DOCUMENTING FUNCTION POINT MEASUREMENTS

By Thomas Stein CFPS, CPA, CDFM

Doc·u·men·ta·tion – The orderly presentation, organization, and communication of recorded special knowledge to produce a historical record of changes in variables.

- *Webster's computer science definition.*

- How does the Webster's definition apply to function point documentation?
- *Effective* function point documentation is a presentation of information.
- *Effective* function point documentation needs to be organized, so information can be located.
- *Effective* function point documentation needs to communicate to users of the information.
- *Effective* function point documentation needs to produce an accurate historical record of information.
- *Effective* documentation accurately records changes to function point information.

Below are some personal observations of function point documentation.

- *Effective Documentation of function point information is the tedious side of measurement, and thus, sometimes not given the attention it requires.*
- *Effective documentation accounts for what has been done and records the issues, and details related to the measurement.*
- *Effective documentation of a function point measurement effort is often overlooked, but it can be just as important as having a raw measurement number.*
- *Effective function point documentation is also overlooked because the effects of poor documentation are more often felt during the subsequent enhancement measurement to the same system.*

Why is effective documentation important to a function point measurement?

1. **To ensure *ease of use* in future measurements. The following are situations that demonstrate this point:**

- Changed system components need to be easily identified in the baseline. For example, when a report or screen is modified it needs to be easily identified in the baseline. Questions sometimes need to be answered such as, “Have additional data elements added to screens or reports increased the complexity rating from low to high? Have additional file types referenced caused complexity ratings to increase?”

- Deleted system components need to be easily identified in the baseline. When a component is removed in a project, the component needs to be deleted from the baseline. Thus, easy identification of the component within the baseline is necessary. Not being able to remove the system components or even worse, *removing the wrong system components* could significantly distort the baseline measurement.

- Determining if a component has been previously measured. Sometimes in an enhancement or modification to a system, the *requirements documentation does not provide clear details* needed to determine if the component existed in the past. Thus, the FP analyst needs to search the baseline records to determine if the system component has been previously measured. Having well documented component identifications will aid this process. If the component is mistakenly missed in the baseline records, the component probably will end up ***double counted*** in the baseline measurement.

- Reuse of component measurements. System component sizes can be reused in other measurement efforts. Time can be saved by not having to repeat questions to experts, request documentation, or analysis of information.

  For example, an enhancement could add a data element to a report. The report size could be quickly determined if it can be located in a well documented baseline that indicated the number of DET (s) associated with the report.

2. The quality of the current baseline, could impact the quality of future measurements.

Another measurement expert may want to *use your baseline* measurement to derive many project measurements in the future. As discussed earlier, they will want to pull system components from the baseline to save time. If the baseline components were not measured correctly from the beginning, then future project measurements could also contain the incorrectly measured components. For example:

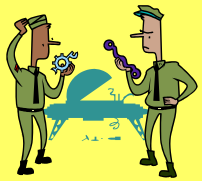
- If FTR(s) and DET(s) are incorrectly measured, future measurements will also contain incorrectly measured FTR(s) and DET(s).
- If an EQ associated with a report is incorrectly documented as an EO, then many future project measurements may show the EQ as an EO.
- If a view is incorrectly recorded as an ILF in the current baseline, then it is very likely to be incorrectly included in many future project measurements.
- If system components are not properly located in the baseline, it could lead to double counted components, incorrect complexity ratings, or unused system components not being deleted.

3. To ensure consistency in future measurements.

Consistent use of assumptions in performing estimates.
Sometimes measurements are requested of systems with *vague or few requirements*. If the system specification documentation is not detailed enough to provide a precise measurement then the assumptions used in creating the measurement should be documented. Future project measurements may use the measurement as a baseline and will need to apply the rules in a consistent manner.

An example of such a situation would be an estimated measurement of new screens. In this situation no supporting information is provided for the screens such as number of tables referenced, or fields on the screens. Thus, assumptions have to be made, such as each screen will be assumed to have 4 high functions. (one for insert, one for update, one for delete, and one for read) Each screen would then account for 24 function points.

