Abstract

Although Functional Size Measurement (FSM) methods have gone a long way, they still provide challenges for software managers. A major challenge is related to the comparison of the well established FSM methods. This paper identifies the conceptual similarities and differences among MkII FPA, COSMIC FFP and IFPUG FPA methods based on a case study, which involves implementation of the methods to measure the functional size of a military inventory management project integrated with a document management system.

1. Introduction

Since the description of the original method in 1979 by Albrecht [4], variations of the functional size metrics and methods have been developed to improve the preceding ones or to extend their applicability to different functional domains [41].

Although the Functional Size Measurement (FSM) methods yield to roughly similar results on average, they have not been designed to reach similar sizes. These methods measure the software using their own concepts and measurement processes and they utilize different metrics for measurement. Therefore, a piece of software has several functional sizes when measured by different methods [16].

These differences result in a number of difficulties in practice. Software practitioners need to compare different methods and decide which to use for measuring the functional size of a software product. Moreover, they often need to compare functional sizes of software products measured by different methods.

In this study, we identify the similarities and differences between FSM methods by mapping the concepts used by the FSM methods and exploring the rules of the methods in detail. Our observations are based on a conceptual comparison in addition to a case study.

The case study involves the implementation of three well-known FSM methods for the measurement of the functional size of a software system; the International Function Point Users Group (IFPUG) Function Point Analysis (FPA) [21], MkII FPA [26] and the Common Software Measurement International Consortium (COSMIC) Full Function Points (FFP) [24]. The results of the case study enabled us to depict the conceptual similarities and differences among FSM methods and to identify the issues concerning further research studies.

This paper is organized in six sections. The related research on software FSM methods are briefly summarized in the second section. The mapping of the concepts and rules used by different FSM methods are discussed in the third section. The case study and the results are presented in the fourth section. The identified issues are discussed in the fifth section. Finally, conclusions are given in the last section.

2. Related Research

There have been several significant attempts to measure functional size. Initially, in 1979, Albrecht designed “Function Points” (FP) metric and Function Points Analysis (FPA) method for measuring software size [4]. This is based on the idea of measuring the amount of functionality delivered to the users in terms of FP.

Within the following years, the original FPA method has been improved [5][6][14][15][17][18][19][20][21]. Several authors have suggested new counting techniques for measuring the amount of functionality [7][35][40][43][44][45] or to extend the applicability of FSM methods to different
functional domains in addition to business application software [1][3][11][32][34][36][42].

In 1996, ISO initiated a workgroup to clarify the conceptual basis and to establish the common principles of FSM methods. In 1998, the first part of ISO/IEC 14143 - Software Measurement - Functional Size Measurement was published. This standard defines the fundamental concepts of FSM such as Functional User Requirements (FURs), Functional Size, Base Functional Component (BFC), BFC Type and the FSM requirements that should be met by a candidate FSM method in order to be conformant. The remaining parts were published during the following years [23][29][30][31][28].

As of today, four methods are certified by ISO as an international FSM standard; MkII FPA [26], IFPUG FPA [25], COSMIC FFP [24] and NESMA FSM [27].

Although all FSM methods measure the functional size, each method defines different metrics and rules for measuring the functional size [12].

In the literature there exist a number of studies on the evaluation and comparison of the FSM methods. In [33], Lother and Dumke evaluate FSM methods with respect to a number of criteria and discussed the issues of FSM such as the suitability of the methods for functional domains and the impact of new technologies. They conclude that further investigation and research is necessary to deal with the convertibility among different FSM methods, the mapping between Unified Modeling Language and FP, measuring the domain of algorithmic/scientific software, measurement of functional reuse and automating FSM process.

Fetcke et al [10] proposed a model as a generalized representation of different FSM methods emphasizing on their commonalities and differences. They concluded that the formalized approach might be used as a basis for the representation of experience data from previous projects and it might allow the automation of functional size measurement.

In another study [8], Demirors and Gencel evaluated three estimation methods - MkII FPA, Jones Very Early Size Predictor and Early FPA - by applying them to a case project at different phases early in the life cycle. They found out that although these methods can be used for early estimation, each of them have their restrictions. Jones Very Early Size Predictor is rather inaccurate for serious size estimation purposes, but it has the virtue of being usable for small projects before any other known form of sizing is possible. MkII FPA can be used with some assumptions and with the support of expert opinion methods in earlier phases. EFPA can achieve more reliable results if the assignments of the method are improved on the basis of gathering more data on other projects and definition of the requirements is formalized in order to decrease its subjectivity.

