COMPUTER SCIENCE

ARRANGEMENT AND MODULATION OF ETL PROCESS IN THE STORAGE

Assistant ***Jala Aghazada***
Azerbaijan, Baku, Azerbaijan State Oil and Industry University

DOI: https://doi.org/10.31435/rsglobal_sr/31012020/6866

ARTICLE INFO

Received 10 November 2019
Accepted 12 January 2020
Published 31 January 2020

KEYWORDS

data storage, ETL process, data removal, data conversion, data loading, model of information movement.

ABSTRACT

Data warehouse (DW) is the basis of systems for operational data analysis (OLAP-Online Analytical Processing). Data extracted from different sources transforms and load in DW. Proper organization of this process, which is called ETL (Extract, Transform, Load) has important significance in creation of DW and analytical data processing. Forms of organization, methods of realization and modeling of ETL processes are considered in this paper.

Citation: Jala Aghazada. (2020) Arrangement and Modulation of ETL Process in the Storage. *Science Review*. 1(28). doi: 10.31435/rsglobal_sr/31012020/6866

Data storage concept, formed by Bill Inman (USA) in the 1990s of the last century, paved the way for further development and widespread use of OLAP systems [1]. The basis of this concept is the preparation of data for further analysis. The VA maintains problem-oriented, integrated, invariable data, and their chronological sequence is ensured [2].

Data assembled in DS for analytical processing can be removed from a variety of sources: databases operating in the field of the considered problem or OLTP (Online Transaction Processing) systems, electronic archives, MS Office documents, standard-reference files, etc. Because of the lack of connection between these sources, or because they are very weak, the data they provide have different structures and description forms. Therefore, it is necessary to convert data obtained from various sources, that is, conform them to each other, make them in the same format, eliminate duplications and error values. The data obtained after the conversion process is uploaded to the storage.

Since the ETL process is an integral part of DS operation, the development of ETL process is one of the most important issues in establishing the DS. The ETL process is characterized by the following properties:

1. The volume of data removed from input sources and uploaded to the DS is quite large (up to gigabytes);
2. The cycle of ETL process is determined not only by the demands for data relevance, but also by the measures of the data portions uploaded to the DS;
3. Metadata characterizing DS at different stages of ETL process is formed and data quality is ensured;
4. The process of data entry into DS must be controlled to prevent data loss during ETL process;
5. The ETL process should ensure the data to be restored in case of an accident without losing it.

The issues arising from these properties should be taken into account in the arrangement of the ETL process.

Methods for arranging the ETL process can be distinguished by the following features:
- the site where data conversion is carried out;
- the person who the data is removed by from the source;
- the site where the data removal process is conducted.

The ETL process can be performed in three ways depending on the site where the data is to be converted [3]:
1) use of intermediate memory;
2) non-use of intermediate memory;
3) data conversion on a DS server.

The commonly used method of arranging the ETL process is based on **the use of intermediate memory** (Figure 1).

The data removed from primary sources is recorded onto intermediate memory. DB or seperate files are used as intermediate memory. Conversion and cleaning procedures are performed on the data stored in the intermediate memory and then it is uploaded to the DS.

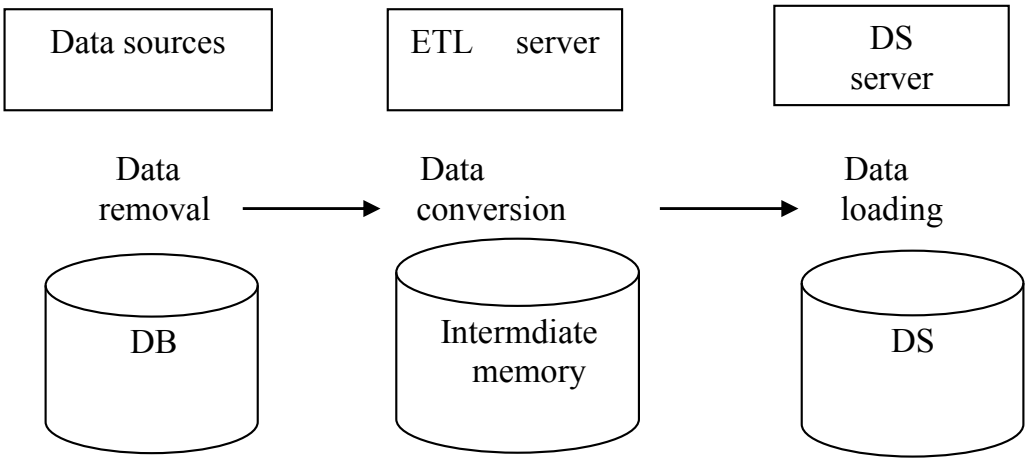

Fig. 1. Arrangement of ETL process using intermediate memory

An alternative way to the considered one performs **converting data in the ETL server RAM**, and the results are directly uploaded to the DS (Figure 2).

