Benchmarking, Estimation & Productivity Assessment Framework for IT Implementation

* *Chandan Kumar Jha*

Abstract

The dependence and fusion of Information Technology in modern organizations has led to the development of extremely mature service delivery models in recent years. Business units are augmenting their IT budget/spending to encapsulate & leverage the latest technology innovations thus empowering their product/service portfolio. Subsequently, technology sector is witnessing wide variance in service offerrings of IT firms offering a bouquet of solutions to meet business requirements. This beckons the need of a robust 'benchmarking and estimation' framework for analyzing the efficacy of vendor solution and accomplish savings in the procurement phase leading to enhanced operational efficiency. Further, a proficient logical model is also needed to measure the productivity of the IT vendor vis-à-vis Industry standards at the end of implementation. In this paper, we would discuss the intricacies of the procurement phase and devise a reformed benchmarking, estimation & productivity assessment framework for IT Implementations.

Index Terms **: Benchmarking, Procurement, Estimation, Productivity, Information Technology**

I. INTRODUCTION

Recent industry trends suggest that Information Technology consumes nearly 10% of the overall budget cost [1] in most of the larger establishments. These huge investments are accompanied by an increased demand for operational efficiency and cost savings. Cost and effort benchmarks coupled with productivity assessment cannot only assist organizations achieve this objective but also help build a database for recording vendor performance and derive several useful metrics for further analysis.

In this paper, we would make an attempt to formulate a comprehensive framework that organizations can readily put in place to derive cost and effort benchmarks for respective projects during the procurement stage and compare the same with vendor estimates. Subsequently, clients can challenge/negotiate with their vendors resulting in direct cost savings. At the end of implementation stage, clients can measure vendor productivity i.e. compare the performance of the vendor against the established benchmarks leading to operational efficiency. Fig. 1 depicts the benchmarking and productivity assessment framework; each of these

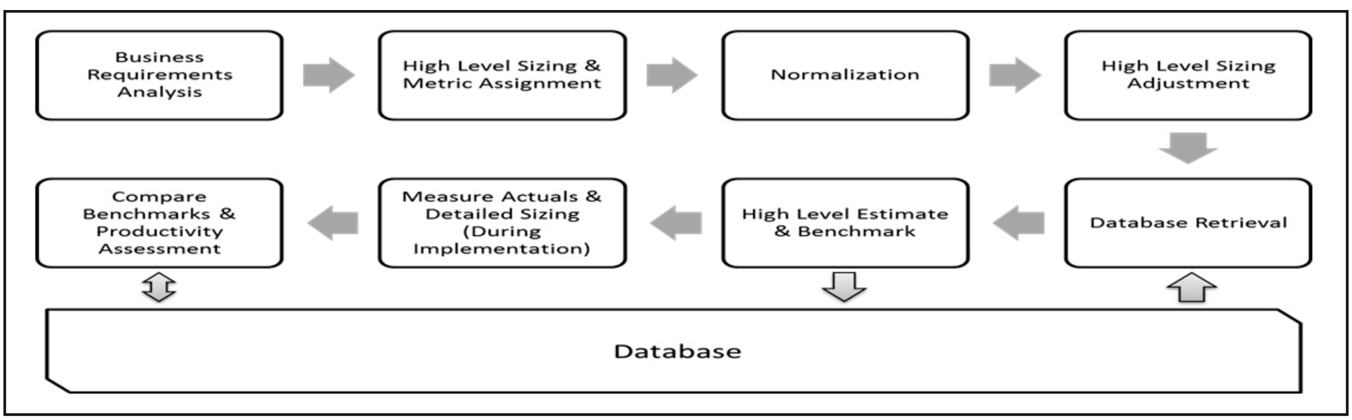

Fig. 1. Benchmarking, Estimation, & Productivity Assessment Framework

* *Indian Institute of Management,* Indore, Senior Management Consultant, Ernst & Young
E-mail : chandanpce@gmail.com

sub activities will be detailed in the subsequent sections.

