International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-11 Issue-5, January 2023

Arriving at the Results by Comparing with the Traditional Approach to My Approach in Deriving Function Points for ETL Operations

A. Rakesh Phanindra, V. B. Narasimha

Abstract: ***It can be hard to guess how much data will need to be put into the data warehouse when the whole history of the transaction system is moved there. This is especially true when the transfer process could take weeks or even months. The ETL system's components must be broken down into their three independent stages; nevertheless, when estimating a large starting load. Data extraction from source systems, creating the dimensional model from data, loading the data warehouse and timing estimates for the extraction process. Surprisingly, data extraction from the source system often accounts for the majority of the ETL procedure. Online transaction processing (OLTP) systems are not designed to return massive data sets from the data warehouse's historical load, which extracts a tremendous quantity of data in a single query. However, the daily incremental loads and the breath-of-life historic database loads are very different. In any case, fact-table filling requires data to be pulled in a different way than what transaction systems can do. ETL extraction procedures often require time-consuming techniques such as views, cursors, stored procedures, and correlated subqueries. It is essential to anticipate how long an extract will take to begin before it does. Calculating the extract time estimate is challenging. Due to the hardware mismatch between the test and production servers, estimates based on the execution of the ETL operations in the test environment may be significantly distorted. Sometimes, working on specific projects where an extract task would run continuously until it eventually failed, at which point it would be restarted and run once more until it failed. Without producing anything, days or even weeks passed. One must divide the extraction process into two simpler steps to overcome the challenges of working with large amounts of data. Response time for queries. the interval between when the query is conducted and when the data starts to be returned. It is pertinent that the effort required for ETL Operations for Data Marts and DWH projects, in terms of function points, is essential. In the previous paper, I discussed general System Characteristics to determine the Value Adjustment Factor. In this paper, I present the results. I compared my findings with the conventional FPA on industrial projects to evaluate the Function Point Analysis's suitability for Data Mart projects. I outline the strategy, implementation, and outcomes analysis of this validation in this section.***

Keywords*: *Function Point Analysis, ETL, Data Marts.

I. INTRODUCTION

The scientific calculation of efforts for ETL Operations in DWH Projects has become paramount in the current IT environment, where the emergence of Big Data and Data Lakes has also led to the increased importance of effort calculation. Function Points is one of the best ways for arriving at the effort estimates for ETL Operations [1]. An Heeringen, n.d. Two crucial things have to be derived from Function Point Analysis. One is Unadjusted Function Points, and the second one is Adjusted Function Points [2]. In the last paper I published, I derived General System Characteristics that are needed for Adjusted Function Points. In this Paper, I present the results of comparing conventional FPA with my findings on several industrial projects. Laird and Brennan (2006). These results can also be applied to a Data Lake or Big Data scenario, yielding good results.

The General System Characteristics (GSC) are the most critical factors to arrive at the adjustment factor needed for ETL operations estimations in function points. The following 14 general system characteristics are identified as part of the research [3].

Calculate the value adjustment factor (VAF) using the 14 system attributes (GSC).

Features of the system generally

A. Data communications, GSC 1

What number of communication tools are available to facilitate the exchange of data with the application or system?

B. Processing of distributed data in GSC 2

The management of distributed data and processing functions.

C. Performance of GSC 3

Was the user requesting a certain throughput or response time?

D. Heavily used setup for GSC 4

How busy is the hardware environment where the programme will run right now?

E. Rate of GSC 5 Transactions

How often are transactions carried out on a daily, weekly, monthly, etc. basis?

F. Data entry using GSC 6 online

What proportion of the data is entered online?

G. GSC 7 User effectiveness

Was the application efficiently created for users?

Manuscript received on 14 November 2022 | Revised Manuscript received on 25 December 2022 | Manuscript Accepted on 15 January 2023 | Manuscript published on 30 January 2023.

*Correspondence Author(s)

A. Rakesh Phanindra*, Information Technology, Institute of Public Enterprise, Survey No. 1266, Shamirpet (V&M), Medchal, Malkajgiri district, Hyderabad (Telangana), India. E-mail: rakeshmtech2011@gmail.com, ORCID ID: https://orcid.org/0000-0003-1712-3894

Dr. V. B. Narasimha, Assistant Professor, Department of Computer Science and Engineering, University College of Engineering, Osmania University, Hyderabad (Telangana), India. E-mail: vbnarasimha@gmail.com

H. Update for GSC 8 online

What percentage of ILFs are updated through online transactions?