4. To enable measurement of separate components, modules, or pieces of the application.



- Sometimes the function point measurement rules can be applied to system components or unique technologies in manners that are not easily understood. Knowing or at least *understanding the reasoning* behind the measurement could be critical to working with it in the future. Without detailed and highly explanative documentation, such measurements could become difficult if not impossible to maintain.

  For example, if someone measures a data warehouse and decides to count staging tables. Knowing how and why the staging tables were measured could prove useful to the next person who works with the measurement baseline. Knowing that the staging table is retained for multiple business cycles, and is used to send files outside the application boundary will be important. Also, knowing why staging tables were not measured could also be useful.

- The system may contain a query tool, help system, conversion application, or COTS package. These items usually need to be identified as a separate part of the measurement.
    - Time and money spent on COTS system components or a query tool will not relate to actual development time and money spent on the rest of the project measurement.
    - A conversion application may be part of a project measurement that has to be removed from the baseline at a future date.
    - Someone may want to know how large a specific part of the application is, such the help system.
- Breaking out system parts can be useful for comparison purposes. When projects have extreme metrics differences, knowing that one project contains purchased software will be important.
- A project may contain parts that are in different stages of development. The part in the requirements stage may need to be separated from the part in design, and the part in programming.

5. **So that when questions arrive months or years after the measurement there is an accurate reference.**

- Many times users of the measurement do not question the size until they see the dollars or hours spent on the project. The FP measurement may arrive months before dollar or hour data is available. Large amounts of hours and dollars matched to few FP(s) can mean lots of questions. Thus, a well documented FP measurement can help answer the questions that arrive months or years after the measurement is complete.

6. To ensure credibility of the function point measurement.

Many times measurements are demanded with few or any requirement details. Such an example would be a requirement to create an accounts payable system with no detail explained or expanded upon. In such a circumstance, some important things to document would be:

- Identifying information of the requirement document, such as a control number
- Assumptions made
- When the measurement was requested
- Explanations of the requirements given by experts

If the function point measurement is questioned at a later date the circumstances surrounding the measurement can be easily recalled and provided.

7. Version updates of IFPUG counting rules.

If the IFPUG counting rules change detailed system component information could be needed to update the baseline. Individual components may need to be deleted or have their *complexity weights revised*.

8. Future unforeseen uses.

There are needs that arise months or years after the measurement, and having a well documented measurement allows those needs to be met. The following are examples:

- Someone wants to know how many reports, tables, or screens are contained within a system.
- Someone wants use the measurement for budgeting.
- Someone wants to know the unadjusted FP size of a system.
- Someone wants to know why the measurement was so large or small relative to another measurement.

What types of information should be documented in a function point measurement?

- Measurement Type – Is the measurement a project measurement or a baseline?
- Application Name – What is the name of the application?
- Release or Project Number – What is the version number or system change number or identification?
- Measurement Date – When was the measurement complete?
- Project Manager – Person in charge of the project. Since different people can manage or be in charge of different applications at different times it is important to keep track of who was responsible at the time of measurement. This person may become an important reference in the future.
- Experts – People who provided information to derive the measurement. The name can be recorded individually with the component for future reference, or in summary with all the experts. These people may become important references in the future.

- Counting Standard – The version of the International Function Point User Group (IFPUG) rules used to derive the measurement. When the ***rules change*** significantly this information will be important for any conversion factors that may be necessary.
- Component Name - This item should be the name used within the functioning system and not the technical name used by the developer. For example identify a report by title, and not the technical name used by a developer to call the report program. However, it is useful to save the technical name with the component so that the component can be identified if only the technical name is available.
- Component Type – External Output (EO), External Input (EI), External Inquiry (EQ), Internal Logical File (ILF), External Interface File (EIF).
- Component Description – Examples include screens, reports, tables, scripts, or interface files. Having this information can be helpful in identifying the component within the measurement, and determining that all the component types have been measured. Also, questions often arise with regard to how many of a specific component type exist within the system.

