



## Joint Application Design & Function Point Analysis – the Perfect Match

By Sherry Ferrell & Roger Heller

### Introduction

The old adage “It’s not what you know but when you know it that counts” is certainly true when it comes to identifying, understanding, and defining user requirements. In today’s world of Rapid Application Development and time boxed schedules it is critical that software developers have a clear understanding early in the project life cycle of what is to be developed and an accurate estimate of how long it should take. To accomplish these objectives it is necessary to have joint participation of the business users and the system developers in the requirements definition processes, as well as a standard technique to utilize the information produced to create accurate project estimates. Through work with our clients we have developed a methodology that merges Joint Application Design (JAD) techniques, Function Point Analysis (FPA) and project estimating to achieve these goals. The remainder of this article describes the Functional Requirements Rapid Analysis methodology in further detail.

### Function Point Analysis

An application’s size is a critical measure to understand since it provides the basis for establishing project estimates. Function Points are a unit of measure; they measure the size of a software application. Conceptually, function points are similar to how square footage can be used to measure the size of a house. In addition to measuring application size, Function Point Analysis is extremely useful in sizing and estimating projects, managing change of scope, measuring productivity, and communicating functional requirements.

One of the initial design criteria for function points was to provide a mechanism that both software developers and users could utilize to define functional requirements. It was determined that the best way to gain an understanding of the users’ needs was to approach their problem from the perspective of how a computer can help a user do their job. Therefore, one of the primary goals of Function Point Analysis is to evaluate a system’s capabilities from a user’s point of view. This is one of the main reasons that marrying the JAD techniques and Function Point Analysis is such a powerful tool in gathering requirements.

Continued on page 2

### In This Issue

Joint Application Design & Function Point Analysis - the Perfect Match .....1

Key Measures for Project Management .....1

## Key Measures for Project Management

By Steven M. Woodward, Q/P Management Group of Canada

### Introduction

Identification of the right project measures needs to be evaluated carefully. Not enough measures and the analysis may be incomplete and misleading. Too many measures and the accuracy of the information collected by the teams may degrade, and the analysis will be more complex.

Depending on your organizations and project goals, different sets of measures with appropriate definitions will need to be established and collected. The “Goal/ Question/ Metric” is a popular approach to identify the measures to collect, by first establishing your projects/ organization goals, then identifying specific questions, then finally the identification of the measure itself. The careful analysis of your goals, the questions which the measures need to answer, and the measures and definitions are key preliminary steps, which cannot be underestimated in their importance.

The six most common software project measures are Function Points, Effort, Cost, Schedule Duration, Defects and Lines of Code. This article will focus on these popular measures, describing their usage, strengths, challenges, and implementation.

### The Measures

#### Function Points

**Definition:** Function Points are a unit of measure that quantifies the volume of software functionality contained in an application or project. Just as liters are a volumetric measure of liquid, Function Points are a volumetric measure of software functionality.

**Description:** The International Function Point Users Group (IFPUG) provides a standard framework with guidelines to communicate and promote consistent functional sizing of any software from real time, military or telecommunication to batch financial applications using function points.

**Utilization:** Software functionality is what IT organizations develop, enhance and support. Therefore function points are a key unit of measure when assessing project size, establishing estimates, and managing risk.

Continued on page 5

The Function Point Analysis process defines the requirements in terms of five basic Function Types. Two of these Function Types address the data requirements of an end user and are referred to as Data Functions. The remaining three types address the user's need to access data and are referred to as Transactional Functions.

The first data function is known as an Internal Logical File (ILF). Its purpose is to store the business data that the user organization is responsible for maintaining. The second data function is called an External Interface File (EIF). As its name implies, this file is outside of the application being analyzed. While it contains business data, the file contents are maintained by a different application. In practice, the application being analyzed can only reference this data. Each logical group of data used by the application is categorized as either an ILF or EIF.

Once the data needs have been addressed, the next step is to categorize the various transactions the application provides to the users. The first transaction is referred to as an External Input (EI). The purpose of an EI is to maintain the data on the files owned by the application (ILFs). The other transactions, External Inquiries and External Outputs, address the users needs to receive data from the application. The External Inquiry (EQ) transaction allows the users to directly retrieve stored raw data from the files (ILFs and EIFs). The last transaction is an External Output (EO). The data contained on an EO is created from the raw data on the files (ILFs and EIFs) through derivation or mathematical logic contained within the application.