I. Processing complex GSC 9

Does the application use a lot of logic or math processing?

J. Usefulness of GSC 10

Was the programme created to serve the demands of a single user or a group of users?

K. Easy GSC 11 Installation

How difficult is installation and conversion?

L. GSC 12 Operational comfort

How well-automated and/or efficient are the start-up, backup, and recovery processes?

M. GSC 13 Multiple locations

Was the programme created, developed, and supported expressly to be installed at different locations for different organizations?

N. GSC 14: Facilitate change:

Did the application have a focus on facilitating change during its development, design, and support?

II. PREPARING FOR THE VALIDATION

On actual Data Mart projects from various Organization, I chose to compare the outcomes of my strategy with those of the conventional FPA to validate it. I required details about how each project was built, including the time it took to develop the Data Mart and the number of FTEs (Full-Time Equivalent) assigned to the project. Finding all of this data on other projects was difficult; I could only find 5 that were suitable for my purposes. Two public sector banks and one government organization each developed one of these projects. Among the projects I looked at were two from Institution J1, one from Institution J2, and two from Institution J3. I conducted structured interviews with the project managers to gather data for my experiment and examined the available system documentation. I gave the project manager the data and measurements I had obtained for each project and verified them. Previous data were measured in person-days or person-months of labour. I used a person-month of working days equivalent to 21, with each Person day's effort being 8 hours, to normalise these data with those from the FPA. Given that the three organizations are giant banks or governmental, they tend to work more consistently, and I think my Estimation is reasonably close to reality. Lastly, to determine the execution effort and duration, one has to multiply the productivity factor of a project, typically unique to the Organisation, with the adjusted function points that are derived. The productivity factor is reliant on the programming language used and can be based on the company's previous data. For the used programming language, C++, Organization J1 already contains historical data. Throughout the Data Mart initiatives. 16.00 hours are spent on each function point in terms of productivity. Organizations J2 and J3 don't have any historical data and haven't used metrics in the past. I took into account the typical productivity factor for J2 for Programming. 8 hours were spent on each function point in the employed language (Visual Basic). Given that J3 uses OLAP ETL Processing tools, I utilized productivity factor, which is been 6.8 hours for each FP.

III. SCALING UP DATA MARTS

For the five Data Mart projects selected, I employed both my proposed strategy and the conventional FPA. I conducted interviews with the members of each project's development team to learn more about the General System Characteristics (from both techniques), the amount of time it took to build them, how many people worked on the programme, and the duration [3]. The outcomes of both approaches for the mentioned 5 Projects are summarized in Table 1.

A. One could note that:

- Since input, output, and queries are the same for all transaction functions (External Input, External Output, and External Queries), in both methods and both strategies, they were counted equally. Why? Because my approach uses more features to compute the total degree of influence (TDI) than the conventional one, the TDI is optimal with my approach about Data Mart [5].
- The external output and external query for all Data Marts are null, since the end user directly creates their queries using OLAP tools at the three Organizations.
- Because I used a scientific methodology, my approach has a higher adjusted function point (AFP) than the conventional approach for all Data Marts.

Formally, it may be said that my estimation is more accurate than the conventional FPA estimation when it comes to real-time data.

