Approach for estimation of Data Ware House Projects/Data Marts using Function Point Analysis

A. Rakesh Phanindra, Dr V B Narasimha
Assistant Professor Information Technology, Institute of Public Enterprise, Hyderabad, India
Assistant Professor, Computer Science and Engineering, College of Engineering, Osmania University, Hyderabad. India

Abstract: An ETL Development, Enterprise approach for estimating data storing tools and software utilizing FPA. The data warehouse's problem of keeping the model which is enterprise is not immediately acknowledged in the Objectives. The "hidden" efforts of a development team in delivering a product's support architecture are sometimes overlooked. When compared to more traditional, and widely apparent, user functionality, data warehouses are poorly compared. In contrast to other traditional methods a Data Warehouse is a software system that utilizes other software systems as data sources rather than creating new data. In general, the environment would be more static. As a result, including Function Point Analysis into data warehousing applications has become more common as arduous as When compared to typical OLTP (Online Transactional Processing), data warehousing has some unique characteristics applications (processing) Data warehouse/data mart applications are unique in that they have certain properties, which includes the usage to solely utilize the developed systems to do query while they give out some report without doing data altering. The process of coming up with the new system usually has its basis on data that already exists from already existing systems which eases the process of coming up with fresh data. Estimating the cost of ETL tools becomes a problem and my paper tries to discuss what ETL tools are to be estimated during the process and how we can use the FPA to estimate the ETL tools. OLTP software systems have unique procedure of coming up with the system as compared to the ones that already exists. The systems uniqueness makes it easy in the adaption process. The measuring technique provided for standard OLTP software to ensure take into account some specific data warehouse/data mart features and provide estimation that will have more accuracy The proposed methodology aids in Using Functional Pointed Analysis to come up with Warehouse/Data Mart Projects, particularly for ETL processes, in a more precise way a method that is both traditional and systematic

Keywords: FPA stands for Functional Pointed Analysis, and ETL stands for Extract, Transform, and Load. DWH - Data warehouse, OLTP - Online Transactional Processing

I.INTRODUCTION

With Data Warehouse Systems, functional software metrics can be implemented in a unique way. The usage of a single standard, such as Function Point, creates major problems in terms of numerical size and functionality compatibility with older systems or paradigms.
Estimation of the amount of time and effort needed to finish the job As a result, unique criteria are required to separate the user's perspective, program boundaries, data, and transactional data components that make up such systems. The way boundaries are defined, in particular, could have a significant impact.

As a result, the measurement findings for a data warehouse project might be tampered with; as a result, massive, unacceptably large variations in effort, time, and cost projections for the given task project could be detected.

The goal of software functional measurement methodologies is to give an unbiased, technology-independent, and user-relevant assessment of software system size. [1]

The point method is a collection of best practices that may be applied to any type of domain or application. IFPUG counting techniques, despite their universality, are not always easy to implement in real-world or novel circumstances.

Aside from potential improvements in the expression of the practices, the basic concept is that the recognizability of the functional sizing components of software systems is dependent on the system user's point of view, which differs significantly from one area to the next. As a result, domain-specific counting criteria are required to identify how to adequately scale a certain system type (in our case, a data warehouse). Rather than being a new sizing method, the proposed methodology should be viewed as an "instantiation" of the generic method concepts in a specific context or domain.
If we adopt a specific measuring approach for a given area, however, we must recognize that general models cannot be used to estimate effort (for development or improvement activities) (unless we accept a high risk of big estimation mistakes). As a result, a generic effort model's "instantiation" will be used. [2]
Estimating the cost of ETL tools becomes a problem and my paper tries to discuss what ETL tools are to be estimated during the process and how we can use the FPA to estimate the ETL tools.

II.FUNCTIONAL POINTS ANALYSIS

Allan J. Albrecht created Function Point Analysis at IBM in 1979, and the International Function Point Users Group updated it subsequently.
Allan J. Albrecht provides the first definition:
From the research I have found that FPA is a useful relative in measuring the functionality degree provided to the people who uses the systems because they have got different functionalities that the system must perform.
The FPA codifies a method for scaling software work items based on their functional requirements [3]. This work product will be the result of future software development and enhancement efforts. The software gets moved to the production application once a project is completed. It assesses the function from a given point view of the user, i.e., what the user expects and receives.
The following are the steps for creating and upgrading a data warehouse database:

• Operational data is moved into the EDW during the Extraction phase (or an independent DM). • The data storage structure is transformed during the transformation process, and the operational data can be saved as records in an RDBMS table or as flat files with delimiters between fields. When the data mart structure is designed, the transformation process begins. • The loading technique is iterative. It guarantees that data is loaded into the data mart, changes data from a transactional structure to one that is appropriate for DSS analysis, and cleans data as needed, all according to the data warehouse manager's requirements. To reflect changes in the operating system, the data warehouse must be updated on a regular and progressive basis.

