# ISO Transposition and Clarifications of the COSMIC FFP Method of Functional Sizing

**Alain Abran, Peter Fagg, Roberto Meli, Charles Symons** [1]

**Abstract**

*This paper describes and explains a few significant changes which have been made to the COSMIC FFP method of functional sizing of software targeted to be published in the Measurement Manual Version 2.2 (October 2002) and in the draft ISO/IEC 19761standard version of the method. These changes have been made to help understanding and consistent use of the method.*

*None of the changes alter the principles of the method. These have never needed to be changed since the method was first defined, and have been confirmed by successful practical use in may organisations on different types of software. All the changes have arisen because it was found either that certain terms and definitions could be misunderstood and needed clarification, or because of the need to ensure consistency with existing ISO/IEC standard terminology and definitions (a design goal of the COSMIC FFP method).*

*Keywords: COSMIC-FFP, Software Functional Size, ISO 19761, Function Points*

## 1. Introduction

The COSMIC FFP method was designed to measure a functional size of software based on its Functional User Requirements ('FUR'), for

- software from the Business (or 'MIS', or 'data-rich'), and Real-time and infrastructure domains, and hybrids of these
- software in any layer of a multi-layer architecture, or any peer item within a layer.

From the outset, the COSMIC FFP method was designed to conform to the existing ISO/IEC standard 14143-1: 1997 (Ref. 1), which sets out the international agreement on the principles of Functional Size Measurement.

The full public definition of the principles and rules and a comprehensive description of the method to help understanding and explain the background was first published in a form intended for practical field trials in the Measurement Manual, version 2.0, in October 1999. (Ref: 2). The experience of the field trials resulted in an improved version 2.1 of the Measurement Manual, published in May 2001 (Ref: 3). The reader is assumed to be familiar with the latter publication.

Over the last year there have been two motivations to further improve the definitions of the method. First, feedback from practical use indicated the need to refine or clarify certain of the method definitions. Second the ISO/IEC Joint Technical Committee 1 accepted a New Work Item to transpose the method definition into an ISO/IEC standard. The process involves a rigorous checking of terminology for consistency with pre-existing ISO/IEC terminology on measurement, and several rounds of commenting from the ISO/IEC National Bodies. It is an excellent process for testing understanding on a wide scale, and to ensure full coherency over the set of measurement concepts, while using a minimal set of definitions and rules..

[1] Alain Abran and Charles Symons are the joint leaders of the COmmon Software Measurement International Consortium (the COSMIC team). Alain Abran was the first Project Editor of the new ISO/IEC 19761 standard for the COSMIC FFP method, and Peter Fagg has recently taken over that role. Roberto Meli is the Chairman of the COSMIC Measurement Practices Committee, which is responsible for maintaining the Measurement Manual.

As an example of the work that had to be done in producing an ISO standard, the rules must be clearly identified within normative clauses, and must be written in a style that specifies compliance criteria using the expression 'shall'. This standardisation in normative clauses then requires that all further informative text be clearly distinguished from compliance text, through Notes to the text, or through clearly segregated Examples. Hence these ISO editing standards also led to the elimination of text that had been included in the MM 2.1 for informative purposes and further guidance.

Hence the resulting ISO/IEC 19761 standard (Ref. 4) will take over as the ultimate authority on the essential principles of the method. The COSMIC Measurement Manual will continue to give a fuller account with background explanations and examples to facilitate training, but henceforth the aim will be to keep it in step with the International Standard. It is constructed to be a guide to the application of the International Standard.

The ISO/IEC process is now reaching its final stage so consequently the changes made for the most recent draft of the International Standard version (Ref. 4) have had to be fed back into an updated version 2.2 of the COSMIC FFP Measurement Manual. (Hereinafter we will abbreviate Measurement Manual to 'MM'.)

The purpose of this paper is to discuss the main areas of debate and consequent changes that have been made in producing the International Standard version and in evolving from version 2.1 to 2.2 of the MM. We do not deal with

- minor editorial changes made in producing the International Standard
- other changes to the MM that are not directly connected to the International Standard, or general improvements to the MM which could not be described as 'corrections'.

Throughout this process of refinement, it must be emphasised that none of the changes that have been made have altered the principles of the method that the authors defined, or intended to define, since the method was first published.