In [38] Rule discussed the similarities and differences between IFPUG FPA and MkII FPA in his study to assist practitioners in understanding the common principles and objectives of these methods.

In [37], Rollo discussed the problems associated with sizing web applications and evaluated IFPUG FPA, MkII FPA and COSMIC FFP by applying them to a sample e-commerce application. He found out that IFPUG FPA method rules necessitates amending to work properly for sizing Web applications and COSMIC FFP works for these kinds of projects.

In [13], Gencel et al. presented the results obtained by applying MkII FPA and COSMIC FFP to a real-time software system. In this study, the similarities and differences among the measurement processes of these methods are discussed. Moreover, difficulties were faced during measurement such as identification of the primary components of FURs when the Software Requirements Specification (SRS) document is not organized in a manner suitable for FSM and measurement of the functional size of algorithmic operations and manipulations.

3. The Similarities and Differences Between FSM Methods

This paper compares the FSM methods with respect to two viewpoints: a) FSM measurement processes and b) the concepts used by FSM methods. In Figure 1, the similarities and differences of the FSM processes for IFPUG FPA, MkII FPA and COSMIC FFP methods are presented.

---

1 FURs: “a sub-set of the user requirements. The FURs represent the user practices and procedures that the software must perform to fulfill the users' needs”.
2 Functional Size: “a size of the software derived by quantifying the FUR”.
3 BFC: “an elementary unit of FUR defined by and used by an FSM Method for measurement purposes”.
4 BFC Type: “A defined category of BFC. A BFC is classified as one and only one BFC Type”.
Figure 1 FSM Process of IFPUG FPA, Mk II FPA and COSMIC FFP

The software entity measured by all FSM methods is the set of FURs. In practice, FUR may exist in the form of a Software Requirements Specification (SRS) document, or they have to be derived from other software engineering artifacts such as architecture, design or even from the artifacts installed on the computer system after it has been implemented [42].

FSM methods define two steps for the abstraction of software which represents the items deemed relevant for functional size [10].

The first step of the FSM process is to extract the FUR from these artifacts and to express them in the form of the measurement model of the FSM method to measure functional size.

The FURs are functionally decomposed into ‘Transactions’, which are sequences of input, processing and output, not necessarily in that order, triggered by events outside the software [31]. From these Transactions, BFCs are identified and then each of these is categorized into BFC Types and the attributes relevant for obtaining the base counts are identified.

The second step of abstraction involves the actual measurement, i.e. the functional size of each BFC is measured by applying a measurement function to the BFC Types and the related attributes. There forward, the results are summed up to compute the overall size of the software system.

The BFC of MkII FPA is the “Logical Transaction (LT)”, which has also one type. In order to assign numeric values to each BFC, some of the FSM methods identify and evaluate the constituent parts from which the BFC types are composed. Mk II FPA defines three constituent parts for a LT: input, process and output components. The unit of measure of MkII FPA is MkII FP.

The BFCs of IFPUG FPA are the “Elementary Processes”. These are further classified as the Transactional Function (TF) Types and Data Function (DF) Types. A TF can be of type: External Input (EI), External Output (EO), or External Inquiry (EQ), whereas a DF may be an External Interface File (EIF) or an Internal Logical File (ILF). The unit of measure of IFPUG FPA is IFPUG FP.

COSMIC FFP method introduces another concept; “Elementary Components”. Each FUR is decomposed into its elementary components, called “Functional Processes”. Each of these Functional Processes is assumed to comprise a set of sub-processes, called Data Movement Types, which are declared as the BFCs of this method. There are four kinds of Data Movement Types: Entry, Exit, Read, and Write. Each of these is defined as a BFC Type. The unit of measure of COSMIC FFP is COSMIC Functional Size Unit (Cfsu).

Although each FSM method has its own metrics and measurement process rules for measuring the functional size, the FSM methods share some common concepts and uses related attributes in their measurement processes [9].