When counting the function points of an application, the function point counter maps each of the user recognizable data files and transactions to one of the 5 Function Types. The mapping is determined based on a set of standard rules that are maintained by the International Function Point Users Group (IFPUG). Each of the functions is assigned some number of function points. The number of function points assigned is determined by standards maintained by IFPUG. The function point values of each file and transaction are added together to determine the function point size of a project or application. The function point size of a project is the input needed to establish an estimate of project effort, schedule and cost. An integrated component of the JAD workshop is the collection of functional information needed to establish the function point count.

### ***Joint Application Design Workshop***

Traditional software development requirements gathering techniques rely on a series of sequential steps centered on structured interviews. The interviews result in pages of detailed notes that eventually are merged together to create functional requirements. For small development efforts this

approach may work. But for large, complicated multi-department solutions this approach often results in an unclear understanding of the need, inconsistent vision of the solution and lack of end user ownership and support. Joint Application Design techniques are structured to eliminate the miscommunication and confusion that results from the traditional techniques.

Joint Application Design (JAD) workshops produce a clearer, more complete definition of the system requirements. Since the workshop brings together both users and developers, requirements can be better understood and defined more quickly than by traditional requirements gathering techniques. During the JAD workshops, users and systems staffs establish a partnership that will lead to joint ownership of the project. A JAD workshop consists of three primary activities: planning and preparing for the workshop, conducting the workshop, and the workshop wrap-up.

### ***Planning and Preparation Phase***

This is a critical activity since it will set the tone for the remainder of the data collection activities. During this stage of the process the JAD facilitator works with the business and systems sponsors to identify and document the business needs from which the requirements will be developed. The JAD facilitator interviews key individuals who are knowledgeable of the specific business needs. This information provides a framework to structure the workshops and helps to ensure that the end results will be consistent with the user expectations.

In preparation for the interviews, the JAD facilitator reviews any high-level functional requirements documents, or documents that reflect the business needs. The facilitator uses this information to guide the interviews with the users and systems subject matter experts (SMEs).

The focus of the interviews will depend on the type of project being evaluated. For an enhancement to an existing application the subject matter experts will provide an overview of the current application and its interfaces with other applications. For planning purposes, the JAD facilitator may review user documentation or other existing systems documents to determine the extent of the project impact. If the project is a new development effort the SMEs will be asked to provide a high level overview of the business needs and to identify any possible interfaces that may be required. The information gathered during these interviews and the documentation review is utilized to identify the scope of the JAD workshop. A Subject Area List is prepared and the workshop participants are identified.

The JAD workshop participants must include users and systems staff with the knowledge of the business and systems

issues that the JAD will address. Business users and systems staff will serve as subject matter experts. Potential participants include:

**Sponsors** – These individuals have knowledge of the strategic business issues, organizational goals and constraints.

**Business users or their representatives** – These participants have extensive knowledge of the current business functions and future business requirements and strategies. They must have authority to make decisions on the functional aspects and deployment of the project or system.

**System Experts** – These individuals have extensive knowledge of existing systems and technology experience as it relates to the application.

**System Specialists** – These participants provide expertise on a specific topic such as database performance. They only attend the workshops when required for their specific topics.

**JAD Leader** – This is the individual who will facilitate the JAD session. They should have effective facilitation skills, be highly experienced in the JAD and Function Point Analysis Methodologies with a strong application development background.

**JAD Analysts** – This individual works with the JAD Leader to document the outcome of the workshops and capture the required information to establish the function point count. The individual should have effective facilitation skills, be highly experienced in JAD and Function Point Analysis Methodologies with a strong application development background.

The final activity in this phase is to use the information collected during the interviews to define the scope of the effort. This is needed in order to determine the amount of time that is required and the number of JAD sessions that will be needed to fulfill the workshop objectives.

### *Conducting the workshop*

To collect the requirements and function point analysis (FPA) information, a series of JAD sessions are conducted based on the scope of the effort and the number of subject areas to be covered. The length of each session depends on the extent of the impact the project will have on a specific subject area. The JAD leader facilitates the session. A structured approach is followed that is designed to obtain both the requirements as well as the information required to complete the function point count.

