

Implementing Business Intelligence System - Case Study

Yasser AL-HADAD¹, Răzvan Daniel ZOTA²

Faculty of Cybernetics, Statistics and Economic Informatics, Department of Economic Informatics and Cybernetics, Bucharest University of Economic Studies, Romania

¹dukeyasser10@yahoo.com, ²zota@ase.ro

Abstract: *Understanding and analysis data is essential for making decision within a system. Any analytical tasks can be implemented directly by the transactional system but it becomes more difficult as the transactional system grows. Analytical systems and their extension appear as a solution for complex and large datasets. We think that it's time for medium companies to get the benefit from such systems as analytical systems become more variant and in hand for every possible user. In this paper, we propose an architecture of analytical system that can adapt and integrate with existent transactional system of timber export company. The proposed analytical system should have the ability of implementing the tasks required by the decision makers of the system. Also, we try to explore the ability of SQL server of implementing our proposed architecture.*

Keywords: *BI(Business intelligence) , DSS(Decision support systems), DW(Data warehouse), OLAP(Online Analytical Processing), ETL(Extract, transform and load), SAS(SQL server services)*

1 Introduction

Business Intelligence (BI) is high business application tools [1] used for collecting, cleansing, processing and analyzing data. evaluating and understanding the results is essential for the efficiency of decisional system. Created knowledge are collected from both internal and external sources [2]. and accumulation them can lead to improve corporate profitability [3]. BI tools becomes more variant and easy to use [4] and their solutions are ranked as one of most important technological stuffs by chief information officers [5]. Investments in BI focuses on achieving business targeted and increase return on investments [6]. So BI is an entire concept that is used for implementing a decision support system (DSS). DSS can be described as the next generation systems that follow transactional and operational systems [7]. Collected data and maybe the resulted information are stored in data warehouse (DW). It is one of the basic component of DSS and BI systems. It includes all spread data over the organization in single pool [8]. So the

existing of DW was necessary as the transactional databases were unable to store the accumulated historical data [9]. DW can be accessed by proper data-analyzing to analyze and store required data [10,11]. DW help to achieve strategic business objectives by offering clean and homogeneous data in real time to support the analytical processes [12,13]. there are many architecture that are suit the design and implementation of DW [10,14,15]. The common idea is to load the propagated data from operational DBMS (database management system) and other sources in DW using a special tools called ETL (extract, transform and load). ETL is the most usable tool for integrating data into data warehouse. Many architectures are proposed for implementing ETL. It was used to integrate the entire data source in [16] even if it is too big. In [17], the data are processed in local repositories before integrating them in the global data warehouse. The difference between ETL and ELT (extract, load and transform) is presented at [18]. In [19], an architecture is proposed for integrating and validation data from multiple data sources. Other approach is presented in [20] that improves the

performance and interacts more with users. At [21], the metadata of ETL is discussed to enable the interoperability between the systems. The incomplete data are treated as grey data and special procedures are proposed to deal with them [22]. Data integration includes data from a single or multiple sources. The problems of data integration from multiple sources are overlapping data and matching records belongs to same entity. The challenges of data integration from single source using backup data files is similar to data integration from multiple sources. Combining integration step with cleansing step can increase the performance. on the other hand, these transformation lose the legacy data form. Legacy data is important for reevaluation of data cleansing process or backflow of cleaned data.

Timely decision making becomes difficult due to the inefficiency of transactional databases to handle the amount of information access, retrieval, update and maintenance. This shortness impact every industry [12]. OLAP (Online Analytical Processing) appears as one of solutions for resolving problems of operational database. Upon the OLAP, many research is built on data cube operator [23]. Also various algorithms for supporting dimension hierarchies [24]. OLAP is a technology that offers users to perform analysis on detailed data from multi prospects. It adopts multidimensional approach for implementing the analytical database. Analytical database analyze complex relation between millions of records for identifying trends and patterns. It stores static data type such as derived and calculated data for providing them in real time whenever there are requested by managers. The perspective of data cube and multidimensional database was adopted by many corporations such as Microsoft [25]. Now, SSAS(SQL Server Analysis Services) is the leader for business logic analysis in services and

