International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.177
Volume 7 Issue V, May 2019- Available at www.ijraset.com

The Extract Transform Load (ETL) Process and Optimization using Ab Initio

Mr. Rohan S. Ratnaparkhi[1], Mr. Prakash Kene[2]
[1]*MCA Department, Modern College of Engineering, Pune*
[2]*Assistant Professor, MCA Department, Modern College of Engineering, Pune*

Abstract: Data is everything! In the technological world, this statement holds true. Technology all over the world produces a huge amount of data every day every minute every second. What do we do with this data? This data is used to produce more data that can help businesses all over the world to innovate more and grow. In this process, an ETL tool is used to extract the data from different RDBMS systems and then that data is transformed by applying some calculations etc, and then load the transformed data into the Data Warehouse. The purpose of this study is to present the structure of ETL process in a structured manner along with optimization of ETL in details. We will be covering how the process of Extraction of data from various sources, transform that data by performing various calculations on it, and loading/storing the data into warehouse for further use takes place and how various tools helped to optimize the ETL process. It is the base of every data warehouse systems, data mining and business intelligence.

I. INTRODUCTION

A data warehouse system receives data from multiple sources. It is also called as sources of the data. For a purpose present its end-users a way to combined and manageable information. At the time of actual implementation the task of data extraction, it has to face some probelms, which can be shortly summarized as follows. Since the data is coming from various multiple sources, it can have different schema which needs to be transformed to a common schema. It is an important task. Another problem it faces is, data which is extracted can have quality problems which ranges from simple spelling errors to inconsistent data. And these kind of data quality errors should be removed from the data to make the data clean and complete. At last, we must consider that the data is populated regularly. That means it is necessary to refresh the data warehouse contents regularly to provide up to date data. All these problems needs that the necessary processes that are constructed by development team are executed in appropriate time intervals for correct and complete data population in data warehouse. This process is called as Extract Transform Load (ETL) process.

The ETL process is responsible for extraction of data from various sources, transforming that data in an appropriate way which will allow it to be further processed according to need, and transforming the data or computation of new values , cleaning and isolation of noisy data to check if the data guarantee business rules and database constraints. Lastly loading the cleaned and transformed data into appropriate data warehouse or data mart.

A. Extraction

A staging area is required during the ETL load. There are various reasons why the staging area is required. The source systems are only available for a specific period of time to extract data. The time required for extraction is less than the loading time. Therefore, the staging area allows you to extract the data from the source system and keeps it in the staging area before the time slot ends.
The staging area is required when you are extracting the data from multiple sources or joining various systems together. For example is is difficult to perform join query on two table which are located in different servers or databases.
Data extractions' time slot for different systems vary as per the time zone and operational hours.
This data extracted from multiple sources can be used in multiple data warehouse systems, data stores. Along with that ETL process allows you to perform complex calculations and transformation and it requires extra space to store the data.

B. Transform

In the transformation phase, you can apply various functions on the data extracted in earlier phase to load it into the data warehouse system. The data which does not need any transformation is called as direct move or direct pass. You can perform various calculations on the data for getting meaningful, accurate and complete data. You can use various functions to generate the data which is not present. For example , you can use average function of sql to calculate the average of a field.

C. Load

In this phase, the data which is extracted and transformed is loaded/stored in the data warehouse system.

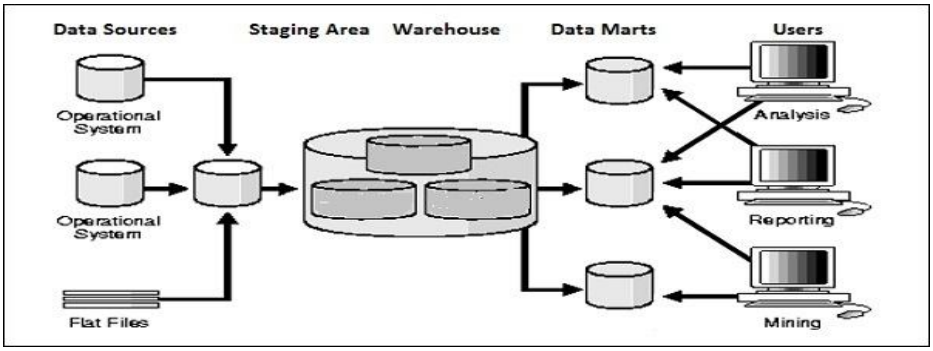

Fig1.1. The ETL process

Fig.1.1 describes the process of ETL. The data is extracted from various sources. It is then dumped into temporary staging area. Transformations takes place on this staging area data. Then the data is loaded into data warehouse. From the data warehouse, various data marts are created and they are used by various users for specific purposes.

