

by  
**Tom Cagley,**  
**VP Software Measurement and IT Performance Improvement**

## ARTICLE

Project sizing is a critical component of software engineering. Sizing is a tool both for managing expectations and for directing work. Size is the basis for developing estimation and accumulating historical performance data that provides a tool for process improvement, estimation and work allocation. *Knowing the size of a project, the amount of effort, the duration and lessons learned from a completed project provide a quantitative platform for process improvement.*

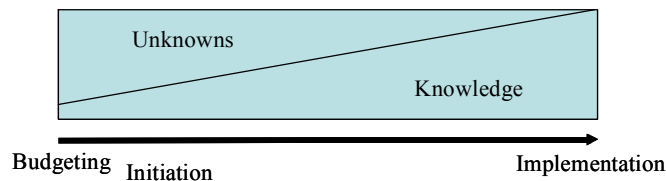
As an adjunct, size is a component of mature change management processes (using your estimation process for pieces during the project). Sizing is an integral part of these processes and other project management processes, and it should be a core competency in all IT organizations however, it tends not to be.

Function Points are a standardized method for measuring the functionality delivered to an end user (See summary at end of article). In this vein, the metric was designed with the following objectives<sup>1</sup>:

- Measures functionality that the user request and receives
- Measures software development and maintenance independently of technology used for implementation
- Simple enough to minimize the overhead of the measurement process
- A consistent measure among various projects and organizations

Quick and Early Function Points (QEFP) build on these objectives to provide a consistent method for sizing when all that is known are 'words'. QEFP reduces the effort required for sizing and the training investment required for using the method. QEFP leverages grade school English as well as the knowledge of basic function point analysis.

While it has been reduced to truism, early in a project what you don't know far outweighs what is known. Project managers need to get the most out of what they do know--a set of high level words (initial requirement that can range from a paragraph written on the proverbial cocktail napkin to a business description).



<sup>1</sup> Function Point Counting Practices Manual Release 4.2.1, January 2005, page 2-2.



## Estimation Simplicity: Turning Words into Numbers

[www.davidconsultinggroup.com](http://www.davidconsultinggroup.com)

The goal of Quick and Early Function Points (QEFP) is to provide a method for systematically turning words into numbers, while respecting the rules of function point counting. In other words, to consistently get the most out of the little that is known. QEFP has been designed with the following capabilities (along with those typically associated with function points):

- Easy to use when all you know is words
- Basis for a better early estimate
- An extensible process
- Create data that will be useful later in the project
- Applicable to Agile methods

QEFP leverages rules in the current IFPUG Counting Practices Manual (CPM 4.2.1) as a starting point, then simplifies, simplifies and then simplifies again. The point is to bring the model to the point they can be applied to the level of words that describe business requirements (or stories).

The method begins by making several simplifying assumptions. The first of the assumptions is that all data and transaction functions are considered to be average (see end of article for a quick overview of function points). This relieves the person developing the size of the need to count data element types (DET), file types referenced (FTR) and record element types (RET), which at this point in the process is probably impossibility anyway unless you're willing to guess. Note that at no point would I suggest that if you know more than the QEFP requires, you do not use what you know.

The second assumption is to set the Value Adjustment Factor (VAF) equal to one. The VAF is a tool used with function point analysis to adjust the function point count based on the answer to 14 questions that 'rate' general system characteristics. General system characteristics reflect the application being developed (or changed) and may be impacted by the project being measured. Early in a project, unless you have previous knowledge, how can you know what the VAF will be without having intimate foreknowledge of the solution? Unless you know the VAF from full counts done earlier, it is easier (and potentially more accurate) to assume the VAF to be one. The third and final assumption is that all counts using this method will be approached as if they are enhancement projects. The Function Point Analysis methodology stipulates three types of counts:

1. Application Count – represents the functional size of the installed application (also called a baseline).
2. Development Count – measures the functions provided with the first installation of an application.
3. Enhancement Count – measures the modifications (added, changed and/or deleted functionality) delivered at the completion of the project being sized.

