Function Points Based Estimation Model for Data Warehouses
Function Points Based Estimation Model for Data Warehouses

Table of Contents
1. Introduction .................................................................................................................................................. 03
2. Estimation Challenges ............................................................................................................................... 03
3. Boundary, Scope and Estimation Approach ............................................................................................... 03

Author Bio
Karthikeyan Sankaran (Karthik) is currently working as a Senior Consultant in the Business Intelligence practice at Hexaware Technologies, a global provider of Information Technology Solutions based in India. Karthik has over 10 years of experience in Business Intelligence domain, having worked as an architect, consultant and project manager for data warehousing projects. Karthik can be reached at karthikeyans@hexaware.com.

Abstract
Data warehousing applications differ from the traditional application development in the way that they are predominantly subject oriented rather than process oriented and have an analytic centric focus rather than a transaction centric focus. To evolve a structured estimation method for such an application is especially challenging as the data is pooled from a plethora of application sources and they are implemented using a myriad of solution domains. The further transformation, source qualification and data cleansing that happens on this data and transgression into a format which supports business analytics adds to further complexity of these estimates.

The user centric approach of this domain lends itself to adopt Function Points since it is a methodology which is based on logical user oriented terms. A structured estimation model is attempted to be evolved based on size which is measured in function points. The model is tweaked and mapped to the various components of the data warehousing domain whose data model is architected using a dimensional model rather than a relational model. Complexity factors other than size and their interrelations have been factored in framing a structured estimation model.
1. Introduction

Business intelligence (BI) is a business management term which refers to applications and technologies which are used to gather, provide access to and analyze data and information about company operations. Business intelligence systems can help companies have a more comprehensive knowledge of the factors affecting their business and thus help companies make better business decisions.

Data Warehousing can be considered as the technology domain that facilitates Business Intelligence in any organization. This technology realm of BI has 3 major components:

**Back-Room Architecture**
Technology components that are used to extract data from source transactional systems, integrate them, transform the data using business rules and load into target data repositories that aid decision making.

**Front-Room Architecture**
Technology components that help the business users analyze the information present using pre-built & ad-hoc reports and utilize the whole range of analytical solutions.

**Data Repository**
Typically called a Data Warehouse / Data Mart / Operational Data store, this layer models the data in a subject oriented, integrated, non-volatile, time-variant fashion that enables the back-room & front-room architectures to work seamlessly.

2. Estimation Challenges

- Business Intelligence systems consist of many tools & technologies, each with its own advantages and constraints for handling a business problem. The nuances of the tool have to be taken into account for effort estimation.
- Backend architecture for DW systems, typically called the ETL layer, has many interlinked processes that are run to gather the information requirements. The interlinking / sequencing of processes that dictate how the business rules are applied in a particular situation pose difficulties for estimation.
- The front-end Reporting / Analytical layer has many user-centric softer aspects to it like performance of reports, the clarity in the semantic layer for adhoc analysis, etc. All these factors need to be taken into account for arriving at proper estimates.
- As the DW / BI system evolves over time, new functionality is being added on a daily basis. For each of these requirements, the process and data should be in conformance to what is already present in the data warehouse. This implies that the effort for regression testing should also be factored into the effort.
- Data warehouses process huge volumes of data on a daily basis. Gauging the effort required to perform load testing for each new requirement is also a difficult task.

3. Boundary, Scope and Estimation Approach

Data warehousing applications typically have a staging area which pools the data from source applications, an Extraction Transformation Loading (ETL) process which is implemented through data integration tools, validations and transformations to enforce the business logic, process alerts to provide the status notification for the ETL process and a target table where the transformed data is loaded.

![Figure 1: DWH Application Boundary](image-url)
This article is restricted to analyzing the estimation model for back room architecture which involves the data integration (ETL) process. The following approach can be adopted for framing the estimation model:

**Step 1:** Gauging the size of the application using Function Points
**Step 2:** Assessing the ETL complexity factors
**Step 3:** Performing regression analysis
**Step 4:** Implementation of the effort estimation model

### Step 1 - Size Estimation Using Function Points

The Function Point model has five basic function types which are bifurcated into data functions and transaction functions. Data functions have two constituents which are internal logical files (ILF) and External Interface Files (EIF). Transaction functions have three types which are External Inputs (EI), External Outputs (EO) and External Inquiries (EQ). The application of Function point types to DWH application is not straightforward as opposed to Web Application or GUI applications. This was mapped to the DWH components as per the following guidelines.