III.TOOLS FOR DATA WAREHOUSES THAT PERFORM ETL (EXTRACT, TRANSFORM, AND LOAD)

The design and management of the ETL process, which is one of the most complicated and resource-intensive components of a data warehouse project, is one of the most difficult and time-consuming tasks [4]. The management of this process is handled by ETL tools in many data warehousing initiatives. For example, Oracle Warehouse Builder (OWB) provides ETL capabilities while simultaneously making use of database features in a single package. Others develop their own ETL tools and methods, which they can use either within or outside of the database.
Aside from extraction, transformation, and loading, there are a few other actions that must be completed in order for an ETL deployment to be successful as part of the data warehouse's everyday operations and future updates. [5] The majority of ETL systems, such as OWB, take care of these tasks in addition to assisting with the creation of a data warehouse and data flow.
You must constantly add data to your data warehouse in order for it to perform its business analysis-assistance function properly. In order to accomplish this, data from one or more operational systems must be retrieved and copied into the data warehouse. It is the responsibility of data warehouse administrators in data warehouse environments to mix, restructure, and condense massive volumes of data from disparate systems into a new unified knowledge base for business intelligence.
Extracting data from source systems and transferring it to a data warehouse is known as ETL, which is an abbreviation for the processes of extraction, transformation, and loading. Please keep in mind that ETL refers to a general procedure rather than three specific procedures. The abbreviation ETL is likely oversimplified due to leaving out the moving step and and shows that the others procedures are differential from previous phase(s). It doesn't matter what you call it, the complete technique is referred to as ETL [6].

It has been well established for many years that ETL is a technique and that its functions are not confined to data warehouse environments: any enterprise's information technology backbone is comprised of a diverse range of proprietary applications and database systems. For applications or systems to integrate and offer at least two apps with the same perspective of the world, data must be shared between them across all of their components. Much of this data sharing was managed through methods that are now commonly known as ETL (enterprise transactional library).
ETL expenses are always the easier side of the equation to estimate because they are based on tools and time [7]. Saving money on ETL requires choosing a solution that is simple to use, connects well with your existing data architecture, runs rapidly, and can expand when you add more data or change your models, among other characteristics as mentioned in Fig.1.

Figure 1: ETL Process

IV.SIX COST COMPONENTS TO TAKE INTO ACCOUNT DURING ETL ESTIMATION

Software. ETL software is available in two flavors: standalone and integrated, and each has its own set of features. Integration of data transformation and loading (ETL) software is often included as part of an end-to-end business intelligence platform, making it the obvious choice if you are already in the market for a business intelligence solution. The use of standalone options is ideal if you already have a business intelligence solution in place or if you need to ETL your data for objectives other than reporting, such as relocating data storage locations or switching database types as mentioned in Fig.2.

Hardware. Unless you already have them, you will require workstations for the initial ETL programming and servers (either on premises or in the cloud) to house your source data and target data repositories, if you don't already have them. The ETL processing itself will also necessitate the use of servers, which will be performed more efficiently if your ETL system has change data capture capabilities. As an alternative to batch-updating the entirety of your repository every night, you can selectively update only stale sections of your target database, lowering the amount of resources required while also enhancing data quality. Because updates must frequently be completed in a short period of time, change efficiency means that you will not have to spend for more powerful processors.

Consulting. Enterprises frequently find it important to have data architects on their payroll in order to assist with post-implementation ETL operations, but some organizations do not have the resources to do so and instead choose to employ a consultant or contractor. A discovery call should be able to assist you in estimating the costs in this situation; consider adding an additional 20% to the price to account for optimism bias and hidden fees.