Stipulating the choice of count type and using the broadest approach simplifies the preparation for a count, no data is lost which allows the counter to use the data to be used for other purposes later in the processes. A side benefit to this approach is drawing attention to probability that the solution will be integrated with other applications. Integration increases the amount of effort required to deliver the project as well as the project's 'size'.

Copyright © 2007 David Consulting Group  
1770 E. Lancaster Ave, Suite 15  
Paoli, PA 19301  
+1.610.644.2856 (F) +1.866.293.0120

None of these simplifications offer *carte blanche* to abandon IFPUG's CPM. The crux of the CPM is a repeatable process backed by scholarship, data and years of process improvement. Variations that abridge compliance will supply you with erratic results, which will erode confidence and value. QEFP embraces the CPM and is a means of sizing quickly and very early in the project life cycle.

### **The Process for Using QEFP**

The QEFP method uses a simple 3 (maybe 3.5 step) process for functional sizing by leveraging the assumptions noted earlier in the paper. The process is:

1. Identify the counting scope and application boundary.
2. Segregate functional, non-functional and technical requirements:
  - Identify and count the data functions to determine their contribution to the unadjusted function point count.
  - Identify and count the transactional functions to determine their contribution to the unadjusted function point count.
3. Calculate the function point count.

Determining the scope and application boundaries in simplest terms is deciding what is being sized and where functionality will cross boundaries. IFPUG's CPM and other papers provide a wide range of examples and guidelines for determining scope and application boundaries. Boundaries are important, as function points recognized transactional functions when they cross a boundary. Crossing the boundary allows the transaction to become user recognizable (Usage Note: Think – transactions = movement of data across boundaries).

The second step (or step and half) provides the real magic of QEFP! Start by segregating the functional, non-functional (e.g. availability, cost of ownership, maintainability) and technical requirements (e.g. number of people who will use the system and where they are located, the numbers and types of transactions). Using three piles of requirements, Function Point Analysis addresses functional requirements. The functional requirements that define what the project is to accomplish in business terms. An example, the requirement 'maintain customer data and do it fast' can be broken into two requirements: The first is a functional requirement, 'maintain customer data,' and the second is a technical requirement, 'do it fast'. While non-functional and technical requirements will required implement in the solution, they do not impact the concept of functional size which is the theory at the core of function points. These types of requirements are dealt with as an addition to the estimate or as part of the productivity signature used to translate size into effort.

As a side benefit, this segregation or classification of requirements will allow the project participants to review the requirements in order to provide a basis for understanding, establish balance and ensure consistency (which fits with the Requirements Management process area in the CMMI). Functional metrics can be used to enhance the analysis of any project (plan-based or agile). Always recognize what your measurement processes can supply to the development process. The example above is just one of the synergies that support delivering functionality and value to a user and customer.

Once you have created your pile of functional requirements, the QEFP process requires that you identify and count both the data and transactional functions. Ninety-six percent (more on the last