software. Its integrated platform is one of largest framework that implement the concept of BI [8]. SAS solution provides full end-to-end technology. Also for data quality and data access, it ensures that large volumes of data are processed and transformed into accurate analysis information and reporting. One of the advantages of SAS for many users are reporting and graphical representations. It offers dynamic views of information, that can be displayed in many forms such as a table, a report and a chart. The main requirement of potential users of Business intelligence solution is to perform analysis of data and testing the results. SAS Enterprise BI Server provides many tools for analysis and forecasting, which is essential to solving problems in order to make the company more competitive. One of the advantages of SAS Enterprise BI Server is the features of web portal which provides some functionalities that enable users to manipulate the contents and layout of the interface. Also it is offers features Web-based distribution and reporting. Users can manage and build reports using custom and exporting them to EXCEL or PDF files. There are functionalities that allows users to observe the results and patterns that cannot be observed in a traditional chart using more interactive methods such as 3D videos and tables with bubbles containing data presentations. SAS product offers an integrated platform that store data and reports. Also its predefined functions allow the developers of IT departments to focus on other tasks such as security of BI, implementation and maintenance.

The objective of the application is to build a decision support system (DSS) for inventory management. The system will be implemented for a medium level company that exporting timber. The objective of the project is to find an appropriate solutions for the firm size and to be adaptable for the business field and decision maker requirements. It is expected that the proposed system will provide timely relevant information to meet the needs of

decision makers so they can respond quickly to market fluctuations through readapt policies and strategies.

2 Development methodologies

Project development is a process that consisting of several different stages. Each stage has its own requirements and targets. Depending on the project type, certain stages gain additional attention in the overall effort. There are many methodologies types that can be used for project development. Software development methodologies can be defined as set of guidelines and rules that are used in the process of each stage. Each of these methodologies has its Characteristics, strengths and weaknesses

[26]. For our case, we have used Prototyping methodology, see figure 1. Prototyping is a methodology that entails building a prototype or demo version of the software product that contains the critical functionality. Demo version includes sufficient information to build a prototype and should be built fast. It is used to refine specifications and it acts as baseline for entire project. it makes the communication process better between project owner and project team [27]. The main characteristic of prototype methodology is that project owner and users is actively involved by evaluating prototypes that are valued over writing specifications and meant to be discarded later.

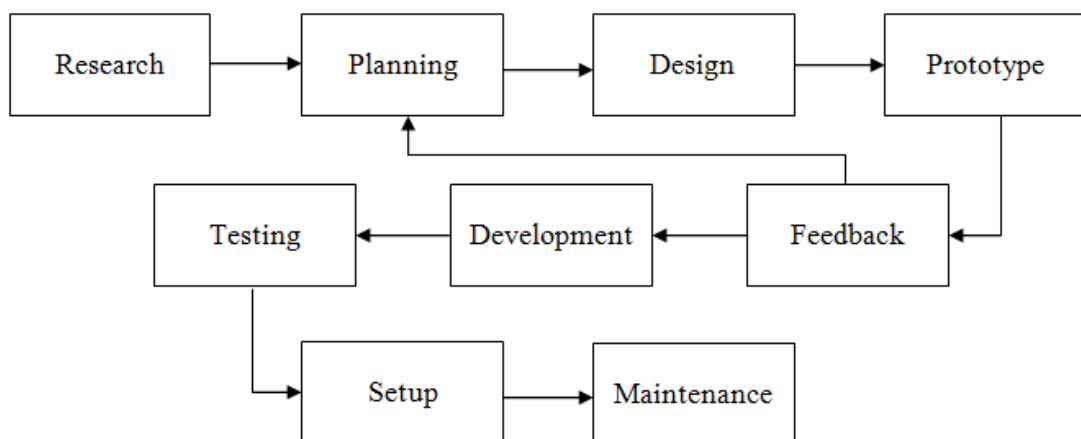


Fig. 1. Prototyping methodology

Demo versions can contain one or more prototypes of the initial models of the software product. In the other hand, Strengths points of this methodology is crucial for good implementation of software project. It involved the potential users of the system in process of implementation and improve their experience. Also the early feedback from users and the project owner help to early identification of any redundant or missing so an accurate identification of software requirements is guaranteed. The main Weaknesses of the prototype methodology is the costs generated by increased programming effort due to building the prototype, but this

Weaknesses can be ignored because SAS platform that is used for implementing our project guaranteed that the programming effort by using SAS predefined functions and layouts will be reduced so the process of building any prototype will be easier.

