Implementing COSMIC-FFP as a replacement for FPA

Frank W. Vogelezang

Sogeti Nederland B.V.

frank.vogelezang@sogeti.nl

Abstract:

This article shows the first results of the adoption of COSMIC Full Function Points as a sizing method replacing function point analysis. The main arguments why COSMIC-FFP was chosen will be explained, the transformation plan will be shown together with the first results of the use of COSMIC-FFP. Next to the management requirement that the new functional sizing method had to be a standard a number of practical requirements were essential before the transformation could start: to find a correlation between Cosmic functional sizing units and function points so that the existing figures for size and product delivery rate could be reused, (early) estimation possibilities and the use of COSMIC-FFP for sizing maintenance projects.

Keywords

COSMIC-FFP, implementation, transformation, function point analysis

1 Organizational setting

This article describes the transformation process of an IT department of Rabobank. This department used function point analysis as functional sizing method since 1995 and adopted COSMIC-FFP in January 2003. In this transformation process Sogeti acted as the knowledge provider.

1.1 Rabobank

Rabobank is one of the larger banks in the Netherlands. It is a cooperation of over 350 local independent banks with a history in retail and agriculture which together form the Rabobank group. Sogeti introduced function point analysis in 1994 to the IT department that serves the processing of payments and savings contracts and transactions but this department did not use the method structurally. In 1999 function point analysis was successfully reintroduced by establishing a metrics group within the IT department [1].

1.2 Sogeti

Sogeti Nederland is a Dutch software services company with 1.900 employees. In august 2002 IQUIP Informatica, Gimbrère & Dohmen and Twinsoft merged to form Sogeti Nederland as part of the worldwide Sogeti Group. Since 1988 IQUIP is known in the Netherlands as a promoter and initiator of functional size measurement. Sogeti continued that leading role of IQUIP by establishing an Expertise Center Metrics. Sogeti plays an active role in the promotion and further development of COSMIC-FFP by participating in working groups of the NESMA (Netherlands Software Metrics Association) and in both the International Advisory Council and the Measurement Practices Committee of COSMIC.

1.3 Metrics group

At the first introduction in 1994 a number of developers was trained to perform function point analyses of their own projects. In the project guidelines the moments for analysis were indicated to ensure the use of these metrics. Despite this, not all projects participated and those projects that did participate experienced the usual start-up problems: differing outcomes of analysis, lack of productivity rates, questions and doubts about the added value and the daily work pressure of the project. This initiative ended after about a year and a half. All people concerned were convinced that function point analysis was useful but not for their projects [2].

In 1999 function point analysis was reintroduced using the implementation model MOUSE [3]. One of the key elements of this implementation model for metrics programs is the establishment of an independent body to oversee the use of metrics [4]. In the case of Rabobank, Sogeti established a metrics group that does not only oversee the use of metrics, but also performs the functional size measurements. In this way functional size measurement is fully independent of the project organization.

2 Prelude to a transformation

2.1 The limitations of function point analysis

Since a few years the market demands more complex financial products with a shorter time-to-market. This made Rabobank look for a new IT strategy. Information systems should be able to focus on the client (a client with one or more products) rather than on the product (each product has its own clients). This demanded a change in architecture from dedicated product systems to architecture with a shared data source for shared (client) data.

The set of dedicated product systems are now being migrated to a net of generic service components organized in (front-end) distribution services, client services and (back-end) product services. IT support for new financial products now usually contains linking services between various existing systems together with new service components. A negative effect of this architectural transformation was that function point analysis no longer gave appropriate functional sizing figures for this architecture, since one of the basic principles of function point analysis is the coupling of data and functionality within the information system to be sized [5].

2.2 The search for a new sizing method

First an attempt was made by the metrics group to modify function point analysis so that it could be used in the new architecture. Rules were drawn up to interpret an overlying layer as the external user and to interpret an underlying layer as one or more Internal Logical Files or External Input Files. These rules were tested with a few projects and it became obvious that the rules that were drawn up for both the overlying as the underlying layer were not unambiguous. It appeared to be impossible to compare the sizing values from different layers. Bending the rules of function point analysis to fit an architecture that used different basic principles appeared to be impractical.

In 2001 a reference model for functional sizing was proposed by Dekkers and Kammelar [6]. With this model a functional sizing method could be designed that is compliant with ISO/IEC 14143 to fit any kind of architecture or environment. Experiments with this tailor-made functional sizing method showed promising results. This method would be a good functional sizing method for estimating internal projects, but had the disadvantage that external benchmarking would not be possible. Since benchmarking was becoming more and more important, a new functional sizing method had to be a method that could be standardized (or could be easily converted to a standardized method). Rabobank is no exception in adopting contemporary architectural views and systems development methods, finding the right functional sizing method should not be a unique problem.