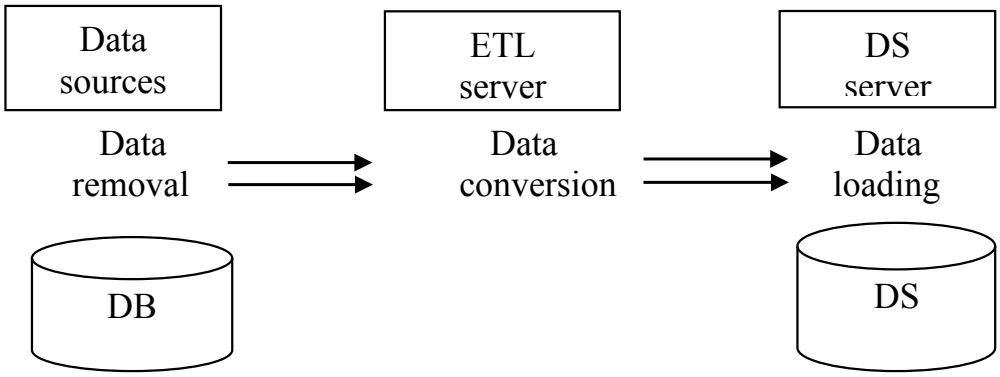

Fig. 2. Arrangement of ETL process without using intermediate memory

Conversion of data in RAM is faster compared to the first method. However, the restriction on RAM limits the size of the data portion uploaded to the DS. Intermediate memory is used when the size of portion is large.

Another way to arrange the ETL process is to perform **data conversion in the DS server** (Figure 3). In this case, data conversion is carried out during the process of their uploading to the DS. The application of this method is determined by computing capabilities of the DS server.

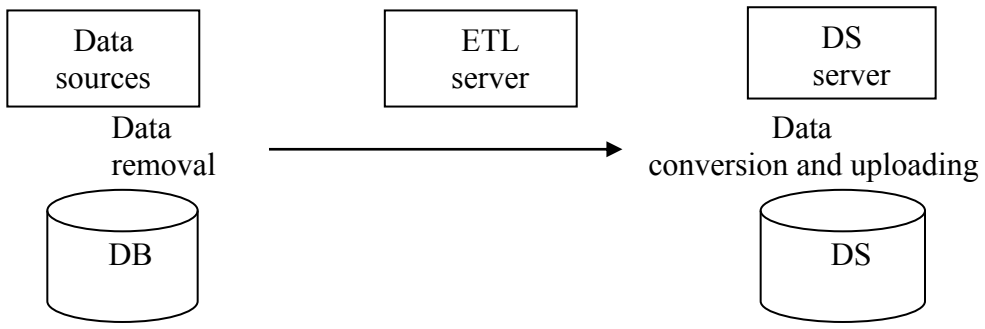

Fig. 3. Arrangement of ETL process by perfoming data conversion on DS server

According to the person who the data is removed from sources by, the ETL process can be arranged by the following ways:

1. The ETL-server periodically connects to the data sources, carries out surveys in them, makes the results of positively answered surveys, and stores them for further processing.
2. Special programs (triggers) installed on data sources observe the changes in the data and place the changed data in a separate spreadsheet. This data is then exported to an ETL server.
3. Specially designed software periodically carries out surveys in data sources and exports data to ETL server.
4. Log-sources (journals) of the data sources are used. These journals store all the transactions implemented for changing data. Changed data is removed from log-journals and stored on the data server. They are then transmitted to the ETL server.

According to the site where the process of data removal is carried out, the ETL process can be implemented by the following ways:

- The ETL process is performed on an ETL server located between the data source and the DS server. In this case, the ETL process does not use the computing resources of the DS server and the data server;
- ETL process is performed on DS server. In this case, the DS server must have sufficient large disk memory;
- ETL process is performed on data source servers. In this case, any changes to the data are immediately reflected in the DS. This method is used in DS operating at real-time mode.

Thus, when designing an ETL process, the site and ways for conducting the ETL process is determined according to the demands on DS activities and based on this, the hardware and software tools are selected for the ETL process.

Since the development of the ETL process is directly related to the data description model in the DS, first that model must be identified. Because OLAP systems created for large enterprises have a central DS with large capacity, the relational model (ROLAP) is often used to describe data.

The development of the ETL process is carried out in the following sequence [3]:

- ETL process planning;
- completing dimension spreadsheets;
- completing the spreadsheets of facts.

The ETL process planning is carried out in two stages: first, summarized plan is established, and then that plan is detailed.

The summarized plan lists the data sources and displays the planned data in the DS. The data sources are determined based on business requirements for DS. Data sources can vary greatly: from databases and text files to SMS messages. This feature complicates the problem of data conversion.