3 Staging method

Staging method removes the performance overhead involved in cross-database joins by creating staging tables in the warehouse database. Staging method extracts data from the sources and loaded them into data warehouse without any transformation. It just makes another copy of legacy data content and loading them into analytical system in order to make them available for

any transformation processes that can be performed later. Obvious method combines both extract and transform steps in the same flow. It is more efficient and no need to store intermediate results. For an mid-level enterprise, the entire size of data is not too large to generate any performance difficulties, so the processing efficiency of the system is not

the most important factor of the system. In our case, the cost of developing and maintenance is much important. as it is seen in figure 2, the flow of staging method is divided in many small steps in order to ease and simply the process of analyzing and implementation. Using obvious method, data are processed directly and loaded in the analytical database or OLAP.

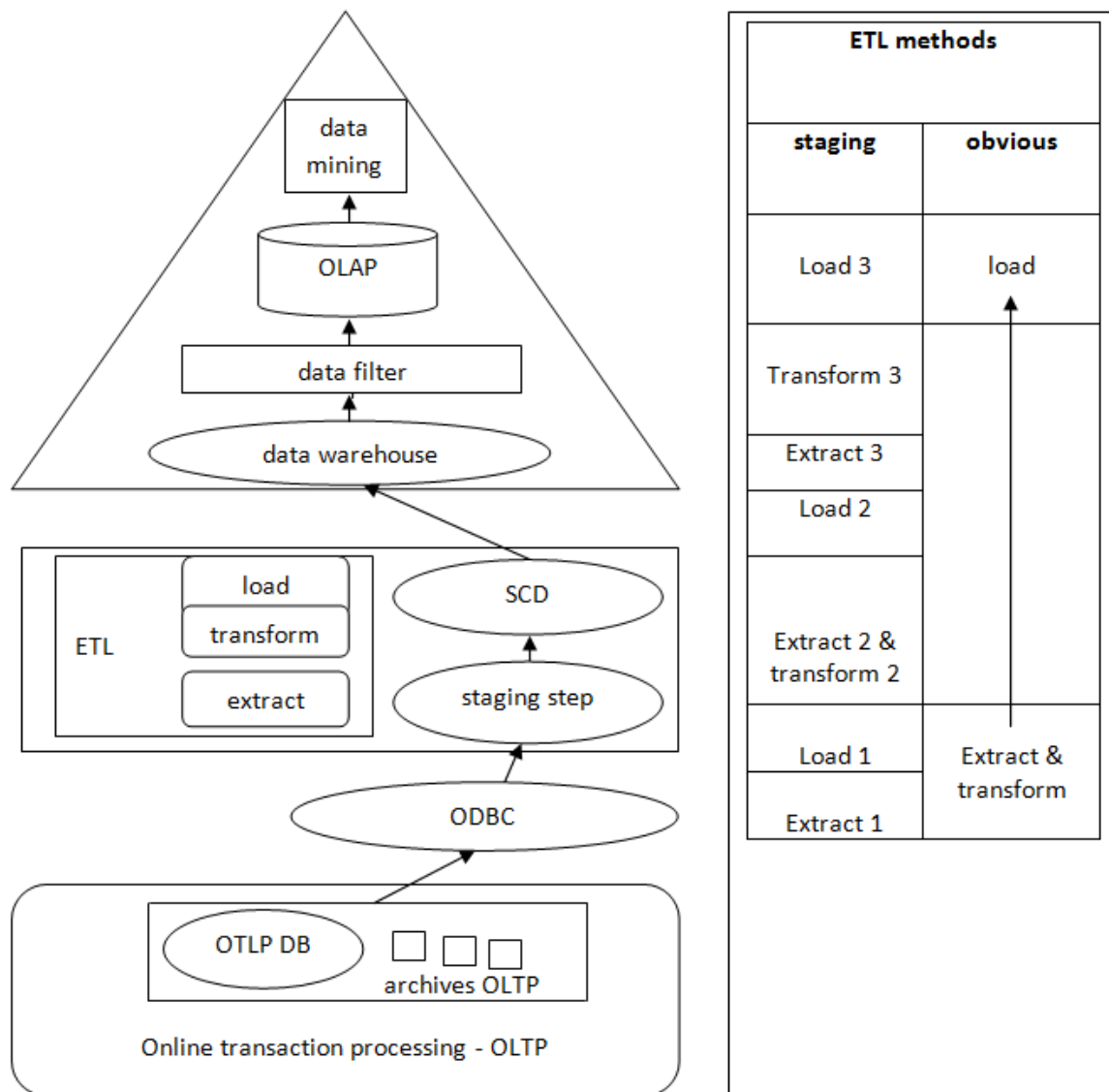


Fig. 2. DSS framework components

4 Case study

In the following section, we are going to propose DSS for our case study. The requirement of system will be analyze according to standpoint of intermediate dealer (quality, production capacity, price, payment conditions , etc). So there

is no need to analyze the process of timber production as the modelled system can validate itself by using data in the transactional database where all the specific and phenomenon of the this field business are hidden in the data. A database for an timber export company with annual turnover

of over two million euro will be used for validation the proposed DSS.

4.1 Exiting transactional system

As the volume of data and activities increase, more challenges appear with respect to process and manipulate collected data. Existing informational system is designed to meet the requirement of transferring big part of processing data burden to an centralized system for operating the firm activities. It replaces the older system which were compose by many uncorrelated systems and procedures. That limitation was preventing the organization to expand their activity. Existing system has proposed to meet the requirement of company. Standardized solution is adopted for such modern system, but other special solutions that adapt some specific requirements.

4.2 Technical equipments

