Vikram Tantravahi
13247361

How Function Point Analysis Works?

Introduction:

Every company in today's world works with a standardized mode of operation. There are few essential components, which are a kind of standard set of elements in the way the company deals its projects. Every company in general works with individual clients or a group of clients and deals with their projects from start to the end in different styles.

Especially software has ruled over all the existing business trends and in such software companies, the tools that are used and the softwares that are developed are on a very large scale and are complex and huge structures. In normal terms, all the analysis of product development and deliverables are dealt and explained from the company's perspective and appears to be a complicated subject. But if you have to analyze a company's productivity and help the developers to estimate better costs and productive schedules, *Function Point Analysis* comes in picture.

History:

Allan J. Albrecht in 1979 defined the Function Points at IBM while *Measuring Application Development Productivity*. The problems that are associated with estimating the costs and productivity assessment in an environment with technological software development are all being addressed by the proposals and study of Allan J. Albrecht. Some other problems were the measure of software code, which is usually determined by the lines of code and in suggesting methods to develop efforts predicting mechanisms. This initially published function point metrics were later modified by a group of analysts established as International Functional Point User's Group (IFPUG). This group was set up to set all the standards, clarify the rules, and for the usage and evolutions along with its respective promotion. A brief history of Function Point Analysis can be seen at the table below.

1979	FPs introduced by Alan Albrecht
1984	First FP guidelines
1986	First IFPUG Board of Directors
1994	CPM Release 4.0
2003	ISO standard

Objectives of FPA:

Everyone might wonder how does a company analyze or measure how well are the user requirements being met. The user requirements, the projects schedules and the associated budget with the project. The function point analysis helps the development team in the company to estimate the costs and schedules based on the user requirements that are gathered for each of the development projects. With this procedure, company can easily utilize its resources in a timely and cost effective manner. Some of the objectives of FPA are:

- Measuring software on the basis of functionality that is requested by the user and the functionality provided to the customer.
- Measuring software development and maintenance independent of technology used for implementation of the product.
- Consistent measurement of software development and maintenance across all the existing projects and organizations. This is one of the critical and crucial aspects of the best values of function point analysis.
- There exists a function point counting process that minimizes the overhead existing in the process of measurement.
- It can provide a distinctive mechanism to track and monitor the scope creep.

Benefits of FPA:

Most of the organizations apply Function Point Analysis for following:

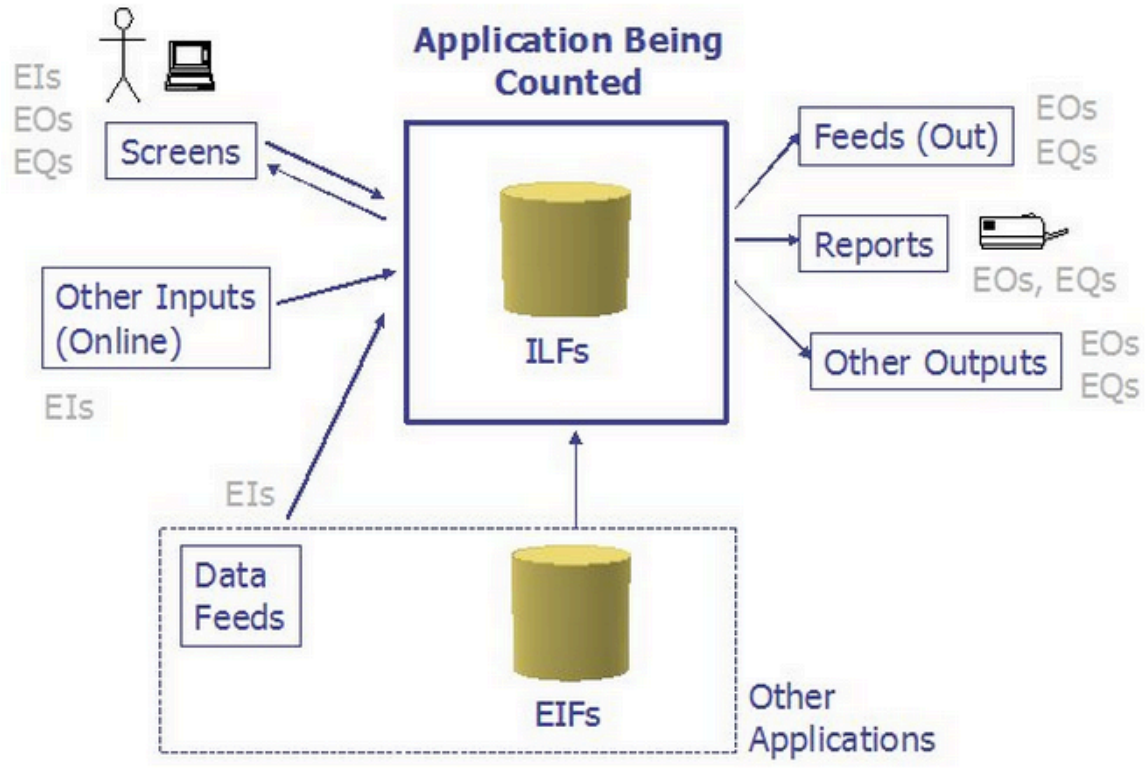
- A powerful tool to determine the size of any desired application packages by counting the number of functions that are included in the package.
- Cost and resource estimation in software development and maintenance.
- Software comparison in the industry and around the similar functionalities.
- In the estimation of the test cases.
- Ability to estimate accurately the project costs, duration and the staffing size.
- Understanding the other related existing metrics like defeat rate of the project, costs included per FP etc.