<table>
<thead>
<tr>
<th>DWH Component</th>
<th>FP Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staging Tables</td>
<td>EIF</td>
</tr>
<tr>
<td>Target Tables</td>
<td>ILF</td>
</tr>
<tr>
<td>Fact Table</td>
<td>ILF</td>
</tr>
<tr>
<td>Dimension Table</td>
<td>ILF</td>
</tr>
<tr>
<td>Look Up Table</td>
<td>ILF</td>
</tr>
<tr>
<td>Mapping</td>
<td>EI</td>
</tr>
<tr>
<td>Process Alerts</td>
<td>EO</td>
</tr>
</tbody>
</table>

**Staging Tables**

Staging tables hold the data which is aggregated from different application sources. Even though the staging tables are resident within the boundary of the data warehouse, they are considered to be External Interface Files (EIF) as they primarily hold the application data of external systems and are maintained within the staging area for performance considerations. If there is any additional processing which is involved before updation of these tables then these files could be considered as Internal Logical Files (ILF).

**Target Tables**

These are destination tables which are considered to be Internal Logical Files (ILF) since they are updated with the transformed data after different transformation and exceptions processing in accordance with the business logic within the boundary of the data warehouse.

**Mapping**

The mapping component in a data warehouse primarily implements the Extraction Transformation and Loading logic by pulling the data from the source table which undergoes transformations before being loaded into a target table. These are considered an External Input (EI) even though the mapping wouldn’t necessarily involve an updation of the target tables as the system’s behavior is altered by way of the transformational expression which implements the business logic. Mapping in such instances will be considered as a control data which alters the system’s behavior.

**Look Up Table**

This table generally maps the Identifier field and the Description field. Since the description will invariably not be available as part of the source data and as the description is required by the business, the look up tables are referenced by the ETL mapping before loading the data into target tables. Even though a look up table resembles a code data which provides the explanatory description for an identifier it is not something which is serving any technical implementation. Hence the look up tables are considered as reference data and are included in the ILF count and counted in the FTR of the ETL mapping process.

**Process Alerts**

These alerts are considered as External Outputs (EO) since these notify the status of the ETL process after updation into the target table.
Step 2 - Assessing the ETL Complexity Factors
Since the 14 general system characteristics given by the IFPUG counting practices manual were inadequate for estimating the effort required given the complexity of a data warehousing application, a separate brainstorming session was conducted and a cause and effect analysis was done to identify the complexity factors which are depicted below.

![Figure2: Cause and Effect Diagram for ETL Complexity Factors](image)

Based on the Initial Analysis 27 complexity factors were identified and a correlation analysis was performed. By eliminating some of the insignificant factors and bringing in some more factors this was later curtailed to 19 complexity factors and a fresh set of data was re-collected for performing the regression analysis which is presented below.

Step 3 - Regression Analysis
Based on the above data points gathered a step wise forward regression analysis was performed to filter out the most influential factors which have a correlation with actual effort. The regression equation that was developed for this specific project looked like:

\[
\text{Actual Effort} = A + B\cdot(X5) + C\cdot(X9) + D\cdot(X15) - E\cdot(X16) - F\cdot(X19) + G\cdot(X2X4) + H\cdot(X3X4)
\]

Where
- \(X5\): Number of Target Columns
- \(X9\): Type of Transformations
- \(X15\): Parameter File to be modified or created
- \(X16\): New Mapping
- \(X19\): Unadjusted Function Points
- \(X2X4\): Amount of data processed * No of transformations (Interrelated Factors)
- \(X3X4\): Performance Criteria * No of transformations (Interrelated Factors)

The Correlation co-efficient R-Sq value 98% indicates a significant correlation of the above identified 7 factors (out of 19 factors) with the actual effort. This R-Sq value signifies 98% variation in output described by these 7 factors.

Step 4 - Implementation of the Effort Estimation Model
The Estimation model was implemented and there was significant improvement in the process capability of the estimation process.
Function Points Based Estimation Model for Data Warehouses

Figure 3: Process Capability before Implementation

![Process Capability before Implementation](image1)

Figure 4: Process Capability after Implementation

![Process Capability after Implementation](image2)
Function Points Based Estimation Model for Data Warehouses

Safe Harbor
Certain statements on this whitepaper concerning our future growth prospects are forward-looking statements, which involve a number of risks, and uncertainties that could cause actual results to differ materially from those in such forward-looking statements. The risks and uncertainties relating to these statements include, but are not limited to, risks and uncertainties regarding fluctuations in earnings, our ability to manage growth, intense competition in IT services including those factors which may affect our cost advantage, wage increases in India, our ability to attract and retain highly skilled professionals, time and cost overruns on fixed-price, fixed-time frame contracts, client concentration, restrictions on immigration, our ability to manage our international operations, reduced demand for technology in our key focus areas, disruptions in telecommunication networks, our ability to successfully complete and integrate potential acquisitions, liability for damages on our service contracts, the success of the companies in which Hexaware has made strategic investments, withdrawal of governmental fiscal incentives, political instability, legal restrictions on raising capital or acquiring companies outside India, and unauthorized use of our intellectual property and general economic conditions affecting our industry.