The concepts and counting rules defined in different FSM methods do not have a one-to-one mapping. Still, most of the concepts and rules utilize the same information. Based on the similarities and differences identified and the concepts defined in ISO/IEC 14143-1 [22], we determined the common concepts of these FSM methods and map the terminologies used by each of them to each other. Figure 2 provides the conceptual mapping of IFPUG FPA, MkII FPA and COSMIC FFP methods.

4. Case Study

A case study is conducted with the aim of exploring the similarities and differences between FSM methods and shed light to the improvement opportunities related to convertibility among different FSM methods. In this case study, the functional sizes were measured by IFPUG FPA, Mark II FPA and COSMIC FFP methods. The details of the case study are presented in the following paragraphs.

4.1. Description of the Case Project

The case project is a web based, military inventory management project integrated with a document management system. It is a data-strong system, which also involves a number of algorithmic operations. The Functional Domain of the case project is determined to be ‘Information System’ by using the CHAR Method defined in [31].

The project started in October 2004 and was completed in December 2005. The efforts utilized for this project are given in Table 1.

The project team consisted of 6 people, 1 project manager, 1 part time test engineer, 2 senior and 2 part time software engineers.

The types of software products and programming language(s) used for the project were; Internal Development Framework and Java as programming languages, IBM WebSphere Application Developer as development environment, Rational Rose as Analysis and Design tool, Oracle 9i as Database Management System, Tomcat as Application Server.
Table 1 Development effort for case project

<table>
<thead>
<tr>
<th>Software Development Life Cycle Phase</th>
<th>Project-3 Effort (person-hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>6,308.00</td>
</tr>
<tr>
<td>Software Requirements Analysis</td>
<td>911.00</td>
</tr>
<tr>
<td>Software Design</td>
<td>698.00</td>
</tr>
<tr>
<td>Software Coding &amp; Unit Testing</td>
<td>3,111.00</td>
</tr>
<tr>
<td>Testing</td>
<td>1,588.00</td>
</tr>
<tr>
<td>Management</td>
<td>225.00</td>
</tr>
<tr>
<td>Testing</td>
<td>240.00</td>
</tr>
<tr>
<td>Supporting</td>
<td>720.00</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>7,493.00</strong></td>
</tr>
</tbody>
</table>

The project documents were prepared in compliance with the organizational quality handbook. The company uses a Software Requirements Specification (SRS) standard, which was developed by the company itself. Size measurements by all three FSM methods were performed using the SRS document of the project, which involves 123 Use Cases.

4.2. Case Study Conduct

Two persons performed the functional size measurement together during the conventional measurement. One of them works for the development organization and was involved in this project. The other is one of the authors of this paper. Both are experienced in using the methods.

The functional size measurement of the case project was performed by IFPUG FPA, Mark II FPA and COSMIC FFP, successively (See Table 2, Table 3 and Table 4).
Table 2 Case Project - IFPUG FPA Size Measurement Details

<table>
<thead>
<tr>
<th>Number of Elementary Processes</th>
<th>ILFs</th>
<th>EIFs</th>
<th>EIs</th>
<th>EOs</th>
<th>EQs</th>
<th>Functional Size (IFPUG FP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>294</td>
<td>0</td>
<td>262</td>
<td>343</td>
<td>26</td>
<td>925</td>
</tr>
</tbody>
</table>

Table 3 Case Project - Mark II FPA Size Measurement Details

<table>
<thead>
<tr>
<th>Number of Logical Transactions</th>
<th>Number of Input DETs</th>
<th>Number of Output DETs</th>
<th>Number of Data Entity Types Referenced</th>
<th>Functional Size (Mark II FP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>559</td>
<td>1,679</td>
<td>343</td>
<td>1,330.14</td>
</tr>
</tbody>
</table>

Table 4 Case Project COSMIC FFP Size Measurement Details

<table>
<thead>
<tr>
<th>Number of Functional Processes</th>
<th>Number of Entries</th>
<th>Number of Exits</th>
<th>Number of Reads</th>
<th>Number of Writes</th>
<th>Functional Size (Cfsu)</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>206</td>
<td>364</td>
<td>334</td>
<td>156</td>
<td>1,060.0</td>
</tr>
</tbody>
</table>

4.3. Case Study Results and Discussion

In order to compare the measurement results of the case study, we consider the differences and similarities between IFPUG FPA, MkII FPA and COSMIC FFP discussed in Section 3 as well as the existing conversion formulae among methods.