Figure 2: ETL Components

Training. In the event that you decide to construct and maintain your ETL in-house rather than contracting with an agency, you will be responsible for training your personnel on the ETL software that you choose. The more easily your product can be used, the less training your engineers will need end-user requirements within the application. Function point analysis is used to quantify and measure a variety of elements, including:

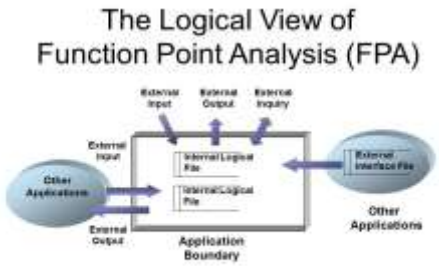

Figure 3: Logical view of FPA

to get them up to speed on it. It is important to note that many ETL systems are intended for multidimensional databases (MDBs) or "cubes," which are nonstandard and often require ETL architects to master proprietary querying languages in order to function properly. It is almost certain that adopting an ETL solution that links to relational databases would result in lower training expenses.

Development. Despite the fact that internal workers will be paid regardless of their tasks, you'll want to include in the opportunity cost of diverting them away from other projects. In order to account for the expected number of hours each team member is likely to spend on ETL-related duties, it's a good idea to include salaries in your ROI calculations when calculating your return on investment.

Upkeep and repair (total cost of ownership). If any of the expenditures listed above reoccur over the life of your ETL processes, include them in the ongoing maintenance costs to calculate the total cost of ownership for your ETL (TCO) [8]. There are some applications that may be obtained for a one-time licensing charge, while others that must be licensed on a periodic basis. It is possible that some firms will hire ETL consultants only during the initial planning stages, but others may seek to establish a long-term relationship with an agency. Training may be expensive at first, but it becomes reasonably inexpensive if internal staff members have gained the necessary skills to train new recruits themselves.

V. How FPA Estimates Cost Of ETL

All this activities of the ETL estimation process can be taken into account by using the FPA analysis tools. FPA is very simple to use and once we get the FPA estimates we can again get the ETL estimates.

Software for function point analysis automates the process of assessing a codebase, making it easier to complete as mentioned in Fig.3. The arduous process of finding, categorizing, and measuring end-user requirements is made simple by this tool as well. An application's function points are a unit of measurement that are used to quantify and measure

Productivity of the team or the individual who has received the functionality, performance of Vendors.

Function point analysis software gives the information that an organization requires in order to discover opportunities for improvement in the software development process. By implementing the findings of this evaluation, your firm will be able to decrease maintenance costs, manage risks, and obtain greater value out of the development team overall.

Due to the fact that the ETL cost is to be estimated, when we use the FPA tools it becomes simple. The cost of hardware and software can be easily estimated using FPA. How extraction will be done can simply be estimated using FPA as when we measure the end user requirements then we are able to know the cost of loading and extraction. The transformation of data is still included because by using FPA

estimation we are able to know how data will be fed to the system, how it will be transformed and later the user can be able to extract the information.

VI.CONCLUSION

Measuring the cost of ETL is very difficult. The best and suitable way to me up with the estimations is carrying out the FPA cost analysis and coming up with the ETL estimations. In the future this is how the ETL estimation will be carried out because this is the only way we can get the estimations. Many software development firms will venture into such kinds of estimation during the development process.

VII.REFERENCES

[1] J. wiley, "Rigorous and practical approaches in software development," *software metrics*, vol. 5, no. PWS Boston, pp. 31-59, 2018.

[2] Inmon, "function point counting practices," *internal function of point users*, vol. 3, no. Ohio IFPUG, pp. 4-6, 2013.

[3] Inmon, "Definition of datawarehouse," *warehousing tools*, vol. 3, no. PWS Boston, pp. 3-4, 2017.

[4] kimball, "Data warehouse toolkit," *data warehousing*, vol. 2, no. PWS Boston, pp. 8-11, 2009.

[5] Reeves, "Software engineering product quality," *software development life cycle*, vol. 1, no. Ohio Publishers, pp. 6-8, 2013

[6] Toffano, "Estimation of datamart," *Adapting function point analysis*, vol. 3, no. PWS Boston, pp. 6-8, 2011.

[7] Kramer, "software cost analysis," Analysis of softwares, vol. 2, no. boston, pp. 5-8, 2011.

[8] Reeves, "ETL estimation," Software metrics, vol. 2, no. PWS Boston, pp. 23-34, 2014.