historical data. It catches the data by implementing multilevel caching technique which separates the freshness of the arriving data. It introduces the double mirror partitioning method to synchronize the data warehouse update process & querying operations. As, it uses caching method, buffer size should be finite. However, if the buffer size increases, efficiency decreases because of reduced hit-rate. Thus, when the data comes with huge volume & velocity, it gets drop because of limited buffer size. Thus, semantics of the original data is lost.

Authors of [8], gave the detailed information about the progression of ETL process. The data streams that are used for real-time ETL process need to address the two characterizes of volume & velocity. The architecture provided for real-time data processing used during extraction while stream processing the data look for few specified patterns of the data, while the rest is discarded. These techniques, it does not address any of the characteristics of the data streams thus, semantics of original data is lost.

Author of [9], provided a sneak peak of existing issues and current solutions of processing historical data as well as data streams. It also gave some vital views in addressing the existing difficult of joining the historical data and real-time data streams by using a buffering method. The transformation phase cannot cope up with the processing of large data streams and thus discussed a new technique called ELT in which data transformation is done before loading of data in data warehouse. However, this technique considers some highly un-structured data, thus, the problem of scaling with volume and velocity is not solved.

The method suggested in [10] first implemented a real-time ETL engine. It consists of RBFs (Remote Buffer Framework) which is responsible in receiving data streams from different sources. This in turn are connected to RIFs (Remote Integrator Framework) whose function is to accumulate data from different RBFs and then pass it on to real-time ETL. However, one of the flaw identified here is that different sources are connected to a single RBF thus making it difficult to receive arriving data from big data streams with different sources as it may run out of memory space because of huge volume. Thus, before processing the semantics of original data is lost.

The method in [11-12] bided to intensified more about traditional data warehouse to support real-time data stream processing. With few modifications to old ETL method like reducing the loading process to help isolate dynamic data faster. However, this modification, cannot cope up with characteristics of data streams. As, traditional ETL uses small sized fixed memory while processing, thus by giving storing issues to arriving data streams. And, the method also does not address the issues with velocity and variety of data streams as well.

Authors of [13] suggested a method to rectify the real-time data integration in the transformation of ETL process. This method implements a technique called Divide Join Data Integration (DJ-DI) whose action depends on the size of arriving data. When a change is analyzed in operational data, they are divided in less sizes and join operation is performed on each partition. As, this method takes only big and structured data but some of the data have huge velocity and different varieties, thus, giving major flaw to this technique.

First time initiated a new method which included web services based on real-time data. The technique attempts to address the real-time data warehouse challenges by integrating real-time data warehouse component from traditional data warehouse. As, it is based on web services, transferring the data from source to real-time data warehouse is very tough, as it does not support heterogeneous data types. As, it uses web services and does not include the challenges that arise due to network related issues such as bandwidth, protocol, security issues and so on because of huge velocity of arriving data streams. [14]

This method addresses the challenges of real-time data warehouse by using the three points: extraction of real-time data, consistency during integration and loading of processed data. As, it uses, log analysis for data streams which is characterized by huge volume and high velocity thus making it much more difficult to handle logs. As it focuses about integrating the data from various sources but lacks in heterogeneity of data while processing. By filtering functionality to remove less important data with respect to the context and thus the losing some of the original data too. [15]

Conclusion:

Real-time ETL techniques that have a potential to handle data streams should first deal with the characteristics for decisive processing of the data. As the rate of data generated is increasing at a rapid pace, the data that we recognized as real-time some years ago is not considered real-time today because of the expected response time decreasing at a faster rate. Therefore, there is a need to fill the big gap identified in this work and come up with a new and appropriate solution to address the issues that arise due to these characteristics of the big data.

References:

[1] K.V. Phainkanth, S.D. Sudarsan, “A Big Data Perspective of Current ETL Techniques”, ICACCE, 8073770, pp. 330-334, 2016.

[2] Xiaofang Li and Yingchi Mao, “Real-Time Data ETL Framework for Big Real-Time Data Analysis”, ICIA, Lijiang, pp. 1289-1294, IEEE International Conference, 2015.
[3] Srividya K. Bansal and Sebastian Kagemann, Integrating Big Data: A Semantic Extract-Transform-Load Framework”, in *Computer*, vol. 48, no. 3, pp. 42-50, IEEE, Mar. 2015.
[4] Marie-Aude Aufaure, Raja Chiky, Olivier Cure, Houda Khrouf, Gabriel Kepeklian, “From Business Intelligence to Semantic data stream management”, *Future Generation Computer Systems, Vol. 63*, pp. *100-107*, Elsevier, *October 2016.*
[5] F. Majeed, Muhammad Sohaib Mahmood and M. Iqbal, "Efficient data streams processing in the real time data warehouse", *ICCSIT, 3rd IEEE International Conference*, Chengdu, pp. 57-60, IEEE, 2010.
[6] J. Song, Y. Bao and J. Shi, "A Triggering and Scheduling Approach for ETL in a Real-time Data Warehouse”, *CIT, 10th International Conference*, Bradford, pp. 91-98, IEEE, 2010.
[7] Shao YiChuan, Xingjia Yao, “Research of Real-time Data warehouse Storage Strategy Based on Multi-Level Caches”, ICSSDMS, Macao, Vol. 25, pp. 2315-2321, ELSEVIER, April 2012.
[8] Kakish Kamal, and Theresa A. Kraft. "ETL evolution for real-time data warehousing", *Proceedings of the Conference on Information Systems Applied Research*, Vol. 2167, p. 1508, 2012.
[9] Revathy S., Saravana Balaji B. and N. K. Karthikeyan, “From Data Warehouse to Streaming Warehouse: A Survey on the Challenges for Real-Time Data Warehousing and Available Solutions”, International Journal of Computer Applications, Vol. 81-no2, 2013.
[10] A. Wibowo, "Problems and available solutions on the stage of Extract, Transform, and Loading in near real-time data warehousing", *ISITIA*, Surabaya, pp. 345-350, IEEE, 2015.
[11] Marcin Gorawski and Anna Gorawska, “Research on the Stream ETL Process”, BDAS, 10th International Conference, Poland, Vol. 424, pp. 61-71, Springer, 2014.
[12] Alfredo Cuzzocrea, Nickerson Ferreira and Pedro Furtado, “Enhancing Traditional Data Warehousing Architectures with Real-Time Capabilities”, ISMIS, 21st International Symposium, Denmark, Vol. 8502, pp. 456- 465, Springer, 2014.
[13] Alfredo Cuzzocrea, Nickerson Ferreira and Pedro Furtado, “Real-Time Data Warehousing: A Rewrite/Merge Approach”, LNCS, Germany, Vol. 8646, pp. 78-88, Springer, 2014.
[14] Imane Lebdaoui, Ghizlane Orhanou and Said Elhajji, “An Integration Adaptation for Real- Time Data Warehousing”, IJSEIA, Vol. 8, pp. 115-128, 2014.
[15] M. Obal□, B. Dursun, Z. Erdem and A. K. Görür, "A real time data warehouse approach for data processing", *SIU*, Haspolat, pp. 1-4, IEEE, 2013.