## 2. The COSMIC-FFP Model

Measurement of the functional size of a 'piece of software' can be carried out whether the software exists only as a statement of FUR, or by inferring its FUR from a piece of software that has already been implemented, or at any stage in between. Regardless, before the measurement can be made the FUR must be mapped into the COSMIC FFP conceptual model.

In brief outline, the key concepts proposed in the COSMIC FFP model and in the process of applying it are as follows.

- It is first necessary to define the *Purpose* and S*cope* of a measurement, together referred to as the *Viewpoint*
- The software within the Scope may need, depending on the Viewpoint, to be broken down into one or more *pieces of software*, in separate *layers*, or *peer items* within a layer
- Each set of FUR for a piece of software whose size has to be measured has one or more *Users* that interact with the software across a *Boundary*
- Any set of FUR can be decomposed into one or more *Functional Processes*
- Any Functional Process can be decomposed into two or more *Data Movements*
- A Data Movement is a functional sub-process that moves a *Data Group* (one or more *data attributes*) about a single *Object of interest*, and which may have some associated data manipulation
- A Data Movement can be one of four types
    - an *Entry* moves a Data Group from a User across the Boundary into the software
    - an *Exit* moves a Data Group from the software across the Boundary to a User
    - a *Write* sends a Data Group from the software to persistent storage
    - a *Read* retrieves a Data Group from persistent storage for the software
- Each Data Movement is allocated one COSMIC FFP functional size unit (or one *'Cfsu'*)
- The size of a piece of software is the sum of the sizes of its constituent Functional Processes; the size of a Functional Process is the sum of the sizes of its constituent Data Movements. (A rule is also given for measuring the size of a required change to a set of FUR).

## 3. Clarifications and Refinements of the Model

The definitions of the concepts of any Functional Size Measurement Method, and similarly the definitions of concepts of the 'meta standard' of ISO/IEC 14143-1, are inevitably inter-dependent. It is difficult to discuss any one definition in isolation from related definitions. The same is true for the content of this paper, so the following is in places an iterative explanation of a series of related clarifications or refinements.

### *The 'User'*

In the MM V2.1, 'Users' are defined as: *'Human beings, software or engineered devices which interact with the measured software'*.

However, ISO/IEC 14143-1 defines a 'User' as *'Any person that specifies Functional User Requirements and/or any person or thing that communicates with or interacts with the software at any time'*. The definition also has a Note: *'Examples of 'thing' include, but are not limited to software applications, animals, sensors or other hardware'*.

With hindsight (since the current authors participated in the development of ISO/IEC 14143-1) this ISO/IEC definition clearly embraces two separate concepts in the one term. First there is what we might call the 'specifying user' (who specifies 'Functional User Requirements'). This use of 'User' is consistent with its use in the expression 'Functional User Requirements'. But the second usage, what we might call the 'physical user' (any 'thing' that actually 'communicates with or interacts with the software…') is a different concept. The 'specifying user' need not actually ever use the software in practice. On the other hand the 'physical user' might be an engineered device that only sends signals to the software, but has nothing to do with its specifications. Unfortunately the ISO/IEC 14143-1 terms and definitions cannot be changed until a formal revision process is started.

In the meantime, the ISO/IEC standard version of the COSMIC FFP method must adopt the term already defined in ISO/IEC 14143-1 and since our goal is to remain compatible and in synch with the International Standard, the MM V2.2 must adopt this definition. However, in the latter the COSMIC team is free to add clarifications.

Hence in the MM V2.2, we clarify that when we use the term 'user' in the MM, we restrict it to the meaning of *'any person or thing that communicates with or interacts with the software at any time'*. When we mean 'specifying user', we will say so explicitly. This distinction also has implications for the definitions and use of the terms 'Boundary' and 'Viewpoint'.

### *The Boundary and (Persistent) Storage*

In ISO/IEC 14143-1, the Boundary is defined as *'A conceptual interface between the software under study and its users'*. (Note that this definition can only be interpreted correctly if 'users' means 'physical users', as we explained above.)

The MM V2.1 definition of Boundary is much longer, but essentially says the same thing. There is no difficulty in the MM V2.2 in adopting the ISO/IEC definition, subject to the correct understanding of 'User'.