### **Kickoff Session**

The initial session is designed to Kickoff the project. It is attended by all of the participants, user and systems sponsors, SMEs, and others as appropriate. Several points

are addressed during this session. The facilitator provides an overview of the JAD process, describing the sequence of events; the roles and responsibilities of the participants and the materials that should be brought to each of the sessions. The Sponsor of the JAD provides an overview of the business needs that are being addressed by the JAD workshop. If the focus is on an enhancement project, the Sponsor will provide an overview of the existing application and an explanation of how the enhancement will address the business needs. This is also an opportunity for the sponsor to show their support for the project and commit the needed resources to the effort.

This session also provides the JAD facilitator an opportunity to verify the project scope and the subject areas that will be included in the JAD scope. It is important that the group reaches consensus on the project scope before proceeding. This is the first opportunity to demonstrate the importance of consensus building to the overall JAD process.

An output from this session will be lists of Assumptions and Issues related to the project scope. Depending on the Assumptions and Issues identified, an Action Items List may be developed which will include follow-up responsibilities and assigned due dates. Finally, the JAD schedule is reviewed and re-confirmed with the participants to obtain their commitments to attend.

### **Subject Area Session 1 to n:**

The subject area sessions are designed to capture the specific requirement details and the information needed to establish the function point count. At least one JAD session will be conducted for each subject area. The actual number of sessions and the length of each session are determined by the impact the project has on the specific subject.

The session begins with the primary Subject Matter Expert (SME) providing an overview of the subject area to the session participants. The SME provides a high level overview of the impact of the enhancements and/or the new development required based on the business needs. This is a level setting presentation to ensure that all of the participants are approaching the business problem from a common understanding.

The facilitator leads the participants through the collection of the functional requirements based on the identified business needs. To aid in the data collection needed to establish the function point count, the facilitator will structure the discussion to ensure that each of the five FPA function types are addressed. Throughout the Subject Area sessions the JAD Analyst is capturing the functional requirements, and translating them into data and transaction functions for input into the function point model.

The results from each JAD session, including the function point details, are documented and reviewed with session participants. Assumptions and Constraints are identified, documented, and reviewed with participants. Unresolved issues are identified and follow-up activities are assigned to specific participants if required. When the subject area workshops are completed a wrap-up session is used to finalize the results.

### Workshop Wrap-up

A final review session is conducted with the workshop participants. In preparation for the session the workshop results are consolidated and distributed for review. The JAD document includes the following:

- Fully-documented requirements including project scope, requirements by functional entity, editing, validation, and processing requirements
- Fully documented function point count(s)
- Assumptions, Issues, and Constraints
- Terms and Definitions
- Internal impacts (operational, department, etc.)
- External impacts (business, regulatory, etc.)
- A project attribute analysis that documents the soft factors that can impact a software development project

Final consensus is reached on the solution and the sponsors provide direction on the next steps required for initiating the project. The documentation, requirements definition and function point details, from the JAD workshop provides valuable information that can be used in the estimating process.

### The Final Step – Project Estimation

Once requirements are clearly defined and the project has been sized, it is important to determine the effort required to develop the system. Function Points are the industry-accepted metric used to establish the size of a software development project. Many estimating models and tools utilize function points as an estimating component.

Through experience we have found that the most effective software estimates are based on historical data that is representative of the effort being analyzed. At Q/P Management Group we utilize our historical benchmark database of thousands of software projects and applications in establishing project estimates. This project database along with industry accepted estimating techniques allows us to establish accurate estimates of effort, schedule, cost, and staffing requirements. The ability to provide reliable estimates for these major project components is critical to a project's success. Our models and benchmark data are available to our clients.

### Summary

This approach defines the future system from the business users' perspective. A team of business managers and system developers who define the business policies, rules, processes and procedures that will govern the new system drive the process. This team is taken through a series of facilitated sessions that define in detail the business requirements including:

information requirements, business logic, processing rules, input criteria, output requirements, update functionality, control processes and interfaces. As functional requirements are identified the information is translated into function point terminology, which results in a function point count that establishes the size of the project.