TABLE I.
DUMMY PROJECT ABC

Req No.	Description	Size (Function Point)
1	Customer report at line item level at required.	50
2	Aggregate sum of pricing quantity need to be displayed along with date	100
3	Ability to collect sample from suppliers	150
4	Various Search option should be available for the buyer	40
5	Record validity periods for the contracts	60
	Total Size	400

II. BUSINESS REQUIREMENT ANALYSIS

Gathering business requirements is an important sub activity of the Planning Phase. According to PMI standards [2] it is expected that project teams will outline the 'As is' and 'To Be' business process in the respective requirement specification or functional design document. The planning teams can utilize the recommended process mapping techniques viz SIPOC, Process Diagrams, Value Stream Mapping, Flowcharting etc. for documenting and designing the business requirements [3]. Depending upon the nature of application prevalent and environment constraints, there are a variety of software tools to capture the business requirement. These requirements need to be agreed upon and signed off by the concerned business owner as these would be the base for future negotiation and deliberations. The signed off document will serve as a single source of truth for resolving scope conflicts between the vendor and the client.

III. HIGH LEVEL SIZING & METRIC ASSIGNMENT

Post finalization and documentation of requirements business units would need to size the same in terms of an appropriate unit of measure. Depending upon the nature of software environment prevalent in the organization, a variety of techniques prescribed for sizing IT application viz. Function Point Analysis, KLOC, Story Point Analysis, COCOMO etc. can be chosen [4]. Each of these techniques derives a common unit of measure by gauging the business/ technical/system parameters. The output of this exercise would be the size of the business requirement in a sizeable unit of measure/metric. It may be noted that the analysis is performed at a very high level when there is limited clarity on the actual requirements and system design; hence there is a scope of variance i.e. upliftment or depreciation of count in the subsequent stages of software design. The count is therefore termed as 'high level count'. Let us take an example to understand this better. Table I comprises dummy requirements for Project ABC and each requirement has been sized using a Sizing technique - Function Point Analysis in this case [5].

IV. NORMALIZATION

It has been observed that almost all software sizing techniques have inherent limitations due to the differences in the technical and implemenatation environment. It is very difficult to choose a sizing technique that would suffice in multiple scenarios; hence, depending upon the organizational and environmental factors we need to adjust the high level count to incorporate the system/technical characteristics [6]. In the above example, let us assume that the associated normalization factors suggest that the count needs to be adjusted by 5%.

V. HIGH LEVEL SIZING ADJUSTMENT

Based upon the inputs received during the normalization process, high level count needs to be adjusted accordingly. Thus, the high level size derived in Table I needs to be incremented by 5% to arrive at 'high level normalized count' i.e. 400+20=420 Function point.

VI. DATABASE RETRIEVAL

Benchmarks are derived in accordance with the data of the historic projects. It is prescribed that organizations maintain the record of the projects implemented in the past, it would help serve as an ammunition and data warehouse for analyzing estimates for future projects. After the nature of the project in consideration has been established, we need to extract the details of the project having similar characteristics from the database. A typical database should have the below minimum specifications (refer table II).

TABLE II.

Project Name	Characteristics 1	Characteristics 2	Characteristics 3	Characteristic ...	(A) Total Cost	(B) Total Effort	(C) Total Size(in metric denominator)	(A/C) Cost/Size	(B/C) Effort/Size

VII. HIGH LEVEL ESTIMATE & BENCHMARK

Once the fitting (similar nature) projects are identified from the database, we need to multiply the size (from table I) with average figures derived from database of historic projects (table II).

Total Cost Estimate= (Cost/Size) X Size of the current project

Total Effort Estimate= (Effort/Size) X Size of the current project

*(Cost/Size) is derived from table II i.e database for similar projects

*Size of the current project is derived from table I

The resultant is the total cost and effort benchmark for the project. Coming back to the example stated earlier, let us assume that the cost/size from (table II) comes out to be $ 150/Function point; hence the total cost of the project shall be $ 150*420=$ 63300 .