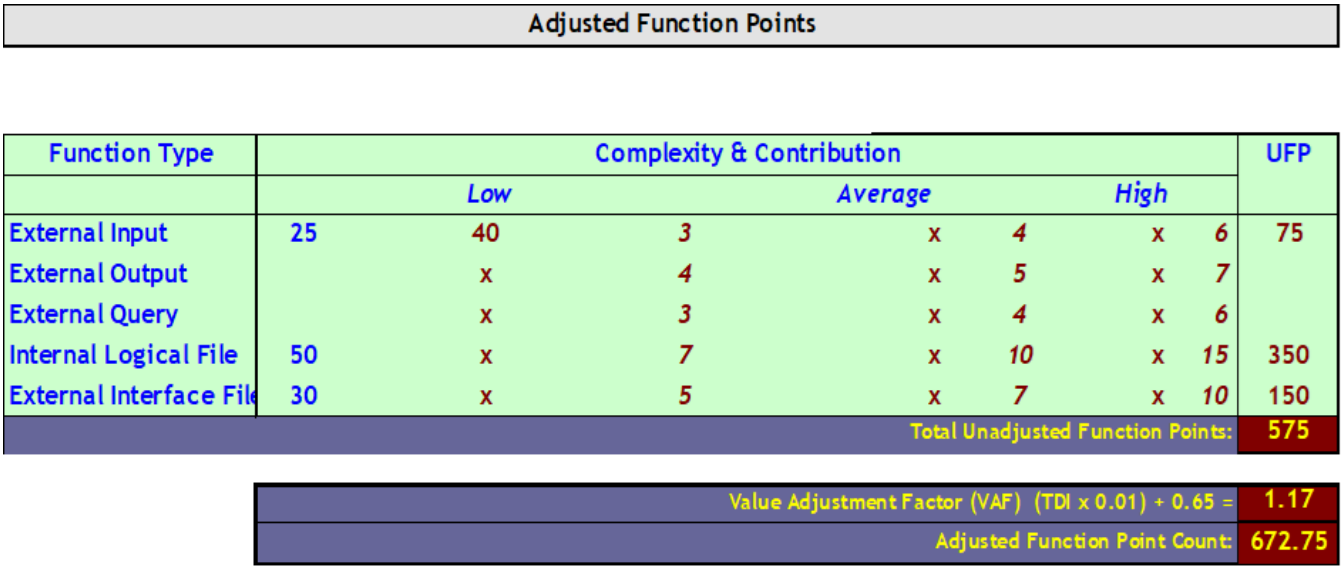
Table- I: Traditional Approach

	ILF	EIF	EI	EO	EQ	UFP	VAF	AFP
J1DM1	128	6	55	0	0	1091	1.17	1276.5
J1DM2	128	25	60	0	0	1201	1.17	1405.2
J2DM1	75	2	40	0	0	700	1.17	819
J3DM1	50	30	25	0	0	575	1.17	672.75
J3DM2	100	65	40	100	0	1545	1.17	1807.7

Table- II: My Approach

	ILF	EIF	EI	EO	EQ	UFP	VAF	AFP
J1DM1	256	6	55	0	0	1987	1.11	2205.6
J1DM2	256	25	60	0	0	2097	1.11	2327.7
J2DM1	150	2	40	0	0	1194	1.11	1325.3
J3DM1	100	30	25	0	0	925	1.11	1026.82
J3DM2	200	65	40	100	0	2245	1.11	2192

Legend:
EI – External Input EO – External Output
EQ – External Inquiries
EIF – External Interface File
ILF – Internal Logical File
AFP – Adjusted Function Point count
VAF – Value Adjusted Factor
UFP – Unadjusted Function Point count