Transactional system is implemented using PHP scripting language, web interface is running by apache server and MySQL server for database. XAMPP server is used to run the application and database. The version XAMPP 1.6.3a is kept up to now to guarantee the working of all facilities of the system. XAMPP server database and application are installed at every unit to enable operating the system at offline mode so the server is installed at every unit. It should be configured properly at every unit to assure that is use an un busy socket and an existing hard disk partition and the platform has the full access to use the partition. In order to operate the system at the offline mode, users has to synchronize their local database using defined procedure for updating database. Synchronizing database is performing via archived files that include the latest data. For supporting the partners of corporation and agents who don't have access to informational system, system offers the

possibility of exporting its data using portable document format (PDF).

4.3 Instruments of proposed system

SSAS(SQL Server Analysis service) is adopted for implementation the proposed analysis application. It offers the main methods and algorithms for analysis purposes. The friendly user interface facilitate the use of its services. Also the wide range of graphical representation for data and information that are available in their services makes viewing and understanding the resulted data more easier. The step of building the model should be followed by data injection step. All the resource data are available in the MySQL server that are not accessible directly for an analysis model. Any input data should be locating in the engine database SQL server. MyODBC driver connector is used to make the data in MYSQL server accessible for integrating them in SQL server. ODBC is a standardized API (application-programming interface) that enable the client-side application to connect to one or multiple databases. MyODBC driver is a member of MySQL ODBC. It provide access to MySQL database throw the standard ODBC. It offers standard interface using driver-manager based and a native interfaces. For Unix and Mac operation system, native MySQL network can be used to communicate with MySQL database. Installing of MySQL connector ODBC is necessary for Windows and application that use the ODBC interface. Figure 3 shows the instruments are used for implementing the application as following:

- *MySQL Server*: represents the platform that hosts the current transactional database. It also used to reload the archived data into database in order to send them to SQL server platform;
- *MyODBC connector*: offers the possibility of accessing MySQL databases directly using SSIS (SQL server integrating services). Data and its structure can be viewed and imported in the SQL server. So there is

no need to export any standard format to be imported in destination;

