Measuring Maintenance Activities Within Development Projects

Lori Holmes and Roger Heller
Q/P Management Group, Inc.

Although function points (FPs) are a good measure of the functionality that is added, changed, or removed through a development project, they do not measure other functions that may be impacted by a specific change but are not actually changed themselves. As well, there is often project work separate from the FP measurable functionality that cannot be counted under the current International Function Point Users Group (IFPUG) 4.2 rules [1]. This article explores impact points (IPs), a measure which accounts for these issues.

Function points are an accepted industry standard method used for measuring the size of software projects and applications. They can be used in conjunction with other data for estimating as well as in conducting productivity and quality analyses. Often organizations use FPs and FP-based measures as the key metric in outsourcing contracts.

One drawback, though, is the inability of FPs to measure other functions that may be impacted by a specific change but are not actually changed themselves. Often, additional testing is required downstream of functions that will be using data from changed functions, but they themselves are not modified by the project and are not considered to be within the project scope. This increases effort and can affect productivity.

There is also work separate from the FP measurable functionality that cannot be counted under IFPUG 4.2 rules, including changes to static Web pages or populating code tables that are not related or used by the FP countable functions. The Q/P Management Group recognized, along with our outsourcing clients, that something was needed to measure the impacted functions not covered by IFPUG 4.2 FPs. It was determined that in order to ensure more accurate estimates and to provide a good foundation for productivity measures for use in outsourcing contracts, a separate measure was required. It was important for the measure to be:

- Consistent.
- Repeatable.
- Unbiased to promote good development techniques.
- Easy to apply with little effort and/or expense.

IP Concept Development

Brainstorming sessions were held to explore ideas to support the need for additional measures. The initial brainstorming activity focused on identifying situations when the FP analysis of projects resulted in zero FPs. These situations were then categorized and analyzed to ensure that they met the criteria of zero FP projects and the functionality was unrelated to functions that could be measured with IFPUG 4.2 FPs.

This type of work is often completed by a separate maintenance group and maintenance measures are used (e.g., application FPs/full-time equivalent support staff for a year). However, not all

> *"It was determined that in order to ensure more accurate estimates and to provide a good foundation for productivity measures ... a separate measure was required."*

organizations have application FP counts to utilize this measure. In these situations, this work is completed as an enhancement project and needs to be measured separately. Depending on the structure of the organization, zero FP projects can equal between 15 and 20 percent of the organization's development work.

Separate sessions were held to identify potential measures for these non-FP countable situations. One concept focused on a technical approach of counting the number of files or number of tables involved. When this concept was tested, issues arose related to consistency, objectivity, and the number of measures. It seemed as though the resulting size measure would be impacted by how the software was developed. As a result, this approach cannot be used to measure pro-

ductivity consistently across technologies or for different development methodologies.

The objective of developing the alternative measure was not to measure the technical aspects of a project; it was to measure user-recognizable functions that are impacted by a project but are not changed. In recent years, Q/P has developed a measure for analyzing functionality that needed to be tested in a project, called test points, to estimate required test cases and testing effort. Building on this work and utilizing the standard IFPUG FP methodology, Q/P developed IPs.

While developing this concept, it became clear that IPs have several benefits. IPs are:

- A single measure for all impacted functionality so it is a manageable addition to a measurement program.
- Independent of technology and implementation techniques so it can be used in all development environments.
- Based on the IFPUG FP methodology. Therefore, it does not require an extensive set of guidelines to be developed and does not require extensive training for employees already familiar with FP counting.
- A consistent measure for zero FP projects so they can be used to quantify productivity rates separately from FP-based productivity.

What Are IPs?

IPs account for functions that are impacted but not changed by a project. They follow the same concept as FPs, but focus on non-FP countable projects and functions within projects. It is imperative that IPs only be used for sizing functionality not accounted for under traditional FP analysis. The intent is not to diminish the use of FP measures with overlapping measures but rather to fill a void that exists in FP-based software measurement. Since the IP measure is intended to be complementary to FPs, it is important to account for each separately. Data related to IPs should be

kept in a separate repository from FPs; IP productivity rates should be developed and reported independently from FP productivity rates.

Once a non-countable function is identified, the IFPUG concepts are used to define the function and measure the complexity.

Projects that can be counted using FP analysis are not candidates for IP counts. IP countable items include:

- **Table Updates.** Examples are rate changes, adding products and/or services, and parameter/configuration changes.
- **Code/Text Changes.** Examples are static page updates, Web content updates, cosmetic changes, format changes, sort changes, adding or changing help, or error messaging.
- **Data Management.** Examples are data migration and database restructuring.
- **Technical.** Examples are multiple browsers and new sources of data (e.g., networks).

In all of these, the functions using the new text, updates, etc., would be identified as impacted functions and counted to derive the project's IPs.

Possible Scenarios for Considering IPs

The following are three examples of possible applications of IPs, including one scenario when IPs shouldn't be used.

Example 1: Use IPs

A project requires changes to a logical file to add additional fields necessary for calculating billing rates. Input screens and display screens of these new fields also require changes. In addition, the bill creation process logic must be changed to incorporate the new calculations. All of this functionality would be FP countable.