Comparison of MkII FPA-IFPUG FPA sizes. In [39], a formula to convert MkII FPA and IFPUG FPA sizes to one another is defined by mapping the BFC types of IFPUG FPA to MkII FPA.

In order to compare IFPUG FPA and MkII FPA measurement results, MkII FPA size measure was converted to IFPUG FPA size measure by utilizing the formula defined in [39]. For the product sizes that are below 1,500 IFPUG FP’s or 2,500 MkII FP, the ratio of the functional sizes is given by the following formula:

\[
\text{MkII}_\text{FP} = 0.9 \times \text{IFPUG}_\text{FP} + 0.0005 \times \text{IFPUG}_\text{FP}^2
\]  

For the product sizes that are above 1,500 IFPUG FP’s or 2,500 MkII FP, the ratio of the functional sizes is given by the following formula:

\[
\frac{\text{MkII}_\text{FP}}{\text{IFPUG}_\text{FP}} = 0.16 \times \frac{\text{No of Entity References}}{\text{No of Entity Types}}
\]

Since the size of the case project is below 2,500 MkII FP, we used Formula 1 and converted the functional size in IFPUG FP to MkII FP. The obtained size in MkII FP is found as 1,260.31 MkII FP. This indicates that there is an error of about -5.25 % with respect to the measured size in MkII FP.

Comparison of IFPUG FPA-COSMIC FFP sizes. In [2], a survey of previous convertibility studies is presented and convertibility analyses from IFPUG FPA to COSMIC FFP size are discussed. However, since the formula varies among the organizations, a unique conversion formula could not be obtained from the datasets used.

In [42], it is stated that an ‘average conversion’ formula to convert COSMIC FFP size to an IFPUG FPA size would be seriously misleading. Three main factors might give rise to divergences between IFPUG FPA and COSMIC FFP sizes. The first one is that if the software being measured has a high proportion of files which are not much referenced by the processes, measurements made by IFPUG FPA scale tend to result in higher sizes than those made by the COSMIC FFP scale.

The second factor arises from allocating size to each BFC whether within limited ranges or with no upper limit. In IFPUG FPA, an External Input can have a size in the range of 3 to 6 FP. In COSMIC FFP, there is no upper limit to the size of a functional process. If the number of sub-processes in a Functional Process is high, the functional size obtained by COSMIC FFP would be much higher.

The third factor is related to the granularity level of the BFC attributes counted by IFPUG FPA and COSMIC FFP. IFPUG FPA assigns a complexity weight to each BFC. This weight depends on the
predetermined interval values of DETs. Therefore, the exact number of DETs is not required in IFPUG FPA. Determining the interval in which the number of DETs falls is sufficient. By IFPUG FPA, the functional size of two BFCs might be measured as being the same although the number of DETs is different. COSMIC FFP does not take into account the number of DETs manipulated by each sub-process. The COSMIC FFP unit of measurement, 1 Cfsu, has been fixed at the level of one data movement [24].

In the case study, the functional size obtained by COSMIC FFP is 12.74 % greater than the one obtained by IFPUG FPA. This shows that the numbers of sub-processes are high for this subsystem.

Comparison of MkII FPA-COSMIC FFP sizes. In the literature, no formula has been defined for the conversion of MkII FPA and COSMIC FFP sizes to one another. Therefore, we compared the results of both methods according to the base counts in order to depict what kind of factors might give rise to obtaining different functional sizes.

In MkII FPA, the size of the processing component of a LT is defined to be proportional to the number of referenced Data Entity Types. A Data Entity Reference in MkII FPA is generally equivalent to a data group Read or Write in COSMIC FFP. Therefore, the sizes of the processing component are roughly equivalent on both scales [24].

The first distinction between these two methods is related to the assumptions of the methods for measuring the size of the processing component. MkII FPA assumes that each LT must have at least one input DET, must make one reference to a Data Entity Type and must have one output DET as a minimum. On the other hand, COSMIC FFP principles states that “A Functional Process comprises at least two data movements, an entry plus either an exit or a write” [24].

