Software Size Estimation

Software Measurement

- There are two categories of Software Management
  - Direct Measures of the
    - Software process (cost, effort, etc).
    - Software product (lines of code produced, execution speed, defects reported over time etc).
  - Indirect Measures of the
    - Software product (functionality, quality, complexity, efficiency, reliability, maintainability etc.)
  - Project metric can be consolidated to create process metrics for an organization.

Size Oriented Metrics

- Derived by normalizing quality and/or productivity measures by considering the size of the software produced.
- Thousands of lines of codes (KLOC) are chosen as the normalization value.
- Metrics include
  - Errors per KLOC per person month
  - Defects per KLOC per person month
  - Cost per KLOC i.e. cost per page of documentation
  - Pages of documentation per KLOC

Size Oriented Metrics …

- Size oriented metrics are not universally accepted as the best way to measure the software process.
- Opponents argue that KLOC measurements
  - Are dependent on the programming language.
  - Penalize well designed but short programs.
  - Cannot easily accommodate nonprocedural languages.
  - Require a level of detail that may be difficult to achieve.

Function Point Oriented Metrics

- Function oriented metrics use a measure of the functionality delivered by the application as a normalized value.
- Function Point is one such.
- FP=count total *[0.65+0.01*sum(value adjustment factors)]

Function Point

- It indicates the indirect measure of functionality and complexity delivered by the software.
- It is calculated after determining the number of user inputs, outputs, queries, files and interfaces.
- These numbers are further valued higher by multiplying by a weighting factor based on whether the inputs are simple, average or complex.
- If majority of entries are complex, the weight will be higher.

Function Point …

- The weighted count of all entries is further multiplied by complexity adjustment values based on 14 general system characteristics.
- Baseline Data For Project Management is given below

Function Point …

Measure	Compilation Time	Effort in Man Month	Cost	Effort by Skill		
				Lang	Tech	Design
FP						
LOC						
Number of pages in documentation						
Number of reviews						

Function Point …

- The value so derived for the software is called Function Point i.e. FP. It is calculated by using the relationship $FP = F * [0.65+0.001*\sum(Fi)]$, where Fi is the complexity adjustment values for each 14 characteristics i.e. i=1 to 14 and F is the Count Total.

Line of Code

- Line of Code is a measure of the length of code the software engineer will write to deliver the software requirement.
- There is relationship between the number of lines and programming language and quality of design.
- FP and LOC are used to calculate the productivity of skilled resources.
- The metric is “LOC per Man Month” or “FP per Man Month”.
- In order to calculate manpower productivity, we have to measure in precise terms the efforts taken by skilled personnel to deliver the FPs of the entire software.

Line of Code …

- FP and LOC metrics are reliable and hence used for planning resources, team building, costing, business proposals etc.

Review

- During the life cycle of the project, number of reviews is taken to confirm and ensure that activities performed are as per the requirement specification and are achieving the individual goals.
- Reviewing a process, activity or task etc its goal is a routine process, performed as a key role by the project manager.

Stages of Review

- Requirement Reviews
- Design Reviews
- Technical Review

Need of Review

- Every organization has review guidelines and there is a set procedure for each review. The review is performed by an authorized person, review meetings are held, an issue list is made and decisions and actions are taken to resolve them. The issues and the solutions are rechecked in the next meeting.
- Metrics are built from each reviews in all projects on issues, causes, solutions and efforts taken to evolve the solutions.

Need of Review

Review Stages	Participants	Focus
Requirement	● Customer ● Project Manager ● User	● Meeting the Requirement ● Software Requirement Specifications
Design	● Project Manager ● System Designer ● Solution Architect ● Customer	● Architecture ● Performance ● Deliverables
Technical	● Project Manager ● Designer ● Solution Architect	● Quality ● Standards ● Deliverables

Software Size Estimation

- The first step in software estimation is assessing the size of the proposed software.
- The size estimate is then used to compute development efforts and resource estimate, cost and development time.
- The size of software is directly related to requirement specifications.
- The size varies with the complexity of the requirement specifications, business conditions and constraints, implementation platform etc.

Software Size Estimation …

- Software Size is also influenced by the software architecture opted for meeting the requirements.
- The software engineer has to play the role of an estimator too.
- In this role he must have a clear understanding of requirement specifications and the complete scope of the system.
- The foremost important thing is to get the precise requirement upon which the customer and the users are in agreement with the developer.

Software Size Estimation …

- Then this requirement specification has to be decomposed into following structure:-
  - Functional Requirement
  - Non Functional Requirement
  - Features
  - Facilities
  - Interfaces with other systems

Software Size Estimation …

Requirement	Stress
Functional	Process execution; updating, triggering, multiple action, transaction processing, application
Non-Functional	Affecting the system performance, speed, cycle time , volume handling, ease of use
Features	Functions are supported with features, more the features, richer the functionality. Example multi currency calculations, graphic display, multi units, multi locations handling of the business online help etc.
Facilities	Functions provide additional service to the users to deal with I/O of the S/W. Automatic multi advice, printing document, triggering a fax, email message, displaying the documents or result before actions are facilities. Facilities improve utility of the software.
Interface	It requires database interface. GUI interface, interface to capture data. Interface is used to connect or integrate two systems.