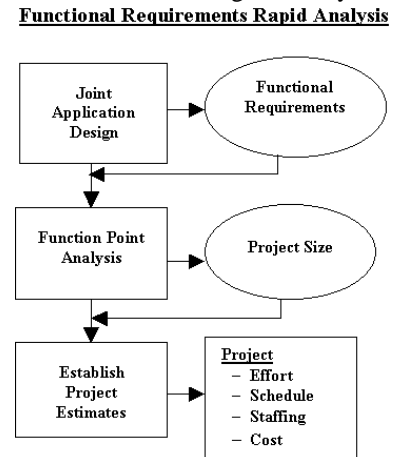
The estimate is based on the FPA size and project attribute information provided by the project team. This information is input into an estimating model. Using this process the initial project effort, staffing, and schedule estimates are developed as a natural by-product of the requirements definition effort.

The estimate is based on the FPA size and project attribute information provided by the project team. This information is input into an estimating model. Using this process the initial project effort, staffing, and schedule estimates are developed as a natural by-product of the requirements definition effort.

### Conclusion

There are several benefits to the approach described above. (1) A complete set of documented, reviewed and approved functional requirements are created. (2) Detailed Function Point Analysis documentation is developed during the sessions that establishes the size of the project and provides input into the estimating process and can be used to manage change of scope. The result of combining the Function Point Analysis and Joint Application Design methodologies produces a blueprint for the future business processes and system. This blueprint, the requirements definition and the Function Point Analysis documentation, is used to guide the organization through project estimation, and the development and implementation of the defined system.

Contact Q/P Management Group for information on utilizing this technique in your organization.



## **Key Measures for Project Management (Continued)**

**Recommendations:** Function Points should be a core measure for any software development/enhancement effort. Function Points are a well-documented standardized process supported by an internationally recognized organization, IFPUG. Their use focuses the technical team on delivering functionality that meets the users needs and, as a result, benefits the organization as a whole.

### *Effort Hours*

**Definition:** Effort Hours reflect the number hours required to complete a software activity and its associated tasks. Effort Hours should be tracked by the activity and summarized for the entire project.

**Description:** The project effort should be captured and tracked at an appropriate level of detail to satisfy the project tracking requirements. Unlike function points, there is no international body that establishes standards for effort tracking. Your organization will need to define what effort hours will be included/excluded. For example, will unpaid overtime, vacation, training, management, administration be included or excluded from the project time.

**Utilization:** Effort Hour units will be used during the project to monitor progress and report productivity metrics. It will subsequently be used to help estimate future projects using the organizations historical data. It is therefore necessary that this measure be consistently and accurately collected across projects to ensure accurate performance analysis and future project estimates.

**Recommendations:** Accurate time accounting is critical in order to evaluate the accuracy of previous estimates and is the source for future estimates based on historical data. Specific guidelines need to be established for your organization. It is critical that the scope of the activities to be included be agreed upon early in the project lifecycle.

### *Cost*

**Definition:** Project Cost refers to the funds expended to develop and deliver the software solution.

**Description:** Project Cost can be extrapolated from the effort hours, by multiplying them by an agreed upon hourly rate. There may be additional costs to consider such as hardware/software, installation, and even maintenance support. Clearly document which costs are included or excluded in your project to aid in comparative analysis.

**Utilization:** Cost is obviously an important aspect to

communicate to the project sponsor. Costs should be estimated, collected and tracked appropriately so that cost analysis can be completed to answer any potential questions.

**Recommendations:** Similar to Effort Hours be very specific of what is included or excluded from consideration. Costs pertaining to purchasing hardware/ software, maintenance, and customization need to be agreed upon early in the project.

### *Project Duration Days*

**Definition:** Project Duration Days is the number of calendar days from the start of the project at the beginning of requirements through implementation.

**Description:** As with effort hours, there are no industry standards that define project duration. Your organization needs to agree on what start/end dates to track, and what criteria should be used to determine when a milestone is met.

**Utilization:** Tracking the project duration helps the team recognize potential schedule compression early. Projects with condensed schedules (less calendar time than normal) usually have higher costs and defect rates. This is primarily due to the fact that larger than necessary project teams inhibit effective communication and coordination of project activities. Project duration is also a required measure when evaluating project progress.