II. LITERATURE REVIEW

The historical background for ETL processes goes all the way back to the birth of information processing software. Software for transforming and filtering information from one (structured, semi-structured, or even unstructured) file to another has been constructed since the early years of data banks, where the relational model and declarative database querying were not invented. Data and software were considered two sides of coin for data management by that time and thus, the ETL software was not treated as a stand-alone, special purpose module of the information system**.**

The EXPRESS system (Shu, Housel, Taylor, Ghosh, & Lum, 1977) is the first attempt that we know with the purpose of producing data transformations, taking as input data definitions or the involved nonprocedural statements.

Through the advanced years, the importance on the data integration problem was significant, and wrapper-based exchange of data between integrated database systems was the closest thing to ETL that we can report – for example, see Roth and Schwarz (1997)[6]. As Vassiliadis and Simitsis (2009) mention “since then, any kind of data processing software that reshapes or filters records, calculates new values and populates another data store than the original one is a form of an ETL program.” After the relational model had been born and the declarative nature of relational database querying had started to gain ground, it was quite natural that the research community would try to apply the declarative paradigm to data transformations[9].

The authors build their approach on a generic structure for the design process for ETL workflows. So, they construct their common design process in six stages, specifically, (i) source choice, (ii) source data transformation/alteration, (iii) joining of data, (iv) selection of the target, (v) mappings of attributes between source and target data and (vi) loading of data.

ETL has taken its name and existence as a separate set of tools and processes in the early ‘00s. Despite the fact that data warehouses had become an established practice in large organizations since the latest part of the ‘90s, it was only in the early ‘00s that the engineering vendors and the examination community cared to deal seriously with the field. It is noteworthy that till then, the research community had typically hidden the internals of ETL process “under the carpet” by treating the data warehouse as a set of materialized views over the sources. At the same time, the industrial vendors were focused on providing fast querying and reporting facilities to end users. Still, once data warehouses were established as a practice, it was time to focus on the tasks faced by the developers. As a result, during the ‘00s, the industrial field is flourishing with tools from both the major database vendors and specialized companies and, at the same time, the research community has abandoned the treatment of data warehouses as collections of materialized views and focuses on the actual issues of ETL processes.

A. Problems of ETL

There are numerous problems to implementing efficient and reliable ETL processes.

1) Technical challenges moving, integrating, and transforming data from disparate environments
2) Short load windows, long load times
3) Inconsistent, difficult to maintain business rules

4) Lack of exposure of business rules to end users
5) Source systems missing certain critical data
6) Poor query performance

III. OPTIMIZATION OF ETL

The minimization of the execution time of an ETL process is of particular importance, since ETL processes have to complete their task within specific time windows. Moreover, in the unfortunate case where a failure occurs during the execution of an ETL process, there must be enough time left for the resumption of the workflow. Traditional optimization methods are not necessarily applicable to ETL scenarios. As mentioned by Tziovara, Vassiliadis & Simitsis (2007) “ETL workflows are NOT big queries: their structure is not a left-deep or bushy tree, black box functions are employed, there is a considerable amount of savepoints to aid faster resumption in cases of failures, and different servers and environments are possibly involved. Moreover, frequently, the objective is to meet specific time constraints with respect to both regular operation and recovery (rather than the best possible throughput).” For all these reasons, the optimization of the execution of an ETL process poses an important research problem with straightforward practical implications[9].

IV. AB INITIO FOR OPTIMIZATION

Ab initio is a Latin term meaning "from the beginning". Ab Initio Software is an American multinational enterprise software corporation. The company specializes in high-volume data processing applications and enterprise application integration. The Ab Initio products are provided on a user-friendly homogeneous and heterogeneous platform for parallel data processing applications. These applications perform functions relating to fourth generation data analysis, batch processing, complex events, quantitative and qualitative data processing, data manipulation graphical user interface (GUI)-based parallel processing software which is commonly used to extract, transform, and load (ETL) data.

A. How it Helps

1) Co-Operating System: The Co>Operating System is an environment for building, integrating, and running enterprise business applications. The heart of the Co>Operating System is a “dataflow engine.” This engine drives a large library of data processing “components” that manipulate the data flowing through an application. Applications are designed, implemented, and maintained graphically through Ab Initio’s Graphical Development Environment™ (GDE). The core principle of the Co>Operating System is that applications are designed and developed in the way most people would design a system on a whiteboard (or even on a napkin). Easily recognizable icons are used to represent the input and output sources, which are then combined with processing boxes and arrows to define the overall processing flow. By selecting the appropriate components from an extensive library and “wiring them up,” you create an Ab Initio application.

Ab Initio flawlessly integrates the design and execution of applications: the drawing is the application. And the resulting application can be batch, near real-time, or real-time in nature, or even a combination of all of these – all united into one consistent and powerful computing environment. The graphical dataflow approach means that Ab Initio can be used to build the vast majority of business applications – like complex event processing , data warehousing and data quality management systems , operational systems, distributed application integration.