Function Point Analysis

- The Software Engineer has to perform various roles.
- They are development engineer, group leader, solution architect and project manager.
- In all these roles it is required to estimate number of attributes of the software for project management, resource management and cost management.
- In due course, these attributes become organization standards for planning, execution and control.

Function Point Analysis …

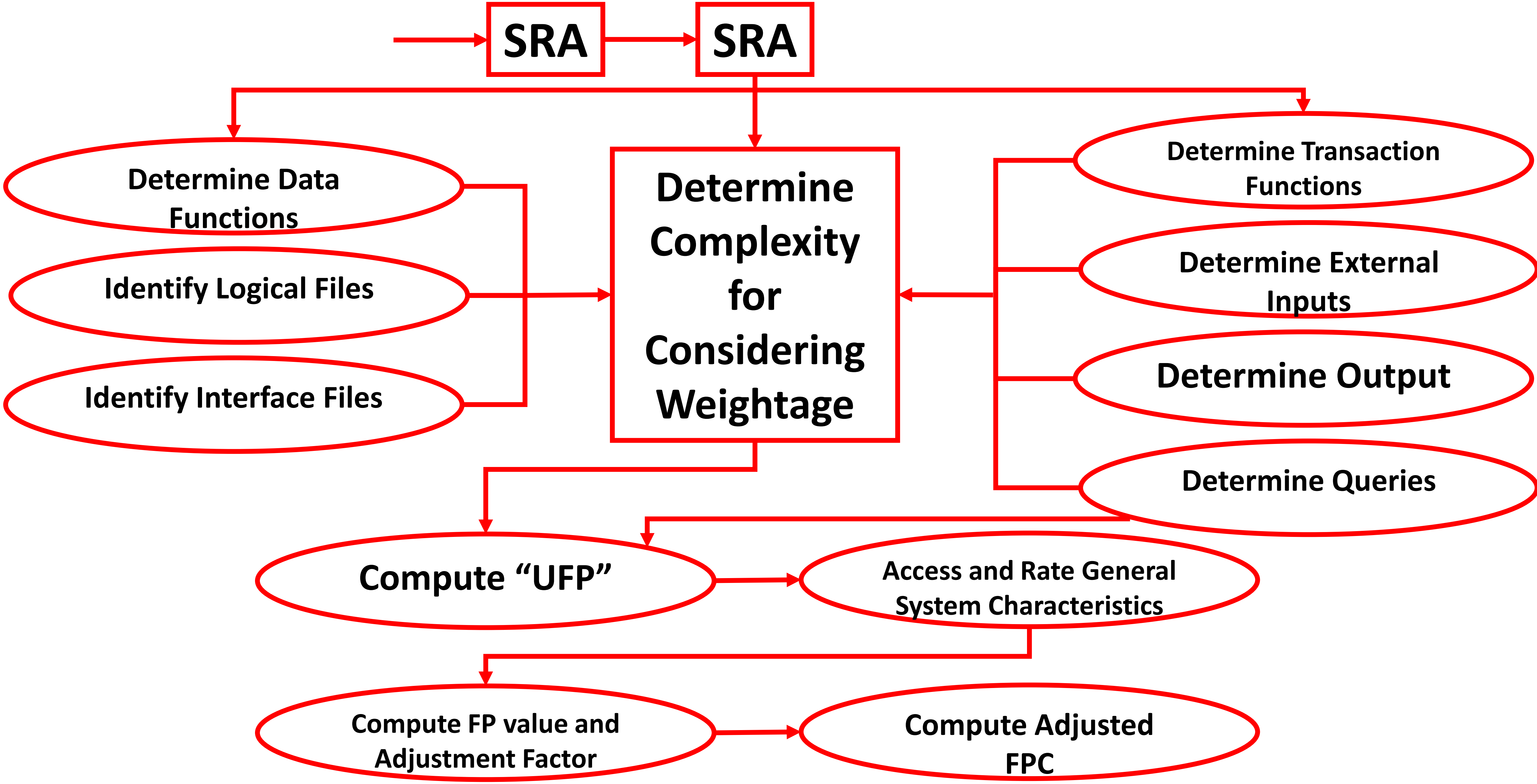
- FPA is used for estimating and measuring the size of the software.
- The stress is on functionality from user perspective.
- It considers user’s requirement and its logical design.
- The size of functionality is calculated in terms of FPC (Function Point Count).