- Record Element Types (RET), File Types Referenced (FTR), or Data Element Types (DET) – Number of data element types, file types used, or record element types used to determine the component complexity, and value.
- General System Characteristics (GSC)s – Information describing how the characteristics were derived is helpful. For example, maybe a sentence explaining why a characteristic was valued at a 4 and not a 5 would be useful in future measurements.
- Added, Changed, Deleted, or Conversion – Is the component new, modified, removed or converted for the project? Baseline measurements only have added components. For conversions that will exist for a lengthy period of time, it may be useful to include them in the baseline. However, differentiate their measurement value somehow in total size measurement. Thus, when they are eventually discarded, and the baseline changes size changes it will be clear why.
- Complexity rating – Is the component rated low, average, or high? This item may appear redundant because the numeric value of the component may also be recorded. However, future imports or exports of measurement data from applications repositories may make this information more useful.

- Narrative – This space is used to document the nature of the measurement, relevant facts relating to the measurement, or important issues pertaining to the measurement. ***Document unusual items here*** such as new technologies, or unique ways of applying function point rules. Also, it could be important to notate the ***nature of the resources*** used to derive the measurement in this space. For example, that fact that vague requirements were the only resource supplied to determine the measurement could be notated in this place. Finally, conversion information could be included in the narrative space. If the baseline will include a large number of conversion components for an indefinite period of time, a narrative space would be a good place to make note of this fact.
- Component identification - Usually a number or value used to specify an individual piece of the application or project that is analyzed for the function point methodology. This type of information is most often system generated in database applications.

- Unusual Components – Some components fall outside of the usual patterns observed through measurement experience. The unique characteristics of these components needs to be recorded with the specific components. Examples of such components:

  - Menus that update tables. Menus typically have no measurement value. Noting that the component is a menu along with its measurement characteristics could be important

  - Screens that are linked to other screens. It commonly needs to be determined if linked screens are one or separate system components. If screens are considered one component, having a notation of the relationship in the baseline records will avoid confusion in future measurements.

  - Views or temp files that are given a measurement value. Such components are typically not given a measurement value. Noting why they were included in the measurement could be useful.

- Extraordinary Information - items that are unusual to the measurement. Sometimes applications contain features or technological factors that make the application unique to the environment or organization. Extraordinary items may be a subset or separately recorded and reported within the measurement. The could be placed in a separate location within the measurement, so that they can be worked with independently if needed. Examples of such items:

  - Query Tools – Usually has a large amount of functionality with little effort. Usually has custom made canned queries using the tool, and an infinite number of user created ad hoc queries.

  - Robohelp – A help application tool that is customized to fit the needs of the application it supports. It should not be measured like most help functions that come with the tool used to create the application. Creating a Robohelp application requires different programming skills, and thus should be reported separately within the measurement.

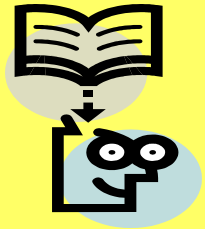
  - Estimated vs. Actual Measurement – Estimated measurements are performed with little documentation, or supporting information. They are performed quickly using the judgment of the measurement expert. Consequently, the reliability of the measurement needs to be understood by the users of the information.

- Warehouses – May require creative application of the IFPUG rules, and thus separating the warehouse part of the measurement effort can aid greatly in understanding the measurement. A data warehouse may contain a large amount of measured functionality relative to other parts of the measured project. The functionality may also have a relatively small amount of hours and costs associated with it. Understanding how the measurement rules were applied, and the nature of the programming effort used to create the warehouse could be helpful in understanding the measurement.

- Customized Off The Shelf Software (COTS) packages – COTS packages are purchased. The cost of a COTS package will not equate with the cost of customized software that is developed within the organization. Each software type needs to be reported separately in order to analyze or understand the issues surrounding the software.

- Programming Languages – Programming languages should be documented in the measurement when they are unusual to the organization. Different languages require different skills, knowledge, and abilities that need to be considered when evaluating the measurement.

Points to keep in mind to improve documentation:



- Avoid abbreviations – you may understand what the abbreviation means, but the next person may not be able to figure it out.
- Acronyms – somewhere in the measurement ***clearly define*** the acronym, if possible do it in multiple places for large measurements.
- Technical Names vs. Functional Names – If possible identify both names with the specific component. Documenting this information is important because experts supply information needed to derive the FP measurement using both names.
- Record non-measurable items such as menus, temporary files, or display screens. You will probably see them again, possibly many times.
- Keep in mind that your measurement today, could impact many other measurements in the future.
- Keep in mind that someone besides yourself, may have to work with the measurement you are creating today in the future.
- If a GSC rating is debatable, notate the reasoning behind the score.