- *SSIS (SQL server integration services)*: offers a wide range of components that can be used for interrogating and integrating the data. The graphical user interface makes it easy to build and configure an integrating services project. This types of projects can deal with many external data resources;
- *SQL server database engine*: The data destination of any integrating services project is in SQL database

engine. It is important to load all necessary data of analyzing process to make it accessible to OLAP process and analysis services;

- *SSAS (SQL server analysis services)*: design prediction or classification models for the entities available in the database for supporting decision system or any simulation system. Furthermore, OLAP process can contribute in implementing analytical database for generating complex reports and preparing data for SSAS data initialization;

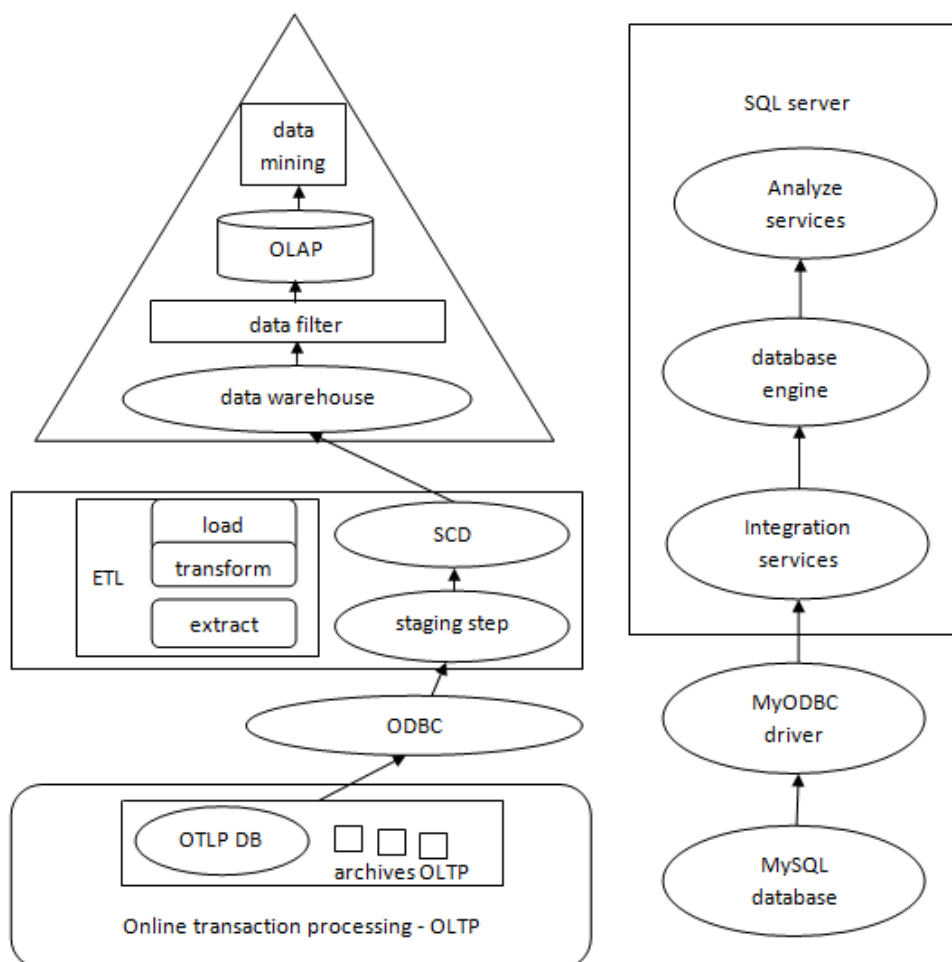


Fig. 3. DSS framework tools

4.4 Application components

Any analysis model can be configured directly from structures interrogating in the integrating services projects. Obvious method combines both extract and transform steps in the same flow and load

the resulted data in the final destination model. It can create some data redundancies and replicate some common functionalities. Top-down methodology involves breaking down of a system to get insight into its compositional sub-systems.