Function Point Analysis …

- The objectives set by International Function Point User's Group (IFPUG) for FPA are
  - Measure functionality of software from user's perspective.
  - Measure the functionality independent of development and implementation technology.
  - Create simple and consistent measures to estimate the size of the software across projects and organizations.
  - Function Point Analysis Method.

Function Point Analysis Methods

- The correct size of software depends on the above inputs and the above depends upon the quality of SRS.
- Process Flow of FPA to Count FPC


SRA
SRA
Determine Data Functions
Identify Logical Files
Identify Interface Files
Determine Complexity for Considering Weightage
Determine Transaction Functions
Determine External Inputs
Determine Output
Determine Queries
Compute "UFP"
Access and Rate General System Characteristics
Compute FP value and Adjustment Factor
Compute Adjusted FPC

Terms used in Function Point Analysis

Terms used in FPA	
• Data Functions	• Relates to holding different types of data complexity and nature of data structure
• Transaction Function	• Considers all processes relating to acquiring data and processing • The complexity of transaction relates to how cumbersome the process is due to rules, checks and controls • Provides methods to process data

Terms used in Function Point Analysis …

Terms used in FPA	
● Data File Types	● Data is stored in files. ● Files are internal logical files and external interface files. ● Record file is internal and master file is external.
● Data Elements and Record Element	● Files are determined by nature of data elements and record elements ● With increase in number of data and record elements, complexity increases.

Terms used in Function Point Analysis …

Terms used in FPA	
● Internal Logical Files	● File stored in the system and maintained by the system and used within the system.
● External Interface Files	● File shared by the system with other systems. ● It is used by the system but not maintained by the system or created by the system.

Terms used in Function Point Analysis …

Terms used in FPA	
● Complexity of data and files	● Complexity of file is estimated by the number of data element types and record element types in each file . ● If the data elements in a file are high, the file is complex. ● Record is a group of data elements created for specific processing needs. ● Number of record types decides the complexity of record file. ● Record elements are determined on the basis of updation, attributes of sub entity etc.

Elements type in File and Unadjusted FP

File containing Data/Record Element Types	Complexity	UFP
<20 / <2	Simple	5
<50 / <5	Normal	8
>50 / >5	Complex	15

Unadjusted Function Point

- As we know that the “function point” is the "unit of measurement" to express the amount of business functionality an information system (as a product) provides to a user.
- Unadjusted Function Points refer to total counted function points in a project.
- A transaction is triggered when an event takes place in the environment. This event calls for the updation of some status through a Transaction Process.
- For example if a person joins the organization, an Employee Master File will be updated; Account Master will be uploaded etc.

Unadjusted Function Point …

- A transaction is effected by external input and concluded by an external output.
- In a pay roll system, the employee code of a new employee is the external data input.
- Then when it updates the strength of payroll employees by one, it is an external output.
- When transaction is a type of viewing, assessing or querying, in which input or output is affected, then it is called external query transactions.

Transactions V/S Unadjusted Function Point

Transaction Data/Record Element Types	Complexity	UFP
<20 / <2	Simple	3
<50 / <5	Normal	5
>50 / >5	Complex	7

Unadjusted Function Point Calculations

Name data / transaction	DET	RET/ FTR	Complexity	UFP
• File 1	30	4	Normal	8
• File 2	20	3	Simple	5
• File 3	60	2	Complex	15
• Function – 1	5	2	Normal	5
• Function – 2	6	3	Simple	7
• Function – 3	2	3	Complex	3
• DET Data Element Type				

Unadjusted Function Point Calculation …

- UFP calculation is a systematic method based on certain characteristics of the software proposed to be developed.
- Considers data and transaction of the software, largely depends on the designer’s conceptualization of the design and architecture.
- It means UFPs may differ from person to person if two designers are estimating the size in their individual views.

Unadjusted Function Point Calculation …

- IFPUG has identified following 14 characteristics that influence the function points calculated for the software.
- The rating is a quantitative measure of influence of these characteristics on the function points.
- The rating is on the scale of 0 to 5.
- Lower rating means less influence and vice versa

Unadjusted Function Point Calculation …

- The characteristics are
  - Data Communication
  - Distributed Processing
  - Performance
  - Configuration
  - Transaction Rate
  - On Line Data Entry
  - End User Efficiency
  - On Line Update

Unadjusted Function Point Calculation …

- The characteristics are …
  - Processing
  - Reusability
  - Installation
  - Operations
  - Multiple Sites
  - Change Requirement

Unadjusted Function Point Calculation …

- Depending upon the values of these 14 characteristics, Value of Adjusted Factor (VAF) is calculated.
- VAF= (Sum of Ratings) *0.01 + 0.65.
- The value of VAF will adjust UFP for the influence arising out of these characteristics.
- High VAF means strong non functional component.
- FPC = UFP * VAF

Advantages of Function Point Count

- FPC is independent of the technology i.e. OS, programming language, database, developer productivity, methodology etc.
- The FPC concept is simple.
- Hence it becomes a good quick measure for any comparative analysis especially between software and productivity of different groups.

