Optimization of ETL Process in Data Warehouse Through a Combination of Parallelization and Shared Cache Memory

M. Faridi Masouleh
Information Technology Management Department
Science and Research Branch, Islamic Azad University
Tehran, Iran
m.faridi@srbiau.ac.ir

M. A. Afshar Kazemi
Information Technology Management Department
Science and Research Branch, Islamic Azad University
Tehran, Iran
m.afsharkazemi@yahoo.com

M. Alborzi
Information Technology Management Department
Science and Research Branch, Islamic Azad University
Tehran, Iran
Mahmood_alborzi@yahoo.com

A. Toloie Eshlaghy
Information Technology Management Department
Science and Research Branch,
Islamic Azad University
Tehran, Iran
toloie@gmail.com

Abstract—**Extraction, Transformation and Loading (ETL) is introduced as one of the notable subjects in optimization, management, improvement and acceleration of processes and operations in data bases and data warehouses. The creation of ETL processes is potentially one of the greatest tasks of data warehouses and so its production is a time-consuming and complicated procedure. Without optimization of these processes, the implementation of projects in data warehouses area is costly, complicated and time-consuming. The present paper used the combination of parallelization methods and shared cache memory in systems distributed on the basis of data warehouse. According to the conducted assessment, the proposed method exhibited 7.1% speed improvement to kattle optimization instrument and 7.9% to talend instrument in terms of implementation time of the ETL process. Therefore, parallelization could notably improve the ETL process. It eventually caused the management and integration processes of big data to be implemented in a simple way and with acceptable speed.**

Keywords-Shared cache memory; ETL Process; parallelization; ETL optimization

I. Introduction

Data warehouse applications have utilized Extraction, Transformation and Loading (ETL) processes through tools that extract data from data resources, transform them to an acceptable format and load them in a data provider [1]. Such processes include a collection of instruments used for extracting, cleaning, customization, remolding, merging and data loading from different far away databases to a data warehouse [2]. During ETL, the data of required data providers (databases, text files, old systems and widespread pages) is extracted and transformed into compatible data within a definite framework and placed into a data reservoir. Different specialties such as commercial analysis, database design and programming are essential for implementation of ETL process. Prior to ETL implementation data providers, their destination and the transformation needed should be recognized and determined. This requires an initial data gathering and modeling stage followed by a more detailed one in the ETL design and implementation stage [3].

A variety of approaches have been discussed to optimize ETL. In [7], authors determined a ETL process path for optimization of implementation time. They improved the operations and tasks related to a process without using the parallelization process. In [1, 4-5], authors proposed a theoretical framework which formally defined the scenario of ETL processes in the form of an undirected acyclic graph. In [6], authors used a law-based optimization method which was a complicated method demanding abundant coding. In [7], authors presented a new solution to discover a standard conceptual model in order to implement the extraction, transformation and loading operations. They categorized their method into three phases: the first is the mapping of terms and instructions, the second phase was based on conceptual structure and the last phase of modeling was based on UML concepts. In [8], authors presented a new method based on stream control in ETL to optimize process speed. They accomplished to commercialize their method and provide a new idea for other researches. In [4], authors used an intelligent method based on grid in physical and cyber environments to manage and improve the ETL process. They generally developed their study on big data for the emergence of the fields related to cyber and physical systems which were based on text. They finally achieved to integrate the spatial and non-spatial data in cyber environment.

With regard to the examination of weak and strong points of former researches, the present paper has presented a new combined method by usage of parallelization techniques and simultaneous use of multiple cores to process and manage different databases in scattered locations as well as the application of cache memory shared between cores which conduct the operations of implementation, transformation and loading of data from distributed data bases in different locations and main data warehouse located in a definite place.

II. CONCEPTS OF ETL

ETL is considered a process which should be continually performed in system. This process is also conducted in return for operative data which come to existence in organization during time. What matters in the establishment of an intelligent business organization is the creation of a proper architecture and structure so that ETL process should be conducted compatible with different operations in which it occurs. So the structure applied for prior ETL has great importance. The ETL process should be conducted in different stages since it is applied in large volume of data and is usually accompanying with data integration. The noteworthy issue is that when ETL process initiates during these stages, the high volume of network traffic and processing of database servers may cause disorder in other intelligent business processes.