**Recommendations:** Project duration should be tracked, using your own definitions for “when does the clock start” and “when does it end”. For example, when is the project considered completed? When is it “installed” at the first site? Or at all sites? If the rollout is at 1,000 sites this distinction is very important.

### *Defects*

**Definition:** A defect is a problem or error that, uncorrected, will produce unsatisfactory results. Unsatisfactory results range from cosmetic, to an inoperable systems.

**Description:** These are often fondly referred to as “bugs”, “problems” or as some software development group call them, “supplemental features”. It’s just not software that can have defects; there may be defects in the requirements, design, code or documentation. Defect information typically includes the origin (source), category, and severity level (impact).

**Utilizations:** Defects are used to help quantify the quality of the delivered product and helps identify where and when corrective action needs to take place during the project. The information is useful to help determine “is it good enough”

and forecast maintenance staffing levels. Defect-prone applications will require more support staff for maintenance support. Root cause defect analysis can also provide useful information to improve requirements definition techniques.

**Recommendations:** Evaluate your cultural readiness; the organizational needs to realize that finding defects before your customer is a good thing to do, not a bad one.

*Lines of Code*

**Definition:** Lines of code refers to executable LOC (Lines of Code), or may also be known as KLOC (Kilo Line of Code) for thousand lines of code.

**Description:** A line of code should be an “executable line of code”, not physical. Comments and blank lines should not be counted. Similarly, if an executable line is dispersed over several physical lines it should only be counted once.

**Utilization:** Useful to internal project teams for planning/tracking purposes, but has limited value as a project measure. I have never had a client’s end user care how many

lines of code it took to deliver a business function; they want the business function. Function Points are a more appropriate measure to communicate an applications size to an end user.

**Recommendations:** It may be easy to get LOC when coding using a language such as COBOL or RPG, but when coding in PowerBuilder™ or VisualBasic™ it becomes more difficult. It is difficult to estimate Lines of Code until the project is near completion; therefore care should be taken when using LOC for planning/ estimation/ tracking purposes.

**Summary**

Careful planning and analysis will permit your organization to identify its key, measures with appropriate definitions to help answer you organization questions for a particular project. When everyone understands “a mile” is “a mile” and “a kilometer” is “a kilometer” and we have specific definitions, which are understandable, clear and repeatable, then the information can be used and leveraged with confidence.

**Q/P Management Group Solutions...**

We can help your organization achieve measurement improvements with the best consulting, training and software in the industry. For more specific information, select your area(s) of interest and mail or fax this form or, for immediate results, telephone us at **781-438-2692**.

**Consulting Solutions Available for...**

- Establishing and Supporting Software Metrics Programs
- Software Development and Maintenance Benchmarking
- Software Project Estimating
- Outsourcing Evaluation and Management
- Function Point Analysis Implementation, Counting and Outsourcing
- Software Quality Assurance Programs and Methods
- Software Engineering Institute (SEI) CMM Assessments
- Project Management Methods
- Systems Development Methods
- Establishing Total Quality Management Programs
- Designing and Implementing ISO Initiatives
- Evaluating Package (COTs) Solutions
- Functional Requirements Rapid Analysis (JAD/FP)

**Training Solutions Available in...**

- Function Point Analysis (IFPUG CPM 4.1)
- Software Metrics – Definition and Reporting
- Quality Inspections Methodology Training and Implementation
- Effective Software Estimating
- Certified Function Point Specialist (CFPS) Exam – Preparatory Training
- Applying Function Point Counting Rules to Emerging Technologies
- The SEI Capability Maturity Model
- Strategies for Software Process Improvement
- Testing Information Systems

**Software Solutions with...**

- PQMPlus™ -- Software estimating and benchmarking with metrics

Name: \_\_\_\_\_ Title: \_\_\_\_\_

Company Name: \_\_\_\_\_

Phone Number: \_\_\_\_\_ Email Address: \_\_\_\_\_

Address: \_\_\_\_\_

City: \_\_\_\_\_ State/Prov: \_\_\_\_\_ Zip/Postal Code: \_\_\_\_\_

Country: \_\_\_\_\_

Q/P Management Group √ Fax: 781-438-5549 √ Email: Info@QPMG.com √ URL: www.QPMG.com