DOCUMENTING FUNCTION POINT MEASUREMENTS

Doc·u·men·ta·tion – The orderly presentation, organization, and communication of recorded special knowledge to produce a historical record of changes in variables.

Webster's computer science definition.

What can we infer from the definition?

- Function point documentation is a presentation.
- Function point documentation needs to be organized.
- Function point documentation needs to communicate to users of the information.
- Function point documentation needs to produce a historical record.
- Documentation records changes to function point information.

This paper is focused on what makes presenting, organizing, communicating and recording function point information effective and successful.

Below are some personal observations of function point documentation.

- *Effective Documentation of function point information is the tedious side of measurement, and thus, sometimes not given the attention it requires.*

- *Effective documentation accounts for what has been done and records the issues, and details related to the measurement.*

- *Effective documentation of a function point measurement effort is often overlooked, but it can be just as important as having a raw measurement number.*

Why is effective documentation important to a function point measurement?

1. **To ensure ease of use in future measurements. The following are situations that demonstrate this point:**

- Changed system components need to be easily identified in the baseline. For example, when a report or screen is modified it needs to be easily identified in the baseline. Questions sometimes need to be answered such as, "Have additional data elements added to screens or reports increased the complexity rating from low to high? Have additional file types referenced caused complexity ratings to increase?"

- Deleted system components need to be easily identified in the baseline. When a component is removed in a project, the component needs to be deleted from the

baseline. Thus, easy identification of the component within the baseline is necessary. Not being able to remove the system components or even worse, removing the wrong system components could significantly distort the baseline measurement.

- Determining if a component has been previously measured. Sometimes in an enhancement or modification to a system, the requirements documentation does not provide clear details needed to determine if the component existed in the past. Thus, the FP analyst needs to search the baseline records to determine if the system component has been previously measured. Having well documented component identifications will aid this process. If the component is mistakenly missed in the baseline records, the component probably will end up double counted in the baseline measurement.

- Reuse of component measurements. System component sizes can be reused in other measurement efforts. Time can be saved by not having to repeat questions to experts, request documentation, or analysis of information.

  For example, an enhancement could add a data element to a report. The report size could be quickly determined if it can be located in a well documented baseline that indicated the number of DET (s) associated with the report.

2. The quality of the current baseline, could impact the quality of future measurements.

Another measurement expert may want to use your baseline measurement to derive many project measurements in the future. As discussed earlier, they will want to pull system components from the baseline to save time. If the baseline components were not measured correctly from the beginning, then future project measurements could also contain the incorrectly measured components. For example:

If FTR(s) and DET(s) are incorrectly measured, future measurements will also contain incorrectly measured FTR(s) and DET(s).

If an EQ associated with a report is incorrectly documented as an EO, then many future project measurements may show the EQ as an EO.

If a view is incorrectly recorded as an ILF in the current baseline, then it is very likely to be incorrectly included in many future project measurements.

If system components are not properly located in the baseline, it could lead to double counted components in the baseline, and incorrect metrics with regard to added functionality in project measurements.

3. **To ensure consistency in future measurements.**

- Consistent use of assumptions in performing estimates. Sometimes measurements are requested of systems with vague or few requirements. If the system specification documentation is not detailed enough to provide a precise measurement then the assumptions used in creating the measurement should be documented. Future project measurements may use the measurement as a baseline and will need to apply the rules in a consistent manner.

  An example of such a situation would be an estimated measurement of new screens. In this situation no supporting information is provided for the screens such as number of tables referenced, or fields on the screens. Thus, assumptions have to be made, such as each screen will be assumed to have 4 high functions. (one for insert, one for update, one for delete, and one for read) Each screen would then account for 24 function points.