Once the bill is generated, it is stored in a logical file and the elementary process is completed. All changes for the project are included in the process that ultimately stores the bill. No software changes are required beyond the saving of the bill. However, functions exist that display the past bills and send the bills to other systems. These functions use the file where the bill is stored, but themselves do not require any software changes. Thus, even though they are not modified by the project, these functions are impacted because they must pass along the bill as they did before. These functions do not generate any FPs but they have been impacted by the change and need to be tested to ensure

that everything works properly.

The functions would then be counted using the IFPUG definitions to determine if they are external outputs or external inquiries (EQs), and the IFPUG complexity ratings would be applied.

Example 2: Use IPs

A request has been made to add fields to multiple Web pages that retrieve information from developer-maintained files and do not calculate, derive, or maintain any data.

Under IFPUG rules, these functions are not countable because they would be EQs with zero file types referenced (FTR), which is not allowed. Data maintained by developers are not considered as

---

"It is imperative that IPs do not overlap with the functionality that is measured by FPs. In other words, IPs should not be used to double-count or overstate the amount of work that needs to be delivered."

---

internal logical files or external interface files, so the files containing the Web page information cannot be FTRs.

IPs allow for inquiry functions (EQs) to be counted when there are zero FTRs accessed. The functional complexity would be determined based on zero FTRs and the number of data element types that are entered or displayed.

Example 3: Do Not Use IPs

A project requires new rates to be added to a table. This does not require any change to the table structure, just rows to be added. It is also necessary that logic be changed in a screen to use the new rates and to utilize different calculations when inputting data.

In this case, the table is not countable with FPs, but the screen is countable due to the logic changes to the calculations. For the specific requirement to be delivered, both changes are necessary. The table is just the development technique to

assure that the rates are available to the screen. This table change is related to the FP countable change, and any effort related to both activities should be considered together when estimating the project or calculating the productivity rate.

In this case, IPs should not be used for the table change because the change is related to a function that is FP countable.

Testing and Implementation of IPs

To implement IPs in an organization, it is important to assess and define the types of projects and situations that will use them. In addition, to use IPs for estimating and measurement, a baseline study should be conducted to quantify the current situation and establish productivity rates. Some steps to consider are to:

- Develop a list of non-FP countable situations and projects then categorize them by type and volume (e.g., rate changes, new products, etc.).
- Conduct IP counts on a representative sample of projects from the non-FP countable project list.
- Capture effort and delivery platform for non-FP countable projects included in the study to baseline their productivity rates.
- Assess the productivity rates (IPs per hour) to determine trends and any further breakdowns/measures needed.
- Develop templates to use going forward to avoid conducting IP counts on all non-FP countable projects.
  ○ If rate changes typically impact the same functions, then the same IP count would be used each time.
  ○ Based on the baseline data, one to three templates may be developed per category (e.g., low, medium, and/or high).

Conclusion

IPs can be a useful tool for organizations that have a large amount of non-FP countable projects or portions of projects. They can be used similarly to FPs in measuring productivity and quality for these projects.

IPs are not intended to replace FPs, but are meant to provide a supplement for the areas FPs do not cover. It is imperative that IPs do not overlap with the functionality that is measured by FPs. In other words, IPs should not be used to double-count or overstate the amount of work that needs to be delivered.

IPs, as a measure for non-FP countable items, are beneficial for the following reasons:

- It provides one measure for all non-FP countable projects.
- Once an impacted function is identified, guidelines for how to count are already available (IFPUG).
- The measure is not impacted by the development techniques or how things are physically implemented.
- Organizations can associate productivity rates for each appropriate segment (e.g., platform, type of change, size, etc.).
- Organizations can develop IP templates for each type of non-FP countable project to reuse on future projects of the same type.

Q/P is confident that IPs will fill the void that organizations have with measuring non-FP countable projects. Using IPs correctly will provide consistent measurement data that can be used in estimating, productivity, and quality analyses, as well as in outsourcing contract negotiations.◆

Reference

1. IFPUG Counting Practices Committee. Function Point Counting Practices Manual. Release 4.2.1, Jan. 2005.

About the Authors

Lori Holmes is a director with Q/P, specializing in software measurement, process improvement, and quality assurance. Her areas of expertise include FP analysis, software project estimation, establishing measurement programs, and software quality assurance. Holmes is recognized as an international consultant, speaker, and instructor. She focuses on helping organizations implement quality and productivity improvement programs utilizing measurement techniques. She is an experienced instructor in change management, quality inspections, measurement, and FP analysis.

Q/P Management Group, Inc.
10 Bow ST
Stoneham, MA 02180-1343
Phone: (410) 544-5781
Fax: (410) 544-5827
E-mail: lori.holmes@qpmg.com

Roger Heller is Q/P's vice president. He is a recognized consultant, instructor, and speaker who specializes in helping organizations improve software quality and productivity through measurement. Heller has assisted numerous organizations in establishing software measurement programs and quality improvement initiatives. His other areas of expertise include FP analysis (and its application techniques in emerging environments), evaluating and measuring technical architectures, software estimating, and application and technology planning.

Q/P Management Group, Inc.
10 Bow ST
Stoneham, MA 02180-1343
Phone: (941) 629-0943
Fax: (941) 629-2467
E-mail: rheller@qpmg.com