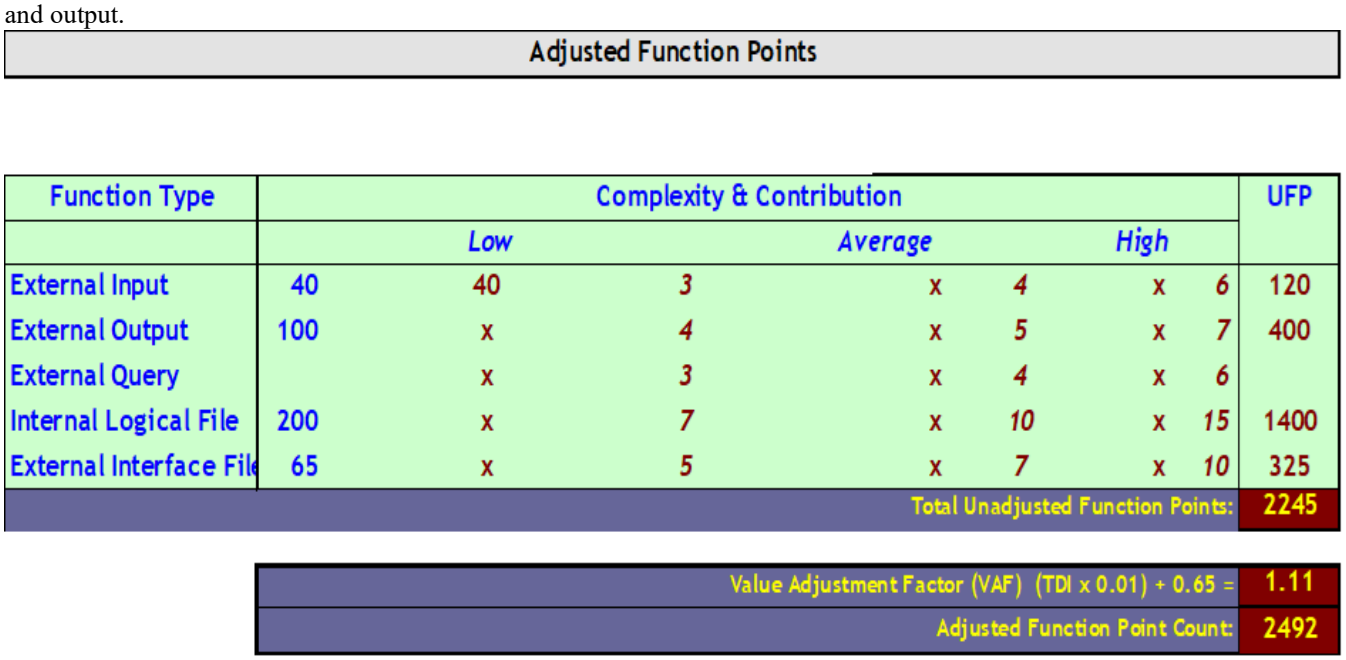
Adjusted Function Points

Function Type	Complexity & Contribution							UFP
		Low			Average		High	
External Input	25	40	3	x	4	x	6	75
External Output		x	4	x	5	x	7	
External Query		x	3	x	4	x	6	
Internal Logical File	50	x	7	x	10	x	15	350
External Interface File	30	x	5	x	7	x	10	150
							Total Unadjusted Function Points:	575

Value Adjustment Factor (VAF) (TDI x 0.01) + 0.65 =	1.17
Adjusted Function Point Count:	672.75

Fig 1: A Result snapshot of the Traditional Approach

External Input (EI) is a transactional functional type that handles data or control information that enters the application from beyond its boundaries. The EI is a simple technique. External Output (EO) is a basic operation that creates data or control information transmitted outside the confines of the application[6]. External Inquiries (EQ): An essential process that produces data retrieval is an EQ, which combines input and output. A user-identifiable collection of logically connected data or control information kept within the application's confines is referred to as an internal logical file, or ILF [7]. External Interface File (EIF): A collection of users' recognised, logically connected data that alludes to a piece of software while being kept inside the confines of another piece of software [8].



Adjusted Function Points

Function Type	Complexity & Contribution							UFP
		Low			Average		High	
External Input	40	40	3	x	4	x	6	120
External Output	100	x	4	x	5	x	7	400
External Query		x	3	x	4	x	6	
Internal Logical File	200	x	7	x	10	x	15	1400
External Interface File	65	x	5	x	7	x	10	325
							Total Unadjusted Function Points:	2245

Value Adjustment Factor (VAF) (TDI x 0.01) + 0.65 =	1.11
Adjusted Function Point Count:	2492

Figure 2: A Result Snapshot of My Approach

In the project I have undertaken, there are 40 low External Inputs, 100 low External Outputs, 200 low internal logical files that are either read, updated, etc., and 65 low External Interface files where the data has been extracted. I applied general system characteristics that I had derived, which enabled me to determine an adjustment factor. After multiplying the unadjusted function by the adjustment factor, I was able to derive the final function points required.

IV. RESULT

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Arriving at the Results by Comparing with Traditional Approach to My Approach in Deriving Function Points for ETL Operations

Internal Logical File (ILF)								
ILF Name	RET	DET	Complexity					Remark
			Low	Average	High	Very High	V.V High	
customer	0	0	200					
TOTAL NO OF ILF :: 200								

External Interface File (EIF)								
EIF Name	RET	DET	Complexity					Remark
			Low	Average	High	Very High	V.V High	
Account	0	0	65					
TOTAL NO OF EIF :: 1								

External Inputs (EI)								
EI Name	FTR	DET	Complexity					Remark
			Low	Average	High	Very High	V.V High	
TOTAL NO OF EI :: 40								

External Outputs (EO)								
EO Name	FTR	DET	Complexity					Remark
			Low	Average	High	Very High	V.V High	
TOTAL NO OF EO :: 100								

External Inquiry (EQ)								
EQ Name	FTR	DET	Complexity					Remark
			Low	Average	High	Very High	V.V High	
TOTAL NO OF EQ :: 0								