Disadvantages of Function Point Count

- Estimation is based on subjective judgment of the software designer.
- It varies from person to person.
- The choice of design and architecture is also subjective.
- As the software system is decomposed into multiple components and then into data and functions, the estimate fails to cater integration requirement of these components.
- Identification of files is tricky and uncertain.
- Internal processing complexity due to complex business rules, algorithms, calculations etc. is not weighted.

Mark II Function Point Analysis

- Proposed by Charles Symons in 1990.
- Treats the software system as a collection of logical transactions.
- A transaction is triggered by an event of the interest of the user.
- A logical transaction is the one that has a consistent process for all events that trigger the transaction.
- It has a clear input, processing logic and output within the boundaries of the scope of the transaction.
- Input and output are Input Data Element and Output Data Element.

Mark II Function Point Analysis …

- Step 1: Determine Scope of Count
- Step 2: Identify Logical Transactions
- Step 3: Count the Entities
- Step 4: Count the Data Elements(DETs)
- Step 5: Calculate the Function Point Count(Index)

Mark II Function Point Analysis …

- Step 1: Determine Scope of Count
  - Type of count: Development/Application or Change?
  - What is the desired accuracy of the count? Earlier means less accurate.
  - What is the boundary of the count, i.e. what logical transactions to include?

Mark II Function Point Analysis …

- Step 2: Identify Logical Transactions
  - Analyze the requirements to find the Logical Transactions,
  - Logical Transaction are “The basic functional component of MK II FPA. The smallest complete unit of information processing that is meaningful to the end user in the business. It is triggered by an event in the real world of interest to the user, or by a request for information. It comprises an input, process and output component. It must be self-contained and leave the application being counted in a consistent state”.

Mark II Function Point Analysis …

- Step 2: Identify Logical Transactions …
    - For example, a requirement to “Maintain Customer” will have to correspond at least these five transactions.

Mark II Function Point Analysis …

- Step 3: Count the entities
  - The processing element of the Function Point Count corresponds to the storage and retrieval of information about Entities of interest to the external world.
  - Draw an Entity Relationship Diagram (ERD) to find the primary Entities in the system
  - Entity (or Data Entity Type) is "A fundamental thing of relevance to the user, about which information is kept. An association between entities that has attributes is itself an entity".

Mark II Function Point Analysis …

- Step 3: Count the entities …
  - Counting rules:
    - Each Logical Transaction must reference at least one Entity.
    - A Logical Transaction counts each Entity it refers to; the same Entity can be counted for more than one Logical Transaction.
    - Sub-Entities are counted separately from the parent when different processing logic is used for the Sub-Entity.
    - Recursive Entities are counted twice for any Logical Transaction that traverses the hierarchy.
    - Count the System Entity once if the Logical Transaction refers to any non-primary Entities; don't count the non-primary Entities themselves

Mark II Function Point Analysis …

- Step 4: Count the Data Elements(DETs)
  - Data Element Type (DET) is “a unique user recognizable, non-recursive item of information. The number of DET’s is used to determine the input and output size of each Logical Transaction”.
  - An Input DET (In-DET) comes from outside the system boundary and changes the state of the system. They are concerned with acquisition and validation of data.
  - An Output DET (Out-DET) goes back across the system boundary so a user can see/use it. They are concerned with formatting and presentation of data.

Mark II Function Point Analysis …

- Step 4: Count the Data Elements(DETs) …
  - Counting Rules:
    - Each Logical Transaction must have at least one In-DET.
    - Count one In-DET for each distinctly validated DET or group of DETs.
    - In the absence of other In-DETs, a Trigger for an event is a DET.
    - Each Logical Transaction must have at least one Out-DET.
    - Count one Out-DET for each distinctly formatted DET or group of DETs.

Mark II Function Point Analysis …

- Step 4: Count the Data Elements(DETs) …
  - Counting Rules: …
    - All Error/Confirmation messages for a Logical Transaction are lumped together as one Out-DET.
    - Count each Total as an Out-DET, separately from the associated detail DET.
    - If an In-DET is redisplayed as part of validation, then also count it as an Out-DET.

Mark II Function Point Analysis …

- Step 5: Calculate the Function Point Index
  - Calculate the functional size or Function Point Index (FPI) of each Logical Transactions whose input and output components cross the system boundary.
  - FPI = 0.58 * Count(In-DET) + 1.66 * Count(Entities-Referenced) + 0.26 * Count(Out-DET)
  - The functional size of the system, i.e. the total FPI, is the sum of the sizes of each of the Logical Transactions.
  - Total FPI = Sum (FPI of all Logical Transactions)