An ETL system has four main sections:

- Extraction
- Transformation
- Loading
- Meta data

A. *Extraction phase*

The data should be initially extracted from respective data providers. In this phase, the data may be deleted from initial data providers or copied in data warehouse without being omitted. The old data are often not applicable in organization's daily affairs whose maintenance is merely for keeping the system history would be deleted from preliminary data providers and transferred to data reservoir. So the efficiency and performance of aforesaid data resources would be kept at a desirable level. Data extracted from initial data providers are usually placed in staging space of data warehouse and processed in other ETL phases. This space is a relational data base emerged as temporary memory space for data processing. The phase of data extraction is usually conducted at the level of data resources especially when the respective data resource is a data base. The prevalent method in old systems for data extraction is the production of text files on the basis of data. The new systems apply ODBC, OLEDB and API for this purpose.

B. *Transformation phase*

After extraction of data, certain processes should be done so that they reach a proper and integrated format. This phase is performed as follows:

- Data Validation: Compatibility and absence of contradiction between new data extracted from data providers and information present in data warehouse is examined.
- Data Verification: Do the fields have correct values? For example, in a field with on and off values, do all the data possess one of the two values?
- Data Transformation: Data originates from diverse data providers and so similar fields may have diverse values. For example, a two value field may be on and off in a data provider and 0 and 1 in other data providers. The entire data entering the warehouse should be modified in this respect.
- Applying Business Regulations: In this phase we should consider if present data is compatible with organizational needs. For example does the customer information includes their first and family name?
- Data Integration: Is it possible for one system to keep customer information and the other system keep the sales information. The data present in both systems should be integrated.

This is actually the most complicated phase in ETL process. A part of this process could be implemented in data extraction phase such as old information systems in which information is gathered from entire data files and a text file is created base on them [7].

C. *Loading Phase*

Data transformed to respective standard form are placed into data warehouse in this phase. The data are loaded periodically and not continually due to their high volume. In other words, when data are transformed in a data provider or new information is added, the changes are not instantaneously transferred to data warehouse. But they are updated periodically and in a regular time span [7].

D. *Meta Data*

Meta data includes information on transmission and conversion of data, data warehouse performance, contradiction in data providers, determined data base diagrams and the data warehouse places in which initial data resources are mapped. The data present in Meta data could be applied in cases such as automatic supervision, prediction of organizational trends and reapplication of information [7].

III. ARCHITECTURE AND ANALYSIS OF THE RECOMMENDED SCHEME

Figure 1 demonstrates the architecture of the recommended scheme. It is observed that all data that belong to distributed databases in diverse locations enters the target operative space and each takes a responsibility in present processor system. The cores also simultaneously receive and process the data and transfer them to data warehouse in a parallel form. The present paper generally used two combined strategies of parallelization and shared cache memory in order to optimize the ETL operations and manage the data in data bases.

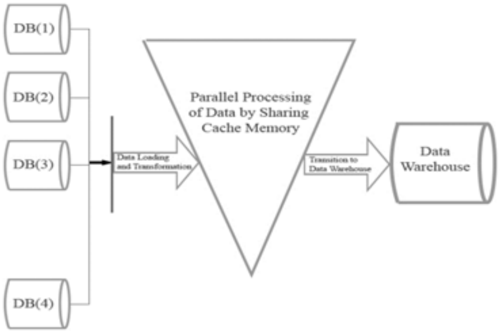
DB(1)
DB(2)
DB(3)
DB(4)
Data Loading and Transformation
Parallel Processing of Data by Sharing Cache Memory
Transition to Data Warehouse
Data Warehouse

Fig. 1. The recommended scheme

A. Shared Cache Memory

One of the notable and remarkable issues in ETL process is the challenge of separate utilization of cache memory. In case that in a distributed operative system, separate cache memories were used in input and output, the processor and main memory are obliged to transfer the caches in each specific operation. It is itself an important issue in great and sensitive matters which demand high speed operation. Figure 2a demonstrates the manner of using cache memory in different instruments. It is seen that this type of systems require transition and duplication in each time of implementation. Figure 2b demonstrates another condition of dependent cache memory. The present paper averts the challenge of transition of cache memory by different providers through application of cache memory shared between diverse providers. So it is not essential to transfer cache memory in each performance by different providers in each system which finally leads to the improvement of speed, ETL process and system operation. Figure3 shows an aspect of common cache memory. The manner of parallelization of processes in system in present study is based on the processing cores existing in operative environments. So the procedure is that by entering a process to ETL process, in case of the inactivity of each processor and its cores, the mentioned operation is selected and processed by inactive core. After the completion of processing, the core is placed in queue and is ready to receive the process. It is to be mentioned that the processors and related active cores are implemented in queues and controls and processes the operation.