However, again with hindsight, there is considerable potential for confusion, this time in the MM V2.1. This document uses 'Boundary', but also uses the terms 'I/O Boundary', 'Storage Boundary', 'Layer Boundary' and 'Software Boundary' in the text and in various diagrams, without defining them. As an example of the possible confusion, it turns out to be a reasonable inference (but never the intention) from the definition of User and the use of the expression 'Storage Boundary' that Users could also include Storage, and that Storage would lie *outside* the Boundary. An inconsistency then arises because only Entries and Exits were defined as moving data across the Boundary, whereas Writes and Reads were not defined as moving data across a Boundary.

To rationalise this, the following changes have been made to the ISO/IEC definitions of the COSMIC FFP method and to the MM V2.2.

Only the term 'Boundary' is used, unqualified by any adjective, with the definition of ISO/IEC 14143-1. It is a conceptual boundary, and should not be confused with any other type of boundary that could be envisaged, e.g. the physical boundaries between one layer of software and its adjacent layers.

The term 'Persistent Storage' is introduced, to replace 'Storage' everywhere it occurs. 'Persistent Storage' is defined as *'storage that is on the software side of the boundary of the software being measured and that is continuously accessible to the software during its execution.'*

Furthermore two Notes have been added to this definition, as follows:

*'NOTE 1 In the COSMIC FFP model, because Persistent Storage is on the software side of the boundary, it is not considered to be a User of the software being measured.*

*NOTE 2 Persistent storage enables a functional process to store data beyond the life of the functional process and/or enables a functional process to retrieve data which was*

- *stored by another functional process*
- *or stored by an earlier occurrence of the same functional process*
- *or stored by some other process, e.g. in the manufacture of a read-only memory'*

The reason for adding the phrase *'that is continuously accessible during its execution'* is to help distinguish persistent storage from storage which is accessed instantaneously, for example the swiping of a credit card as a means of entering data.

The addition of the phrase *'storage that is on the software side of the boundary …'*, and of Note 1, is needed because this aspect of the definition has always been understood but has not previously been made explicit.

***Viewpoint (Purpose and Scope)***

We have already pointed out that the ISO/IEC 14143-1 standard fails to distinguish 'physical user' and 'specifying user'. Similarly, in the MM V2.1 the same word 'user' is employed, on occasion, for 'physical user' and 'end-user' (e.g. see section 2.3.1, CASE 2).

We will define an 'end-user' as a person who uses a business application and 'sees' only the application functionality, being unaware of the functionality provided by the infrastructure software (operating systems, device drivers, utilities, etc) on which the business application depends. Contrast this with a physical user who only 'sees' the functional processes of the particular piece of software with which he/she/it interacts.

It has become much clearer, as a result of analyzing the difficulties with the term 'user', that the concept of the 'Viewpoint' of a measurement is of vital importance. In the new ISO/IEC standard for the COSMIC-FFP method and in the MM V2.2, the Viewpoint is given much more prominence, and must be reported with any measurement result. 'Viewpoint' is at the moment defined as an umbrella term for the Purpose and Scope of the measurement.

Some other Function Point methods such as the IFPUG (Ref 5.) and the MkII (Ref. 6) methods were designed to measure a functional size from the Viewpoint of the 'end-user', as we have defined above. The COSMIC-FFP method can be used to measure this size, but can also be used to measure the size from the Viewpoint of the 'specifying user' (hopefully the same Viewpoint as that of the 'developer', if they are communicating properly) and from any other Viewpoint as long as it is defined.

The sizes seen from the end-user and from the developer's Viewpoints could, of course be very different. Hence the emphasis now in both the new ISO/IEC standard for COSMIC-FFP and in the MM V2.2 on first agreeing the Viewpoint for a measurement and the need to specify the Viewpoint whenever a measurement result is reported. The Viewpoint has also recently been added as a

mandatory parameter for the reporting of COSMIC-FFP measurement-based results to the ISBSG database (Ref. 7)

Ideally, at some time in the future certain Viewpoints should be standardized, so that measurement results from different sources can be safely compared. The 'end-user' (for business applications, where the 'end-user' is restricted to a 'human') and 'developer' Viewpoints would be the first major candidates to be defined and standardized. Until this stage is reached, one must be very careful in using data from sources where no Viewpoint has been defined, and before comparing data to measure convertibility between measurements based on the COSMIC-FFP method and those based on 'traditional' Function Point methods. It is to be noted that COSMIC-FFP can take into account all the Viewpoints allowable by the ISO meta-model 14143-1, its 'user' definition including humans, engineering devices and other software as users of the piece of software being measured. But the Viewpoint must always be stated if the measurement is to be understood properly.