Total Unadjusted Function Points: 2245 Value Adjustment Factor (VAF) (TDI x 0.01) + 0.65 = 1.11 Adjusted Function Point Count: 2492

V. METHODOLOGY

Comparing the data obtained from the five projects using the traditional approach and my approach (which utilises newly defined general system characteristics) is my goal in terms of validation. My technique for determining FPs for the ETL Operations is quite close to the actual figures that the Project Manager and Developers came up with after the Projects were completed. The conventional method was not producing the proper FPs. There is a lot of variation, and my results are relatively accurate, both from the perspective of depicting ILFs, EIFs, EI, EO, and EQs as well as from the standpoint of General System Characteristics. The General System Characteristics are more specific to ETL Operations rather than the Generic Ones, which are there for Traditional Function Points [4]. According to Caper Jones and QSM, the productivity factors might change. However, if thoroughly analysed from a given set of diversified organisations, as I have done in this case, it yields accurate results. The number of person-days, man-hours, and eventually the duration could be calculated if we know the FP count. I thought it best not to calculate the person-days, person-hours, and duration, as every Project Manager and Program Manager will have different opinions on implementation, such as using the Waterfall Model or Agile Model.

V. CONCLUSION

According to the Analysis and comparison, my method has produced better outcomes in terms of Function Points for ETL Operations. The methodology I have employed would yield better results in terms of actuals because innovation is occurring so quickly and there is a need to evaluate massive data structures where Organizations are moving toward data lakes.

DECLARATION

Funding/ Grants/ Financial Support	No, I did not receive.
Conflicts of Interest/ Competing Interests	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval or consent to participate, as it presents evidence that is already publicly available.
Availability of Data and Material/ Data Access Statement	If not applicable: Not relevant.
Authors Contributions	All authors have equal participation in this article.

REFERENCES

1. Verner, June M. and Tate, Graham, "A Model for Software Sizing", Journal of Systems and Software, IEEE Software, pp. 173-177, July 1987. [CrossRef]
2. Albrecht, Allan J. and Gaffney (Jr), John E., "Software Function Source Lines of Code and Development Effort Reduction: A Software Science Validation", IEEE Transactions on Software Engineering, Vol. SE-9, No. 6, pp. 639-647, Nov. 1983. [CrossRef]
3. L. M. Laird and M. C. Brennan, 2006. Software Measurement and Estimation: A Practical Approach, Wiley-IEEE Computer Society Press, ISBN: 0-471-67622-5. [CrossRef]
4. IFPUG. IFPUG, International Function Point Users Group. Function Point Counting Practices Manual: Release 4.1. IFPUG, Ohio, release 4.1 edition, 2000.
5. C. R. Symons, Software Sizing and Estimating– MkII FPA (Function Point Analysis), John Wiley and Sons, Chichester, U.K., 1991
6. A. Abran, M. Maya, J. M. Desharnais, and D. St-Pierre, "Adapting function points to real-time software," American Programmer, Vol. 10, 1997, pp. 32-43.
7. Calazans, K. de Oliveira, and R. Santos. Adapting function point analysis to estimate data

mart size. In Software Metrics, 2004. Proceedings. 10th International Symposium on, pages 300--311, 10th IEEE International Symposium on Software Metrics (METRICS'04), sept. 2004. IEEE International Symposium.

8. L. Santillo. Size & estimation of data warehouse systems. The European Software Measurement Conference - FESMA/DASMA, 1(1): :0, 2001. Germany.

AUTHORS PROFILE

A. Rakesh Phanindra is an M.Tech in Computer Science and Engineering. Assistant Professor of Information Technology at the Institute of Public Enterprise, Shamirpet Campus, Hyderabad. He is a Life member of ISTE and CSI. He is currently working towards his PhD Degree in the Department of Computer Science and Engineering at the University College of Engineering, Osmania University. His Teaching and Research interest areas are Software Engineering, Cloud Computing, Network Security and Cyber Security.

Dr. V. B. Narsimha Asst. Professor, Dept. Of Computer Science and Engineering, University College of Engineering, Osmania University, Hyderabad and Joint Director of PGRRCDE, Osmania University. His more than 20 Years of Teaching and research experience span the areas of computer networks, Data Mining, Cloud Computing, and Analytics.