By MkII FPA, the number of references to Data Entity Types was found to be 343 for the case project. By COSMIC FFP, the total numbers of data groups that are read or written were found to be 490 in Project-3. In all three cases, the base counts related to the processing components are higher in COSMIC FFP than in MkII FPA. This points out that there exist Data Entity Types which are both read and written in a Transaction. These are counted only once in MkII FPA, whereas they are counted separately as Entries and Exits in COSMIC FFP.

By MkII FPA, the functional size of the processing component of Project-3 is 364.9. By COSMIC FFP, the functional size of the processing component is 490 Cfsu. Although a higher functional size by COSMIC FFP is expected, the weight factor used in MkII FPA changes the result. That is, when calculating the functional size of the processing component, we multiply the number of references by 1.66 in MkII FPA, whereas there is no weight factor in COSMIC FFP.

The second distinction between MkII FPA and COSMIC FFP is the difference in the granularity level of the BFC attributes counted by each. COSMIC FFP method estimates functional size at a higher level of granularity than MkII FPA. The COSMIC FFP unit of measurement, 1 Cfsu, has been fixed at the level of one data movement. On the other hand, in MkII FPA method, the size of the input and output components of a LT is defined to be proportional to the number of DETs in the input and output components.

For the case project, in MkII FPA measurement, the number of input DETs is 559 and the number of output DETs is 1,679. By COSMIC FFP, the number of Entries is 206 and the number of Exits is 364.

The third distinction between MkII FPA and COSMIC FFP is related to the relationship between the functional sizes of different BFC Types and constituent parts of BFC Types. MkII FPA makes use of weight factors which are calibrated to industry-average relative effort to develop each component. This enables these three kinds of functionality to be combined into a single value for a functional size. On the other hand, COSMIC FFP assumes that the average number of DETs per data movement does not vary much across the four BFC Types, i.e. Entry, Exit, Read, Write. Thus, the contribution of each to functional size is assumed to be the same.

In the case study, the distinctions of MkII FPA and COSMIC FFP discussed above resulted in higher size values by MkII FPA than values by COSMIC FFP, with a difference of 20 %.

5. Conclusion

Among the other software size measurement methods, the FSM methods and metrics have become widely used. However, these methods measure functional size utilizing different metrics and rules.

In order to be able to evaluate and compare different FSM methods, the differences and similarities among them should be known. Moreover, when the functional sizes of past and current projects obtained by different methods need to be compared, we need to convert the functional size obtained by one method to another.

In this study, we compared the measurement processes of FSM methods and identified the
conceptual differences between them. We also conducted a case study to evaluate the effects of these differences on the measurement results and to bring into light the improvement opportunities related to convertibility of different functional sizes.

The results of the case study showed that for projects of size below 2,500 MkII FP or 1,500 IFPUG FP, we can convert functional sizes obtained by IFPUG FPA and MkII FPA to one another by using a conversion formula (1) defined by Symons [39]. However, we should consider the assumptions of the formula when applying to different cases. Formula (2) should be evaluated for larger projects.

To convert functional sizes obtained by IFPUG FPA and COSMIC FFP, it is stated that an ‘average conversion’ formula to convert COSMIC FFP size to an IFPUG FPA size would be grossly misleading [42]. However, some form of conversion might be possible by making some assumptions if the details of measurements are available in a projects dataset.

For the convertibility between MkII FPA and COSMIC FFP, the designers of COSMIC FFP stated that an ‘average conversion’ formula would result in a software being under-sized or over-sized [42]. We agree with this statement by adding that a more precise conversion formula of MkII FPA measurement to COSMIC FFP is possible, but the reverse is not true. If the system is estimated by MkII FPA and the result is to be converted to COSMIC FFP, we have detailed information on the number of data groups and data movement types. However, when we want to convert COSMIC FFP size to MkII FPA, we do not have information on the number of DETs, i.e. we do not know if the system has a high or low number of DETs to decide on a formula.

Few studies exist to evaluate and compare different FSM methods. More theoretical and empirical studies are required in order to explore their differences and similarities. These studies would also help to define some form of conversion among FSM methods so that the measurement results would be comparable.

6. References