It assure that any additional functionalities can be added with less effort. Staging method avoids any possibility of replicating common functionalities by using some intermediary steps. figure 2 shows the difference between the staging and obvious method. The components of staging methods as it seen in the figure 2 can be summarized as follow:

- *Extraxt1 & load1*: includes copying data in the intermediary database without change its structures or values;
- *Extract2 & trasform2 & load2*: extract any new data instances in staging database to load it in the data warehouse. This stage can include data transformation to adapt the data to the requirements of data warehouse;
- *Extract3 & transform3 & load3*: Extracted data from data warehouse are filtered and transformed to meet the requirement of analysis system;

4.5 Data source integration

Backup data file represents a copy of database. It is generated in certain time and has the same data as operational database (ODB) at that period. This content also includes erroneous data that exists at ODB. These type of files can be used in case of inexistent a system for data archiving and enable recovering data in case of any system failure. OTLP(online transaction processing) or ODB is a database that is designed and normalized to avoid redundant data and facilitate the operations (insertion, deletion and updating) captured from transactional system, it is used to store data using the relational database technology to ensure the integrity. It dedicates to serve the transactional systems so it just include the current activity without historical data.

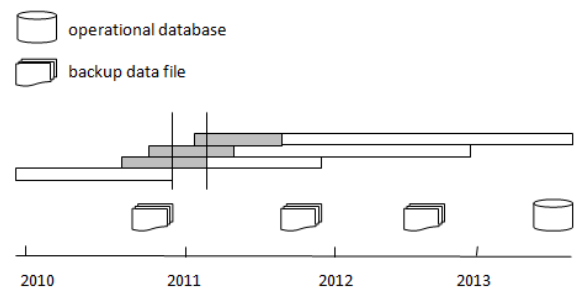


Fig. 4. Recovering historical data using backup data files

figure 4 shows components used for building data warehouse for our case study. OBD and three backup data file. It shows data content for every source and its period. The grey portion indicates that data is partial and not complete. Older backup file is used for recovering data of the grey zone. The example presented in the figure 4 shows that the entire historical data can't be recovered so it remains some periods where data is not complete. Some procedures should be taken for dealing with this situation. ETL is used to update the data warehouse by capturing changes data from transactional system and loading them in data warehouse. The process of elimination method is used in SSIS by the component slowly changing dimension (SCD) that is shown in figure 3. It is dealing with insert and update commands properly but it does not process the delete command so all the deleted data captured from backup data files are migrated to data warehouse. Most of deleted records represent historical data and should be kept in data warehouse. The deleted records can be classified as following:

- *Missed historical data*: are valid data that should be kept in the data warehouse to serve the analytical purposes. Missed historical data represent the instances of historical data that cannot be recovered or not completed;
- *Cancelled data*: present removed data from the transactional system that should be identified and isolated. These records were removed from the transactional system because of

correcting users mistakes, duplicate insert, eliminate transaction such as an cancelled selling transaction. We can mention to these records as cancelled data;

4.6 Data quality problem

Last step of designing data warehouse is to define the rules used for data validation. Many Erroneous data can be revealed at stage of data integration. Determining the reason of appearing erroneous data can ease the task of cleansing them. By analytical point of view, a classification of erroneous data is proposed as following:

- *Platform error*: any erroneous data generated by system without intervention of users can be considered as system error, the reason of existing such these erroneous data is due to platform reliability;
- *System error*: error generated by transactional system due to problems of modelling or programming the system;
- *Users mistakes*: mistakes that are generated and not affect the transactional application. such these mistakes should be identified cause they can affected the analytical database. An example of these errors is reversing the dimensions values of length and width. This reversing is not affect the total quantity of materials but it can affect any clustering or classification method;
- *Cancelled data*: mistakes records that registered in system and removed later, the gap time between inserted and deleted operation enable to some of these records to be captured and inserted in data warehouse at the integration data step;

Some validation data rules have been defined to capture and isolate these data. The validation rules are classifies in two categories, unconditional and conditional validation rule. Unconditional validation rule: where there are two related table,

validation of the first table is dependent on the second table. Unconditional validation means that no need of pre validation for the second table. We mentioned the second table as validation table. In case conditional validation rule, the data of validation entity should be verified and approved before using it in validation another entity. This type of rule requires some pre-validation. Different rules require different pre-validation steps. Thus, a flow that shows dependencies between rules is important to indicate the order in which these rules should be executed.