***Definitions of 'Read' and 'Entry'***

Certain additions to the rules for these concepts have been made to reduce the possibility of mis-interpretation.

First, it had not previously been made explicit that in physical reality, Entry and Read data movements may involve bi-directional exchanges. Thus a non-triggering Entry may be preceded by a 'request to receive the Entry' and a Read must be preceded by a 'request to Read'. Nevertheless, in both cases a logical view is taken and only a single Data Movement is assigned, that is one Cfsu, to any Entry or Read in the functional process where it is identified.

It is now stated explicitly that any 'request to receive an Entry' functionality and any 'request for a Read' functionality is not measured separately from the Entry and from the Read respectively.

Second there is a need to eliminate any measurer concluding that it is within the rules to include in the measurement the use of intermediate variables, etc as Reads (or Writes). This is addressed now by adding the following to the rules for a Read.

*'During a functional process, the Read or Write of a data group can only be performed on the data describing an Object of Interest to the user. Constants or variables which are internal to the functional process or intermediate results in a calculation, or data stored by a functional process resulting only from the implementation rather than the FUR, are not data groups, and are not measured.'*

***Minor clarifications***

Use of the terms 'client' and 'server' has been eliminated from the ISO/IEC standard for COSMIC -FP and from the MM V 2.2. These terms were used in the description of the relationships between layers and between peer pieces of software within the same layer. But the terms 'client' and 'server' have connotations beyond that intended in the COSMIC-FFP method, and represented a potential source of confusion.

In the rules for identifying functional processes for real-time software, there has always been the statement that a functional process 'terminates when a point of asynchronous timing is reached'. This has not been clear and could have been interpretedas a technical, not a functional characteristic. Furthermore, if the end-user of a business application chooses to take a break whilst part-way through entering data for a given functional process a point of asynchronous timing may be reached, but this was never intended to indicate that the functional process could be complete

The rule has been clarified by amending it to state that a functional process 'terminates when a point of asynchronous timing is reached *according to its FUR*'.

## 4. Conclusion

Feedback from field experience of using the COSMIC-FFP method and the discipline of producing an ISO/IEC standard version of the method has been extremely helpful in discovering areas of weakness in the method's concept definitions.

We believe that the new ISO/IEC 19761 standard for the COSMIC-FFP method and the new MM V2.2 will significantly improve the probability of consistent interpretation of the method and hence of obtaining repeatable functional size measurements.

**Acknowledgements**

The authors wish to thank the following for their important contributions to the discussions which have led to the clarifications and refinements described in this paper.

Moritsugu Araki, Jean-Marc Desharnais, Vihn Ho, Pam Morris, Marie O'Neill, Tony Rollo, Grant Rule, and Denis St Pierre.

The authors also wish to acknowledge with gratitude the feedback received from users of the COSMIC-FFP method in the field, and from the many experts who studied the drafts of the ISO/IEC standard for COSMIC-FFP and submitted comments via their National Bodies.

**References**

1. ISO, ISO/IEC 14143-1997 – Software Engineering – Software measurement – Functional size measurement – Part 1 : Definition of concepts.

2. Abran, A., Desharnais,J.M., Oligny, S., St-Pierre, D. and Symons, C., "COSMIC FFP – Measurement Manual Version 2,0 - Field Trials Version" Université du Québec à Montréal, Montréal, October 1999.

3. Abran, A., Desharnais, J.M., Oligny, S., St-Pierre, D. and Symons, C., "COSMIC FFP – Measurement Manual Version 2,1" Université du Québec à Montréal, Montréal, May 2001.

4. ISO. ISO/IEC 19761 (2002) 'Software Engineering. COSMIC-FFP. A Functional size measurement method'. (This standard is in its Final Draft International Standard status and is currently out for final National Body voting)

5. The International Function Point User Group, Counting Practices Manual, V4.1, 2000.

6. The UK Software Metrics Association, MkII Function Point Analysis. Counting Practices Manual, V 1.3.1, September 1998.

7. ISBSG. The International Software Benchmarking Standards Group. See 'www.isbsg.org'.