4. **To enable measurement of separate components, modules, or pieces of the application.**

- Sometimes the function point measurement rules can be applied to system components or unique technologies in manners that are not easily understood. Knowing or at least understanding the reasoning behind the measurement could be critical to working with it in the future. Without detailed and highly explanative documentation, such measurements could become difficult if not impossible to maintain.

  For example, if someone measures a data warehouse and decides to count staging tables. Knowing how and why the staging tables were measured could prove useful to the next person who works with the measurement baseline. Knowing that the staging table is retained for multiple business cycles, and is used to send files outside the application boundary will be important. Also, knowing why staging tables were not measured could also be useful.

- The system may contain a query tool, help system, conversion application, or COTS package. These items usually need to be identified as a separate part of the measurement.

  - Time and money spent on COTS system components or a query tool will not relate to actual development time and money spent on the rest of the project measurement.
  - A conversion application may be part of a project measurement that has to be removed from the baseline at a future date.
  - Someone may want to know how large a specific part of the application is, such the help system.

- Breaking out system parts can be useful for comparison purposes. When projects have extreme metrics differences, knowing that one project contains purchased software will be important.

5. **So that when questions arrive months or years after the measurement there is an accurate reference.**

   Many times users of the measurement do not question the size until they see the dollars or hours spent on the project. The FP measurement may arrive months before dollar or hour data is available. Large amounts of hours and dollars matched to few FP(s) can mean lots of questions. Thus, a well documented FP measurement can help answer the questions that arrive months or years after the measurement is complete.

6. **To ensure credibility of the function point measurement.**

   Many times measurements are demanded with few or any requirement details. Such an example would be a requirement to create an accounts payable system with no detail explained or expanded upon. In such a circumstance, some important things to document would be:

   - Identifying information of the requirement document, such as a control number
   - Assumptions made
   - When the measurement was requested
   - Explanations of the requirements given by experts

   If the function point measurement is questioned at a later date the circumstances surrounding the measurement can be easily recalled and provided.

7. **Version updates of IFPUG counting rules.**

   If the IFPUG counting rules change detailed system component information could be needed to update the baseline. Individual components may need to be deleted or have their complexity weights revised.

8. **Future unforeseen uses.**

   There are needs that arise months or years after the measurement, and having a well documented measurement allows those needs to be met. The following are examples:

   a. Someone wants to know how many reports, tables, or screens are contained within a system.
   b. Someone wants use the measurement for budgeting.
   c. Someone wants to know the unadjusted FP size of a system.
   d. Someone wants to know why the measurement was so large or small relative to another measurement.

What types of information should be documented in a function point measurement?

- Measurement Type – Is the measurement a project measurement or a baseline?
- Application Name – What is the name of the application?
- Release or Project Number – What is the version number or system change number or identification?
- Measurement Date – When was the measurement complete?
- Project Manager – Person in charge of the project. Since different people can manage or be in charge of different applications at different times it is important to keep track of who was responsible at the time of measurement. This person may become an important reference in the future.
- Experts – People who provided information to derive the measurement. The name can be recorded individually with the component for future reference, or in summary with all the experts. These people may become important references in the future.
- Counting Standard – The version of the International Function Point User Group (IFPUG) rules used to derive the measurement. When the rules change significantly this information will be important for any conversion factors that may be necessary.
- Component Name - This item should be the name used within the functioning system and not the technical name used by the developer. For example identify a report by title, and not the technical name used by a developer to call the report program. However, it is useful to save the technical name with the component so that the component can be identified if only the technical name is available.
- Component Type – External Output (EO), External Input (EI), External Inquiry (EQ), Internal Logical File (ILF), External Interface File (EIF).
- Component Description – Examples include screens, reports, tables, scripts, or interface files. Having this information can be helpful in identifying the component within the measurement, and determining that all the component types have been measured. Also, questions often arise with regard to how many of a specific component type exist within the system.
- Record Element Types (RET), File Types Referenced (FTR), or Data Element Types (DET) – Number of data element types, file types used, or record element types used to determine the component complexity, and value.
- General System Characteristics (GSC)s – Information describing how the characteristics were derived is helpful. For example, maybe a sentence explaining why a characteristic was valued at a 4 and not a 5 would be useful in future measurements.
- Added, Changed, Deleted, or Conversion – Is the component new, modified, removed or converted for the project? Baseline measurements only have added components. For conversions that will exist for a lengthy period of time, it may be useful to include them in the baseline. However, differentiate their measurement value somehow in total size measurement. Thus, when they are eventually discarded, and the baseline changes size changes it will be clear why.
- Complexity rating – Is the component rated low, average, or high? This item may appear redundant because the numeric value of the component may also be recorded. However, future imports or exports of measurement data from applications repositories may make this information more useful.