VIII. MEASURE ACTUALS & DETAILED SIZING (DURING IMPLEMENTATION)

It is an established fact that project requirements are clearer after the Design stage [7]. This can lead to deviation between the planned cost and actual cost. Hence, it is prescribed that that project teams should size the project against actual requirements (including any change requests etc.) and alter the high level project estimate if required after the implementation stage. This detailed sizing should be considered for measuring the final productivity of the project. In our example we observed that the project had a high level estimate of 420 Function points equivalent to $ 63300 .Let us assume that the detailed estimate after the implementation (including the change requests) signified the project size as 450 Functions Points i.e. 450*150=$ 67500.The updated actual cost should be analyzed for deriving the productivity of the project.

IX. COMPARE BENCHMARKS & PRODUCTIVITY ASSIGNMENT

Finally, the actual cost incurred needs to be compared with the prescribed estimate from database and the deviation will determine the productivity. This productivity factor will highlight the savings/overshoot in terms of benchmarks vis-à-vis actual figures (refer table III). If compared against the high level estimates, it also shows the efficiency/accuracy of the project team during the planning phase.

TABLE III.

Project Name	High Level Estimate before Implementation (from section 7)	Revised Estimate during implementation (from section 8)	Actuals Incurred in project (from Section 9)	Difference & Productivity Assessment (from Section 10)

X. CONCLUSION

Once we have sufficient number of data points, organizations can derive multiple indirect metrics for deriving productivity. A productivity assessment matrix is proposed in fig. 2, the example taken in the previous sections has also been plotted in the figure and it falls in the third quadrant.

1. Quadrant 1-Value for Money viz. low project implementation cost & low effort per function point (metric) value delivered by project

2. Quadrant 2- Expensive performers viz. high project cost but good effort efficiency & resource utilization

3. Quadrant 3-Economical laggards viz. low cost but poor resource utilization & effort efficiency

4. Quadrant 4-Twin Leakage viz. high effort utilization and huge cost; inefficient portfolio.

TABLE IV.

	Project Size (FP count)	Total Cost (in 000)	Total effort (mandays)	Effort/ FP	Cost/ FP
Project 1	1040	364	603	0.58	350
Project 2	1859	660	1171	0.63	355
Project 3	1356	434	922	0.68	320
Project 4	1515	500	1197	0.79	330
Project 5	2003	801	1742	0.87	400
Project 6	1397	545	1677	1.20	390
Project 7	1759	563	2287	1.30	320
Project 8	1714	300	1029	0.60	175
Project 9	2609	600	1826	0.70	230
Project 10	4000	800	3200	0.80	200
Project 11	2713	434	2441	0.90	160
Project 12	4275	855	4361	1.02	200
Project 13	625	150	750	1.20	240
Project 14	2989	538	3886	1.30	180
Example used	450	67.5	495	1.10	150

Fig. 2. Description:

- X axis denotes the EFFORT (in mandays) per unit of measure(Function point in this example)
- Y Axis denotes the cost (in dollars) per unit of measure (Function point in this example)
- Size of the bubble denotes the total cost of the project (in '000 dollars)
- The diagram is a 'productivity analysis' specimen using dummy data for 15 projects (mentioned in Table IV).
- From the example mentioned in the previous sections (bubble marked with the number 67.5 in Fig. 2), the specimen project lies in 3rd Quadrant i.e. **economical laggards** viz. low cost but poor resource utilization & effort efficiency
- It should be the goal of the organizations to bring all projects in 1st Quadrant i.e. **value for money** viz. low project implementation cost & low effort per function point(metric) value delivered by project

REFERENCES

[1] The Deloitte CIO Survey 2014 CIOs: At the Tech-Junction. Available: http://www2.deloitte.com/content/dam/Deloitte/fi/Documents/technology/CIO%20Survey%202014.pdf, p. 5, 2014.

[2] P. Burek. "Creating Clear Project Requirements – Differentiating "What" from "How"". Originally published as part of 2008 PMI Global Congress Proceedings – Denver, Colorado, USA, 2008.