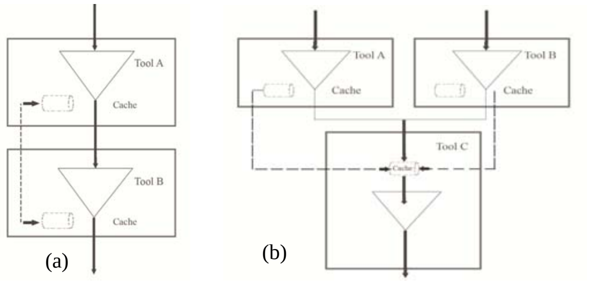

Fig. 2. Utilization of dependent cache memory (a) in diverse instruments (b) in different instruments

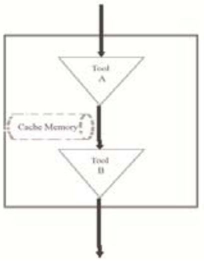

Fig. 3. Utilization of common cache memory shared in different instrument

IV. Results evaluation

C# language is used for the optimization of ETL operation in distributed data bases in this paper (Table I). The comparisons conducted in terms of speed, optimization level of ETL process by this instrument or others including kettle [9] and talend [10] will be described in following section. It is to be mentioned that the following Jquery is used to assess the results and extract records:

```
SELECT d_year,  c_nation,
SUM (lo_revenue - lo_supplycost) AS profit
FROM date, customer, supplier, part, lineorder
WHERE lo_custkey = c_custkey AND lo_suppkey = s_suppkey AND
      lo_partkey = p_partkey AND lo_orderdate = d_datekey AND
      c_region = ’AMERICA’ AND s_region = ’AMERICA’ AND
      (p_mfgr = ’MFGR#1’ or p_mfgr = ’MFGR#2’)
GROUP BY d_year, c_nation
ORDER BY d_year, c_nation
```

Five different data bases and different numbers of samples were employed. Further, shared cache memory, different number of cores and serial and parallel implementation were also investigated. Results are depicted in Figures 4-5. It is shown that the level of difference between parallel implementation of ETL process with parallel and shared cache memory is very significant, however it functions 263 times better than average condition (Figure 6). Table II demonstrates a brief account of different implementation times in m/S compared to recommended method. Figure 7 demonstrates the comparison between the recommended scheme and other ETL optimization instruments. As shown, the proposed method exhibits about 7.1% speed improvement compared to kattle optimization instrument and 7.9% to talend instrument.

TABLE I. OUTPUT AND RESULTS

Volume(M-Byte)	Samples Number	Serial Running Time	Parallel Running Time
0.143051	100	5133	14
0.715255	500	28,111	27
1.430511	1,000	54,834	39
2.861022	2,000	106,960	52
7.152557	5,000	208,638	65
14.30511	10,000	406,974	77
28.61022	20,000	793,853	90
42.91534	30,000	1,548,508	103