5 Conclusions

Business Intelligence is concept that include a set of techniques and methods that aim to configure high level tool that served the analytical purposes in order to support the decision maker. Tools of business intelligence become more variant and in hand for every possible user. So it is time for medium companies to get the benefit from such solution. Medium companies have a large data amount that need to be analyzed but it cannot invest a lot. At this paper, we try to build a decision support system tailored for timber export company. Prototyping methodology is used as it suitable for innovative projects where no previous examples of this type exist. There are many varies architectures for development DSS. Our proposed system is divided into small and isolated tasks or components. This enable every task to be smaller and specific but the system is more flexible for maintenance and evolve. SQL server services (SAS) is one of largest companies that also offers BI solutions addressed to medium companies. SAS has a bunch of predefined functions and layouts that decrease programming effort, so implementing the prototyping methodology becomes easier. Building data warehouse requires integration and validating data from different sources. This task includes several steps or processes such as extact, transform, filtering, cleaning and load. Data cleansing is one of most difficult tasks.

Different approaches and techniques are proposed for validation and cleansing data from single or multiple sources.

In this paper, It was solved the problem of integration from single source but by using backup data files and not archiving system. The accuracy of integrated data can not be guareented 100% especially when the legacy data is not complete. The purpose of data warehouse is to serve applications dedicates to solve analytical problem. Thus, it is important to adopt solution that balanced between the quality and cost. Automat solution is prefered in data cleansing than manual working. In our case stude, we apply some tehniques for filtering and cleaning the historical data for backup data files problem.

Acknowledgment

Part of this work is done under the auspices of the doctoral studies within the Doctoral School of Economic Informatics, Bucharest University of Economic Studies.

References

- [1] A. Sohollo, "Using Business Intelligence in IT Governance Decision Making," in *Governance and Sustainability in IS*, M. Nüttgens, A. Gadatsch, K. Kautz, I. Schirmer and N. Blinn, Eds. Germany: Springer, IFIP AICT 366, IFIP International Federation for Information Processing 2011, Part 1, pp. 3–15.
- [2] J.A. O'Brien and G.M. Marakas, *Management Information Systems*, 10th ed. New York, U.S.A.: The McGraw-Hill Companies, 2011.
- [3] R.L. Sallam, J. Richardson, J. Hagerty and B. Hostmann. (January 2011). *Magic Quadrant for Business Intelligence Platforms*. Available: [http://www.board.com/download/press/E N/Gartner_BI_MagicQuadrant_2011.pdf](http://www.board.com/download/press/E%20N/Gartner_BI_MagicQuadrant_2011.pdf)
- [4] Gartner. (September 2012). *Business Intelligence, Mobile and Cloud Top the Technology Priority List for CIOs in Asia: Gartner Executive Programs Survey*, Available: <http://www.gartner.com/it/page.jsp?id=2159315>.
- [5] Gartner. (April 2012a). *Gartner Says Worldwide Business Intelligence, Analytics and Performance Management Software Market Surpassed the \$12 Billion Mark in 2011*, Available: <http://www.gartner.com/it/page.jsp?id=1971516>.
- [6] H.J. Watson, C. Fuller and T. Ariyachandra, "Data warehouse governance: best practices at Blue Cross and Blue Shield of North Carolina," *Decision Support Systems*, Vol. 38(3), pp. 435-450, 2004.
- [7] T. Ariyachandra, H. J. Watson, "Key organizational factors in data warehouse architecture selection", *Decision Support Systems* 49, 2010, 200–212.
- [8] T. R. Sahama, P. R. Croll, "A Data Warehouse Architecture for Clinical Data Warehousing", in Roddick, J. F. and Warren, J. R., Eds. *Proceedings Australasian Workshop on Health Knowledge Management and Discovery (HKMD 2007) CRPIT*, 68, pages pp. 227-232, Ballarat, Victoria.
- [9] B.A. Devlin, P.T. Murphy, "An architecture for a business and information system," *IBM Systems Journal* 27 (1) (1988) 60 – 80.
- [10] W.H. Inmon., "DW 2.0 Architecture for the Next Generation of Data Warehousing", *DM Review*, Apr 2006, Vol. 16 Issue 4, p.8-25.
- [11] W.H. Inmon, "Building the Data Warehouse", *Third Edition*, York: John Wiley & Sons, 2002.
- [12] H.-G. Hwang, C.-Y. Ku, D. Yen and C.-C. Cheng, "Critical factors influencing the adoption of data warehouse technology: a study of the banking industry in Taiwan," *Decision Support Systems* 37.1 (2004): 1-21.
- [13] Nilakanta, Sree, K. Scheibe, and A. Rai. "Dimensional issues in agricultural data warehouse designs,"