Implementation of Function Point Analysis:

The initial method of determining FP is the identification of the counting boundary i.e., the border existing between project being measured and the applications in the external or the user domain. The functions that are there in the counting boundary are:

1. Data Functions:
    - Internal Logic Files
    - External Interface Files
2. Transactional Functions:
    - External Inputs
    - External Outputs
    - External Inquiries

These can be understood in a better way from the figure below:



There are some calculations that are made based on the values that are obtained for these individual functions. They are:-

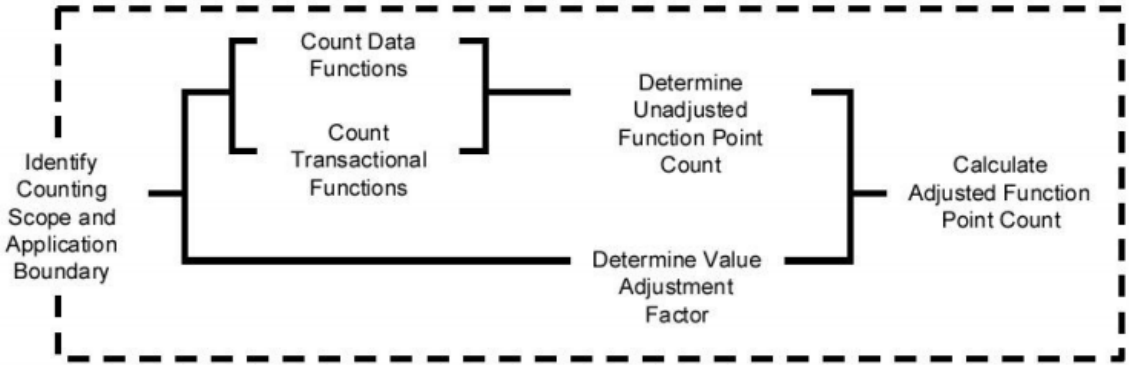
UFP Calculation Table: This is the next step in determining the unadjusted function point (UFP), which depicts the countable functionality given to the user in a specified manner.

Function Type	Functional Complexity		Complexity Totals	Function Type Totals
ILFs	Low	X 7 =		
	Average	X 10 =		
	High	X 15 =		
EIFs	Low	X 5 =		
	Average	X 7 =		
	High	X 10 =		
EIs	Low	X 3 =		
	Average	X 4 =		
	High	X 6 =		
EOs	Low	X 4 =		
	Average	X 5 =		
	High	X 7 =		
EQs	Low	X 3 =		
	Average	X 4 =		
	High	X 6 =		
		Total Unadjusted Function Point Count		

Value Adjustment Factor Table

General System Characteristics (GSCs)	Degree of Influence (DI) 0 - 5
1. Data Communications	
2. Distributed Data Processing	
3. Performance	
4. Heavily Used Configuration	
5. Transaction Rate	
6. Online Data Entry	
7. End-User Efficiency	
8. Online Update	
9. Complex Processing	
10. Reusability	
11. Installation Ease	
12. Operational Ease	
13. Multiple Sites	
14. Facilitate Change	
Total Degree of Influence (TDI)	
Value Adjustment Factor (VAF)	
	VAF = (TDI * 0.01) + 0.65

The overall FPA Procedure:



Citations:

[1] http://www.lrgl.uqam.ca/
[2] IFPUG: Function Point Counting Practices Manual, Release 4.1.1
[3] http://www.umsl.edu/~sauterv/analysis/function_point/FPARP488.html
[4] http://www.softwaremetrics.com/fpafund.htm
[5] http://www.softwaremetrics.com/Articles/using.htm
[6] https://cs.uwaterloo.ca/~apidduck/CS846/Seminars