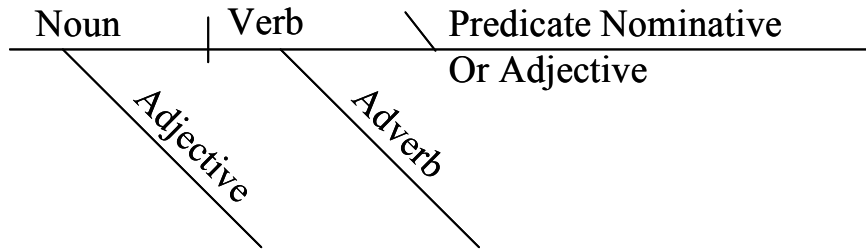


david consulting group

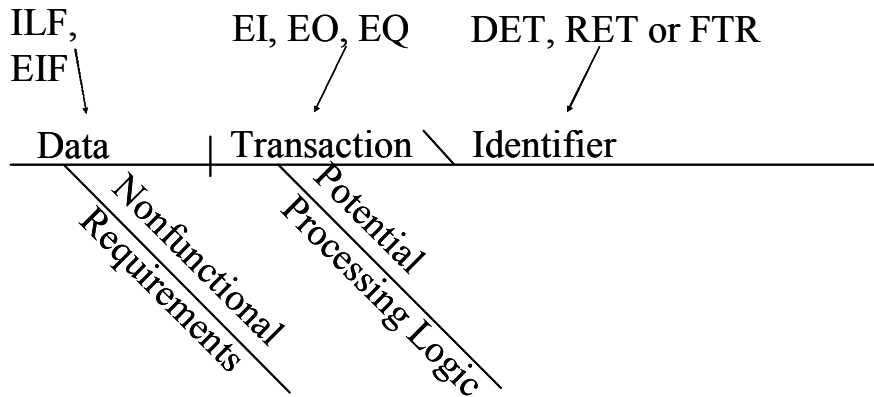
# Estimation Simplicity: Turning Words into Numbers

[www.davidconsultinggroup.com](http://www.davidconsultinggroup.com)

four percent later) of effort in the process is in recognizing the functions. QEFP uses a tool many people were introduced to in elementary school, sentence diagramming, to facilitate usage. The diagram below shows an outline basic sentence diagram:



How does a sentence diagram translate into function points in a manner that will allow you to quickly estimate the size of a project? Nouns indicate data that are Internal Logical Files or External Interface Files (the stuff that gets maintained by transactions). Verbs describe transactions (movement or activity) that are External Inputs, External Outputs and External Inquires. Other components such as adjectives, adverbs and others (I think my English teacher made up some parts of speech to torture us as students) reflect other components of the function point model or other types of requirements. This is important information, but not germane to QEFP. The following diagram provides a visual summary.



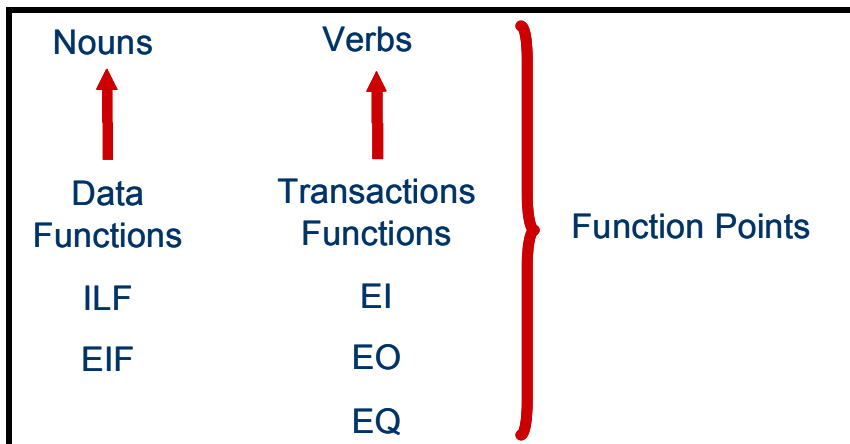
I suggest three cautionary notes. The first is to remember that no matter how much fun sentence diagramming is, the goal is not diagram the whole requirement sentence, but rather to use the logic of the diagramming process to get to the measurable component. Fearlessly sift through the data to focus solely on noun and verbs (sizing data). The second cautionary note is that you still need to understand the differences between the different types of data functions and transaction functions so that you can translate the words into proper functions (this is a bit of foreshadowing for trigger words). The third note is that you will need to apply the rule of uniqueness unmercifully. In the function point methodology, functionality is only counted once.

Copyright © 2007 David Consulting Group  
1770 E. Lancaster Ave, Suite 15  
Paoli, PA 19301  
+1.610.644.2856 (F) +1.866.293.0120

For example, if the requirements of a project call for the need for customer data, there would still only be one customer ILF. Uniqueness is applied to all functions

(data and transactions).

To support making sizing easier and more applicable earlier in the process, QAFP leverages the concept of trigger words. The relationship between sentence components and function point components provide a platform to translate requirements into a functional size. To make the process faster and more consistent trigger words can be used to provide context.