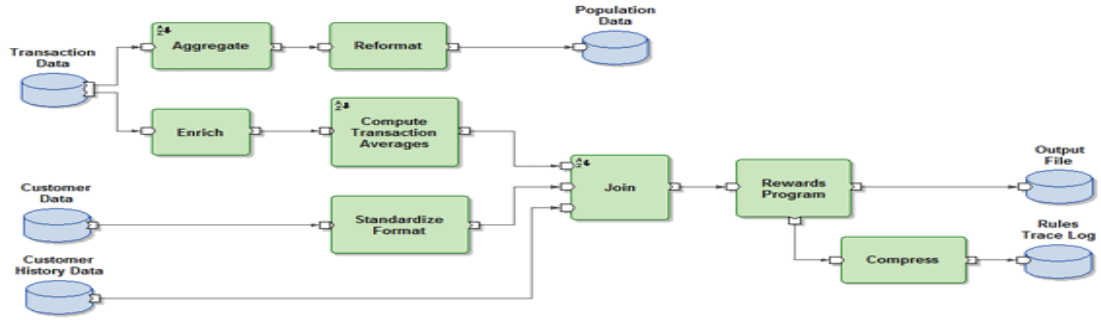

Fig 1.2 Ab Initio Graph describing ETL

Fig 1.2 describes how Ab Initio provides various components and functionality by using which we can perform ETL process with ease.

2) *Achieving Performance For Different Real-Time Execution Models:* Application architects are often challenged by the need to meet seemingly conflicting performance requirements:

Batch applications need to process data as professionally as possible. A batch job may take a long time to run because there are so many transactions to process, and none of the results are accessible until the job has completed. But while it is running, a batch job is expected to process a very high number of records per second. "Mini-batch" applications are groups of batch jobs that separately process slight volumes of data. However, there may be thousands or even tens of thousands of such small jobs that run each day. By restricting the amount of data treated by a job, the response time for each job is lessened. This method also allows the same applications to process very large data volumes competently in a traditional batch setting. Asynchronous messaging applications connect to message lines and also need to process transactions as proficiently as possible. However, the downstream systems usually expect their answer messages within a few seconds to tens of seconds. Certainly, if an asynchronous application can respond within a second or two, it can support interactive use. "Request/response" or synchronous messaging applications are expected to process a transaction as soon as it shows up and to reply as quickly as possible, usually with a latency of less than a second. If multiple such applications work together to process a transaction, individual applications may need to turn around responses in tenths to hundredths of a second. Ab Initio directly addresses this "sweet spot" of reliably performing meaningful units of work in the tens of milliseconds range (in contrast to the extremes that some narrow, specialized systems go to).

V. CONCLUSION

This survey has presented the research work in the field of Extraction-Transformation-Loading (ETL) processes and tools. Ab Initio and many different ETL development applications helps to optimize the process of ETL. Use of these applications have helped the organizations to grow and innovate. Proper and accurate data produced in this process can be used by organizations to build a better organizational structure and planning and management. ETL is the heart of any business organization.

REFERENCES

[1] Nutt, W.; Sagiv, Y.; Shurin, S.: Deciding Equivalence among Aggregate Queries, in: 17th Symposium on Principles of Database Systems (PODS'98, Seattle,Washington, USA, June 1-3), 1998
[2] Shim J.; Scheuermann, P.; Vingralek, R.: Dynamic Caching of Query Results for Decision Support Systems, in: Proceedings of the 11th International Conference on Scientific and Statistical Database Management (SSDBM'99, Cleveland, Ohio, USA, July 28-30)
[3] Barateiro, J., & Galhardas, H. (2005). A Survey of Data Quality Tools. Datenbank-Spektrum 14, 15- 21
[4] Carreira, P., Galhardas, H., Lopes A., & Pereira J. (2007). One-to-many data transformations through data mappers. Data Knowledge Engineering, 62, 3, 483-503
[5] Cui, Y., & Widom, J. (2001). Lineage Tracing for General Data Warehouse Transformations. Proceedings of 27th International Conference on Very Large Data Bases (VLDB 2001), pp.: 471-480, September 11-14, 2001, Roma, Italy
[6] Davidson, S., & Kosky, A. (1999). Specifying Database Transformations in WOL. Bulletin of the Technical Committee on Data Engineering, 22, 1, 25-30.
[7] Shu, N., Housel, B., Taylor, R., Ghosh, S., & Lum, V. (1977). EXPRESS: A Data EXtraction, Processing, and REStructuring System. ACM Transactions on Database Systems, 2, 2, 134-174.
[8] Trujillo, J., & Luján-Mora, S. (2003). A UML Based Approach for Modeling ETL Processes in Data Warehouses. In Proceedings of 22nd International Conference on Conceptual Modeling (ER 2003), pp. 307-320, Chicago, IL, USA, October 13-16, 2003.
[9] Vassiliadis, P., Simitsis, A., Georgantas, P., Terrovitis, M., & Skiadopoulos, S. (2005). A generic and customizable framework for the design of ETL scenarios. Information Systems, 30, 7, 492-525.