- Narrative – This space is used to document the nature of the measurement, relevant facts relating to the measurement, or important issues pertaining to the measurement. Document unusual items here such as new technologies, or unique ways of applying function point rules. Also, it could be important to notate the nature of the resources used to derive the measurement in this space. For example, that fact that vague requirements were the only resource supplied to determine the measurement could be notated in this place. Finally, conversion information could be included in the narrative space. If the baseline will include a large number of conversion components for an indefinite period of time, a narrative space would be a good place to make note of this fact.
- Component identification - Usually a number or value used to specify an individual piece of the application or project that is analyzed for the function point methodology. This type of information is most often system generated in database applications.
- Unusual Components – Some components fall outside of the usual patterns observed through measurement experience. The unique characteristics of these components needs to be recorded with the specific components. Examples of such components:
    - Menus that update tables. Menus typically have no measurement value. Noting that the component is a menu along with its measurement characteristics could be important
    - Screens that are linked to other screens. It commonly needs to be determined if linked screens are one or separate system components. If screens are considered one component, having a notation of the relationship in the baseline records will avoid confusion in future measurements.
    - Views or temp files that are given a measurement value. Such components are typically not given a measurement value. Noting why they were included in the measurement could be useful.
- Extraordinary Information - items that are unusual to the measurement. Sometimes applications contain features or technological factors that make the application unique to the environment or organization. Extraordinary items may be a subset or separately recorded and reported within the measurement. The could be placed in a separate location within the measurement, so that they can be worked with independently if needed. Examples of such items:
    - Query Tools – Usually has a large amount of functionality with little effort. Usually has custom made canned queries using the tool, and an infinite number of user created ad hoc queries.
    - Robohelp – A help application tool that is customized to fit the needs of the application it supports. It should not be measured like most help functions that come with the tool used to create the application. Creating a Robohelp application requires different programming skills, and thus should be reported separately within the measurement.
    - Estimated vs. Actual Measurement – Estimated measurements are performed with little documentation, or supporting information. They are

performed quickly using the judgment of the measurement expert. Consequently, the reliability of the measurement needs to be understood by the users of the information.

- Warehouses – May require creative application of the IFPUG rules, and thus separating the warehouse part of the measurement effort can aid greatly in understanding the measurement. A data warehouse may contain a large amount of measured functionality relative to other parts of the measured project. The functionality may also have a relatively small amount of hours and costs associated with it. Understanding how the measurement rules were applied, and the nature of the programming effort used to create the warehouse could be helpful in understanding the measurement.
- Customized Off The Shelf Software (COTS) packages – COTS packages are purchased. The cost of a COTS package will not equate with the cost of customized software that is developed within the organization. Each software type needs to be reported separately in order to analyze or understand the issues surrounding the software.
- Programming Languages – Programming languages should be documented in the measurement when they are unusual to the organization. Different languages require different skills, knowledge, and abilities that need to be considered when evaluating the measurement.

Points to keep in mind to improve documentation:

- Avoid abbreviations – you may understand what the abbreviation means, but the next person may not be able to figure it out.
- Acronyms – somewhere in the measurement clearly define the acronym, if possible do it in multiple places for large measurements.
- Technical Names vs. Functional Names – If possible identify both names with the specific component. Documenting this information is important because experts supply information needed to derive the FP measurement using both names.
- Record non-measurable items such as menus, temporary files, or display screens. Also, record why the item is a non-measurable item. You will probably see them again, possibly many times.
- Keep in mind that your measurement today, could impact many other measurements in the future.
- Keep in mind that someone besides yourself, may have to work with the measurement you are creating today in the future.
- If a GSC rating is debatable, notate the reasoning behind the score.