The design of the summarized plan starts after the ROLAP model of the database is established. Data sources- spreadsheets are defined for each table of the ROLAP scheme. At this stage, any incompatibility encountered in the coding scheme and data assignment should be noted.

Detailed planning of the ETL process depends on the application of the selected ETL instruments. Currently, a number of ETL tools have been developed by companies specializing in DS (IBM, Oracle, Microsoft, etc.) as well as other software manufacturers (such as Sunopsis). Therefore, the problem of selecting suitable ETL instruments should be solved before detailed planning.

ETL process software can be developed manually or using specific ETL tools. Each of them has its positive and negative features. In recent years, the second method has been more preferable.

The process of **completing dimension spreadsheets** is carried out depending on whether spreadsheets change or not change over time.

One of the major issues in the timetable that remains constant over time is the selection of the spreadsheet's primary key. Selection of the key is carried out by the designer based on the analysis of the data sources. The second major task is to check whether there is "one-to-one" or "one-to-many" relations in the dimension table. Regulation is usually used to conduct such an inspection.

Variable dimension spreadsheets are then reviewed, the type of change is defined, and the working procedures are defined in these spreadsheets.

Dimension spreadsheets are loaded either by recording them totally, or by loading only the changes into the dimension spreadsheets.

The following problems are solved in the process of **completing the spreadsheets of facts**:

- Analysis of spreadsheets of facts;
- loading spreadsheets of facts;
- analysis of the installed units;
- loading of units.

Spreadsheets of facts are loaded in two ways: primary loading of spreadsheets and periodic loading of changes. Due to the high volume of data, there is a high probability of errors at primary loading. Therefore, it is important to evaluate which type of loading is most suitable for spreadsheets of facts.

In the ETL process, the dimension tables must be updated before the spreadsheets of facts, as the newly loaded facts are placed according to the corresponding rows of the dimension spreadsheets [5].

As mentioned above, the ETL process consists of 3 components: data removal, data conversion and data loading.

The purpose of **data removal** process is to periodically remove relevant data from the data sources. This process can be carried out by the following tools:

1) Removing data using programs based on SQL commands. These programs operate in connection with other applications of the data source systems.

2) Removing data using the mechanisms included in the DBIS for import/ export of data. The use of such mechanisms accelerates data removal.

3) Removal of data via specially designed programs. This option is relatively expensive.

The data removal process can be conducted once a day, week, or rarely a month. Some systems require data removal to be conducted on a real-time scale: for example, in systems analyzing stock exchange operations or in systems of telecommunication field.

Data removal process can be conducted both in the OLTP and OLAP environment.

The data conversion process mainly involves the following operations:

- Conversion of data types: change of data code (for example, to "UniCode"), conversion of date and time into appropriate format, etc.;
- Conversion related to normalization or denormalization of data schemes;
- Converions of keys;
- Conversions for data quality ensurance in VA.

The date is "**cleaned**" to ensure its quality. The cleaning process involves: conforming data formats, coding data, removing unnecessary attributes, replacing codes with values (for example, replacing an object code with its name), combining the data obtained from various sources (for example, collecting all data about the products released by an enterprise) under the common key and so on.

Cleaning the data is divided into the following types:

- Conversion and normalization of data (text with the same code, time expression with the same format, etc.);
- standardization of names' spelling, address descriptions, removing duplications;
- standardization of tables' name, indexes, etc.;
- business-based cleaning in the area under consideration.

The following factors should be taken into account in the procces of **loading data**.

1) Due to the high volume of data uploaded to the DS, it is important to speed up the loading process. For this purpose the followings should be taken into account:

- the speed of data loading is low by using SQL commands. Therefore, it is more profitable to carry out the loading process via import/export tools included in the DBIS;
- the speed of loading table indexes is low, so, it is often advantageous to rebuild indexes;
- paralelizm should maximally be used when loading data, for example, spreadsheets of facts and dimension spreadsheets can be loaded at the same time;
- reference completeness must be provided during the process of loading, and the assembles must be installed and uploaded at the same time as the data they contain.

2) Evaluation of data loading productivity is carried out by the DS administrator via the procedures included in the applicable DBI.

The construction and application of a model that graphically illustrates the removal, conversion, and loading of data is of great importance in the design of the ETL process. CASE tools are widely used to construct such models [4,314-320]. The system of "Power Designer", a product of Sybase, took into account the information movement model (Information Liquidity Model-ILM) for this purpose.

The information movement model includes three types of diagrams: information movement diagrams, data conversion diagrams, and management diagram of conversion.

Information movement diagram is a high-level diagram that enables to model the ETL process with the following general concepts:
- data input sources: databases, XML documents, business processes, unstructured files, etc.;
- conversion process: illustrates the site where the conversion is carried out;
- data output sources: database store or database.