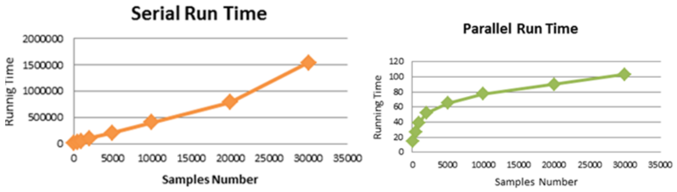

Fig. 4. Implementation time

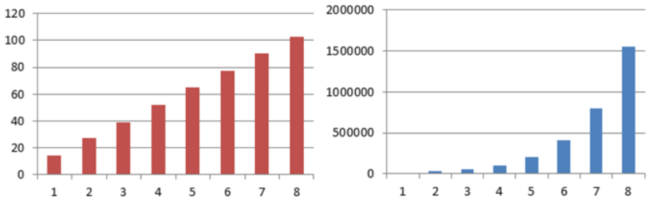

Fig. 5. Serial implementation time

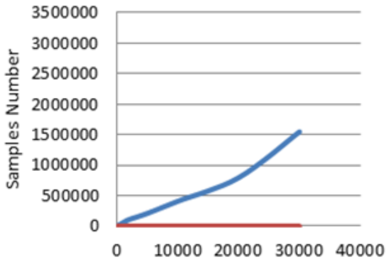

Fig. 6. Comparison between parallel implementation time and shared memory with serial implementation (m/S)

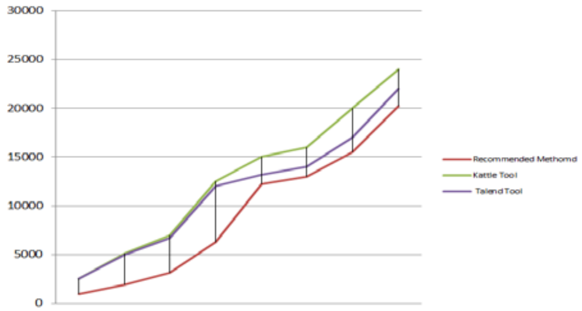

Fig. 7. Comparison between recommended methods with other ETL optimization instruments

TABLE II. COMPARISON BETWEEN IMPLEMENTATION TIME OF RECOMMENDED METHOD AND ETL OPTIMIZATION INSTRUMENTS

Volume(Gig)	***Recommended Method***	***Kattle Tool***	***Talend Tool***
1	927	2500	2500
2	1900	5100	5000
3	3150	7000	6700
4	6300	12500	12000
5	12200	15000	13200
6	13000	16000	14000
7	15500	20000	17000
8	20200	24000	22000

V. Conclusion

A variety of methods have been proposed for ETL optimization in distributed and big data banks that integrate various instruments. With regard to the importance of the issue and the challenges in this area, including confidence and speed, the present paper introduced a new method that includes both methods of parallelization and shared cache memory. The proposed scheme shows almost 7.1% speed improvement compared to kattle optimization instrument and 7.9% compared to talend instrument. Future work may focus in the utilization of real parallelization hardware instead of virtual hardware and optimization of ETL process in a cloud environment.

References

[1] A. Simitsis, P. Vassiliadis, T. Sellis, "Optimizing ETL Processes in Data Warehouses", IEEE 21st International Conference on Data Engineering (ICDE'05), pp. 2-4, 2005

[2] J. A. Sharp, Data Flow Computing: Theory and Practice, Intellect Books, 1992.

[3] M. Bala, O. Boussaid, Z. Alimazighi, "Big-ETL: Extracting-Transforming-Loading Approach for Big Data", International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA), pp. 1-4, 2015

[4] A. V. Simitsis, P. Vassiliadis, T. Sellis "Optimizing ETL Processes in Data Warehouses", 21st International Conference on Data Engineering (ICDE 2005), pp. 564–575, 2005

[5] A. W. Simitsis, , K. Wilkinson, U. Dayal, M. Castellanos, "Optimizing ETL Workflows for Fault-tolerance", 26st International Conference on Data Engineering, pp. 385–396, 2010

[6] A. Behrend, "Optimized Incremental ETL Jobs for Maintaining Data Warehouses", 14th International Database Engineering & Applications Symposium, pp. 216-224, Montreal, Quebec, Canada — August 16 - 18, 2010

[7] S. H. A. El-Sappagh, A. M. A. Hendawi, A. H. El Bastawissy, "A proposed model for data warehouse ETL processes", Journal of King Saud University Computer and Information Sciences, Vol. 23, No. 2, pp. 91-104, 2011

[8] A. Longo, S. Giacovelli, M. Bochicchio, "Fact – Centered ETL: A Proposal for Speeding Business Analytics up", Procedia Technology, Vol. 16, pp. 471-480, 2014

[9] P. Kettle, "Pentaho Kettle Project", Kettle Project, 2014

[10] X. Liu, Optimizing ETL Dataflow Using Shared Caching and Parallelization Methods. Arxiv, CoRR abs/1409.1639, 2014