- Computers and electronics in agriculture* 60.2 (2008): 263-278.
- [14] W.H. Inmon, "Building the Data Warehouse," Wiley, New York, 1996.
- [15] R. Kimball, "The Data Warehouse Toolkit", Wiley, New York, 1996.
- [16] C. BOJA, A. POCOVNICU and L. BĂTĂGAN, *Distributed Parallel Architecture for "Big Data"*, Informatica Economică, Vol. 16, 2012.
- [17] Q. Chen, U. Dayal and M. Hsu, *A Distributed OLAP Infrastructure for E-Commerce*, HP Laboratories, Palo Alto, California, USA.
- [18] R. Davenport, *ETL vs ELT*, Insource IT Consultancy, Insource Data Academy, June 2008.
- [19] E. Rahm and H. H. Do, *Data Cleaning: Problems and Current Approaches*, IEEE Data Engineering Bulletin, Vol. 23, 2000.
- [20] V. Raman and J. M. Hellerstein, *Potter's Wheel: An Interactive Data Cleaning System*, 2001.
- [21] H. H. Do and E. Rahm, *On Metadata Interoperability in Data Warehouse*, Techn. Report, Dept. of Computer Science, <http://dc-pubs.dbs.uni-leipzig.de/files/Rahm2000DataCleanin gProblemsand.pdf> [May 3, 2016].
- [22] S. F. Liu and Y. Lin, 2010, *Grey Information. Theory and practical Applications*, Springer-Verlag, London, 2010.
- [23] L. Cabibbo and R. Torlone, *Querying multidimensional databases*, Proceedings of the 6th DBLP Workshop, pages 253–269, 1997.
- [24] F. Dehne, T. Eavis, S. Hambrusch and A. Rau-Chaplin, *Parallelizing the datacube*, International Conference on Database Theory, 2001.
- [25] C. Diaconu, C. Freedman, E. Ismert, P.-A. Larson, P. Mittal and R. Stonecipher, N. Verma, and M. Zwillig. Hekaton, *SQL server's memory-optimized OLTP engine*, Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, SIGMOD '13, pages 1243–1254, New York, NY, USA, ACM, 2013.
- [26] M. L. DESPA, *Comparative study on software development methodologies*, Database Systems Journal vol. V, no. 3/2014.
- [27] J. E. Cooling, T. S. Hughes, "The emergence of rapid prototyping as a realtime software development tool", *Proceedings of the Second International Conference on Software Engineering for Real Time Systems*, 18-20 Sep. 1989, Cirencester, UK, Publisher: IET, 1989, pg. 60-64.

Yasser AL-HADAD has graduated the Faculty of Information Management at the Romanian American University in 2008. He received a master's degree in Economic Informatics from Romanian American University in 2010. Since then he is a PhD candidate, studying to obtain his PhD in the field of economic informatics.

Răzvan Daniel ZOTA has graduated the Faculty of Mathematics – Computer Science Section at the University of Bucharest in 1992. He has also a Bachelor degree in Economics, a postgraduate degree in Management from SNSPA Bucharest, Romania. In 2000 he has received the PhD title from the Academy of Economic Studies in the field of Cybernetics and Economic Informatics. From 2010 he is supervising PhD thesis in the field of Economic Informatics.