[3] OSSS Process Management Manual. Available:

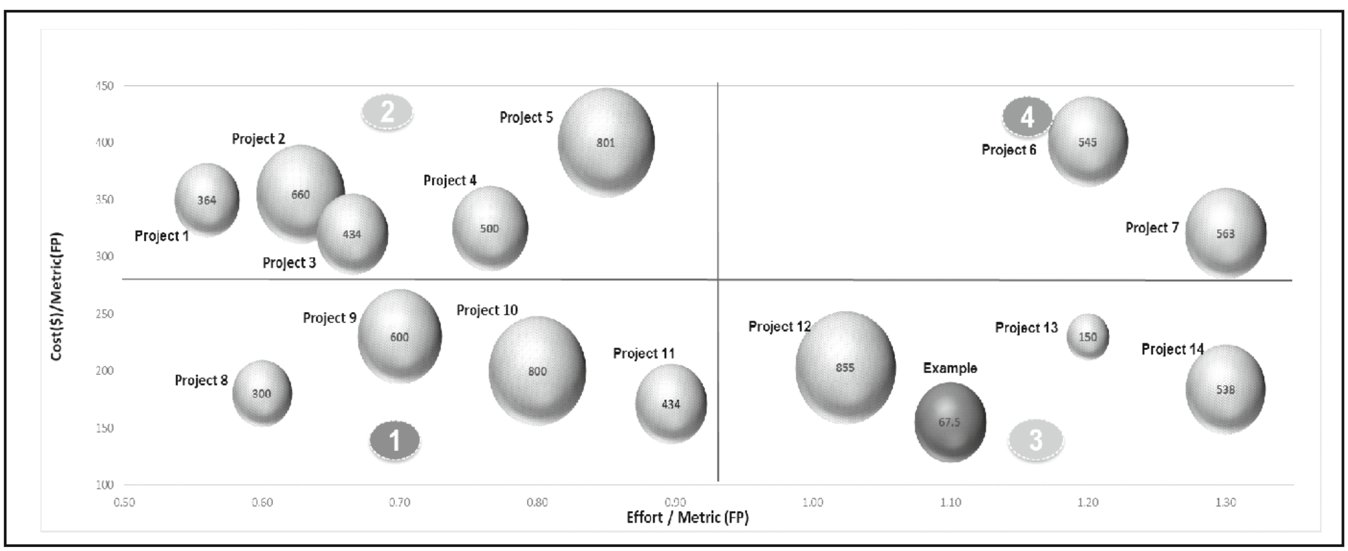

Fig. 2. Description

http://www.opensourcesixsigma.com/v/vspfiles/files/process%20managment%20manual%20sample.pdf, p. 81
[4] C. Jones, Evaluating software metrics and software measurment practices, 2014
[5] A. Albrecht. Available: http://www.ifpug.org/about-ifpug/about-function-point-analysis/, 1979
[6] S. L. Pfleeger, F. Wu, and R. Lewis, *Software cost estimation and sizing methods : Issues and guidelines.* Santa Monica, CA: RAND Corporation, 2005
[7] S. Desai and A. Srivastava, *Software testing : A practical approach*. p. 19. PHI Learning.

About the Author

Chandan K. Jha has completed his Bachelors in Computer Science Engineering from Nagpur University and Masters in Management from Indian Institute of Management. Indore (IIM Indore).

He has experience in Strategy Planning & Implementation, Program & Project Management, Supply Chain Management, Benchmarking & Project Estimation, Vendor Management, Risk Management, Procurement, Business Process Analysis & Re engineering, Operational Excellence, e-Governance, Business & Corporate Strategy.

He has been associated with prestigious firms like Ernst & Young, Tech Mahindra & Grant Thornton in his previous engagements. Over the years, he has worked with esteemed clients such as Unilever, Vodafone, Ministry of Home Affairs & Tata Motors etc. and gained exposure in Supply Chain, Information Technology, Telecom, Education, Government & Manufacturing sectors.