Trigger words leverage a physiological and psychological response that can be summarized as 'contextual queues influence recognition'. A list of words (trigger) will act as a means of identifying contextual queues to draw your eye. The ability to draw your eye (to focus attention) will help you to quickly develop estimates of the functional size. Nouns, trigger words for data functions, refer to a person, place, thing, event, or substance. Creating a list of nouns has less contextual value than a list of trigger words to identify potential transactions. The following table summarizes some typical trigger words for transactions:

External Inputs	External Outputs	External Inquires
Add	Adjust	Browse
Allocate	Export	Display
Assign	Generate	Enquire
Associate	Notify	Extract
Change	Print	Inquire
Create	Report	List
Delete		Pick List
Import		View
Make Inactive		
Modify		

The QAFP methodology uses trigger words to attract the eye. Upon recognition a trigger word, the methodology invokes standard function point qualification rules to determine whether a transaction exists and whether you have counted it before (the uniqueness rule again).



## Estimation Simplicity: Turning Words into Numbers

[www.davidconsultinggroup.com](http://www.davidconsultinggroup.com)

Words are a bit of a wet baby. Trigger words are influenced by language with is a function of culture (personal, organization and national). The QEFP has been used in both Anglophile and Francophile cultures. In each case the organizations had to spend a few moments tuning the list of trigger words to reflect how they use language. This is does not need to be a difficult process. Leverage a quick brainstorming or affinity diagramming session to drive out a list of words, then test them in action.

### Summary:

Early in the life cycle of all projects, you will be asked for an estimate. Estimating is a function of size and other attributes. Quick and Early Function Points (QEFP) provides a method for converting words into numbers, requirements into size. QEFP provides a construct for this conversion very early in the project based on a common set of rules tailored to specifically deal with what you know, don't know and CAN'T know at the beginning of a project. QEFP is based on the standard function point analysis process tailored for application early in a project. In short, QEFP turns perfectly good words into numbers.

Thomas M. Cagley Jr.  
[tcagley@davidconsultinggroup.com](mailto:tcagley@davidconsultinggroup.com)  
440.668.5717

For more information:

- The David Consulting Group
  - [www.davidconsultinggroup.com](http://www.davidconsultinggroup.com)
- International Software Benchmarking Standards Group
  - [www.isbsg.org.au](http://www.isbsg.org.au)
- International Function Point Users Group
  - [www.ifpug.org](http://www.ifpug.org)
- Software Engineering Institute, Project Sizing & Estimating
  - [www.cmu.sei.edu](http://www.cmu.sei.edu)
- Sentence Diagrams by Eugene R. Moutoux
  - [www.geocities.com/gene\\_moutoux/diagrams.htm](http://www.geocities.com/gene_moutoux/diagrams.htm)

For further information on this topic or to talk to a DCG expert, contact us at 610.644.2856 or send inquiry e-mail to [info@davidconsultinggroup.com](mailto:info@davidconsultinggroup.com)

Copyright © 2007 David Consulting Group  
1770 E. Lancaster Ave, Suite 15  
Paoli, PA 19301  
+1.610.644.2856 (F) +1.866.293.0120

## **Function Point Analysis Concepts**

### **Data and Transaction Function Types in Function Point Analysis**

Function point analysis measures software by quantifying the functionality the software provides to the user based primarily on logical design. Function points look at functionality provided by data and transactions. These categories can be broken into five basic items (all applications have all or most of these items regardless of what they are called). The basic groups of functionality are listed below with a short definition.

#### Data Functions

- Internal Logical File (ILF) – user identifiable group of logically related data or control information maintained within the boundary of the application.
- External Interface Files (EIF) – user identifiable group of logically related data or control information referenced by the application, but maintained within the boundary of another application.

Complexity for data functions is determined by counting the number of data elements (DET) and the record elements (RET) maintained or referenced.

#### Transaction Functions

- External Inputs – an elementary process that processes data or control information that comes from outside the application boundary.
- External Outputs – an elementary process that sends data or control information outside the application boundary (must contain math, create derived data and/or maintain an ILF).
- External Inquiry – an elementary process that sends data or control information outside the application boundary (must contain math, create derived data and/or maintain an ILF).

Complexity for transaction functions is determined by counting the number of data elements and file types referenced (FTR).

The CPM 4.2.1 details the rules for determining a data and / or transactional function. The CPM also includes the detailed rules for counting DETs, RETs and FTRs.