Data conversion diagram is a low-level diagram that describes specific conversion issues: how data is removed from the input source, how it is converted, and how it is loaded into output sources.

Conversion management diagram is a low-level diagram that shows the sequence in which the conversion issues are implemented.

Let's consider the information movement model as a sample of an information system (IS) designed to collect, store and process data reflecting the operation of an enterprise in a manufacturing enterprise. The information movement diagram that forms the basis of the model is shown in Figure 4. The input sources of data are the operational database (ODB), which reflects the operational activities of the enterprise, HTML and XML documents which are implemented in the Internet/ Intranet environment, standard-reference database (SSDB), and other sources (DM). Once the data removed from those sources go through the conversion process, the data is uploaded to the output sources, data store (DS) or database (DB).

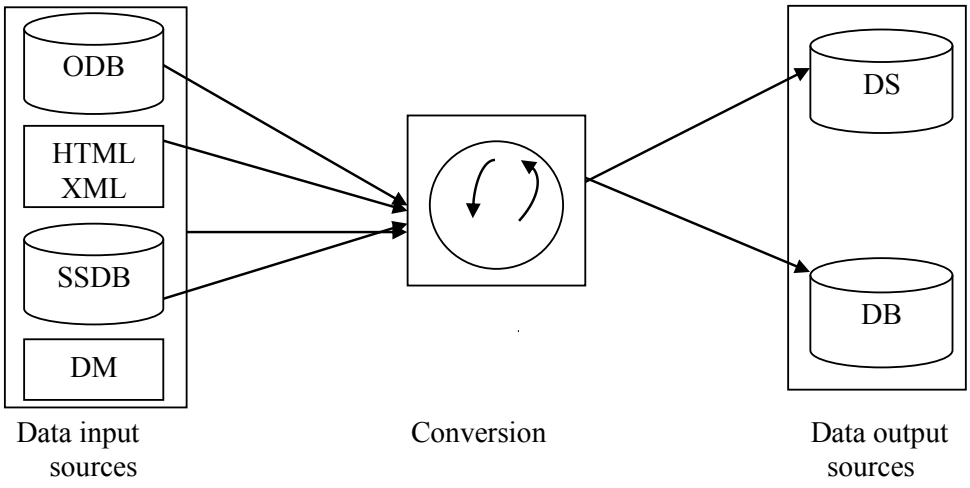

Fig. 4. Information movement diagram

The low-level data conversion diagram is shown in Figure 5. In most cases, the conversion process consists of three consecutive operations:
- Combining the data removed from various input sources and bringing them into a common format (BUF);
- data cleaning (T);
- data adjustment (N).

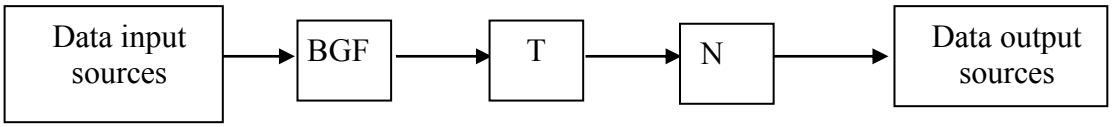

Fig. 5. Data conversion diagram

Another diagram showing the sequence of low-level conversion management is shown in Figure 6. Here the following abbreviations are accepted:

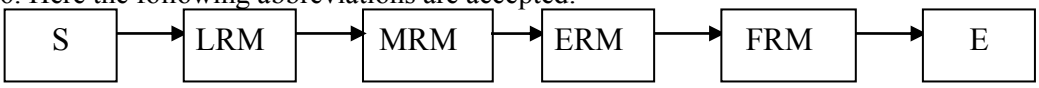

Fig. 6. Conversion management diagram

S-start, LRM-labor resources merger, MRM - material resources merger, ERM-energy resources merger, FRM-financial resources merger, E- the end of conversion process.

Conclusions. The methods of arranging ETL process vary depending on where and by whom data removal and conversion processes are carried out. The development of the ETL process starts with its planning and then the DS spreadsheets of dimensions and facts are completed. One of the important issues in the design of the ETL process is the construction of a graphical model that reflects the information movement via CASE tools.

In the article, these issues are practically investigated and explained.

REFERENCES

1. Inmon W.H. Building the Data Warehouse, 1992.
2. Karimov S.G. Information systems. Baku: Elm, 2008.-616 p.
3. Ostrovsky E.V. The procedure for developing ETL processes. citcity.ru/ 11144
4. Karimov S.G. Management information technologies and corporative information systems. Textbook-Baku, 2010.-426 p
5. Implementation of the ETL subsystem of corporative data storage. www.pry-exp.ru/dwh/structure_of_etl_process.php