2.3 COSMIC-FFP, a next generation functional sizing method

In the 1980’s and 1990’s, researchers have documented a number of theoretical flaws in function point analysis. These studies had little impact on the practical value of the method. It only discredited function point analysis as a valid scientific research topic. In late 1998, some members of the ISO working group on functional size measurement decided to develop a new functional sizing method, starting from basic established software engineering principles.

This method should be equally applicable to business application software, real-time software and to infrastructure software and was aimed to be compliant with ISO/IEC 14143 from the outset.

The development of this new method resulted in COSMIC-FFP. The first public version of the method, COSMIC-FFP v2.0, was published in October 1999. COSMIC published its latest definition of the method, v2.2, in January 2003.

COSMIC-FFP is the first so-called next generation functional sizing method that is specifically designed to meet the generic scientific principles of ISO/IEC 14143 [7]. It was designed to be able to meet the constraints of the many new (and complex) types of data-driven and event-driven software as well as the type of software served by first generation functional sizing methods. For example COSMIC-FFP is able to recognize the use of different layers in software and is able to measure functional size from different measurement viewpoints, thus helping to overcome the uncertainty on what is meant by ‘functional’ in the user requirements. It has also been designed to be easy to train, understand, and use with consistency without recourse to inter-related rules and exceptions.

3 Requisites for the transformation

Within Sogeti it was verified that COSMIC-FFP met the technical requirements of Rabobank for a functional sizing method. Some questions remained that could not be answered beforehand:

- Is it possible to convert function point analysis data to COSMIC-FFP?
- Can COSMIC-FFP be used for early estimation?
- How well can maintenance projects be estimated with COSMIC-FFP?

Sogeti and Rabobank together financed a research project to investigate the above questions and to rearrange the sizing and estimation process.

3.1 Conversion possibilities

To determine a possible correlation between function points and COSMIC functional sizing units (cfsu) only those projects were selected that had made an unadapted use of function point analysis. If usually are the result of a mismatch between the principles of the functional sizing method and the development method used to design the software. If interpretations of the counting rules had been made to measure a project, this measurement would be dismissed from the conversion project. Interpretations can be considered as small adaptations of the function point analysis method. We decided that a valid comparison could only be made if both methods to be compared would be used without modification.

NESMA 2.2	COSMIC 2.2
39	23
52	29
260	81
170	109
120	115
249	173
218	181
224	182
380	368
766	810
1424	1662

Table 1: Sizing results

If the preconditions were met a project could be resized using COSMIC-FFP. The End User Measurement Viewpoint was used because this viewpoint uses a definition of the user that is the most similar to that of Function Point Analysis. In the beginning of 2003 eleven projects have been sized with both methods (see table 1 and figure 1).

To evaluate whether there is any form of correlation between the size of a project in function points and in cfsu linear regression was used. Since both methods should describe the same attribute of a software project: the functional size as seen from the perspective of the end user, it seemed reasonable to expect a linear correlation between the two methods.

Using linear regression the conversion formula from function points to cfsu was derived as:

$$\mathbf{Y\ (cfsu) = -87 + 1.2\ X\ (fp)}$$

The correlation coefficient is 0.99 and the standard deviation in the difference in the Y-value is 59. This means that in the Rabobank environment there is a fairly good correlation between the size in function points and the size in cfsu.

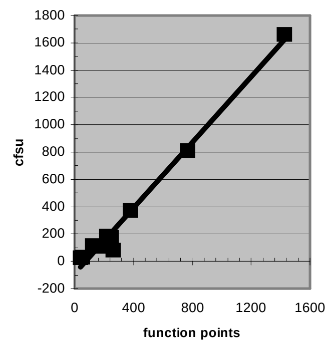

Figure 1: Correlation between fp and cfsu

The fact that there is an offset in this conversion formula might be explained by the fact that a substantial part of the size in function points (usually 30-40%) comes from the ILF and EIF. The existence of an ILF or EIF always leads to the same count in function points, whether they are fully maintained or not. COSMIC-FFP counts the use of data: if some entity is not fully maintained (which is often the case in our set of projects) this will lead to less data movements per entity and thus to a negative offset in the conversion formula from function points to cfsu. This theory has not yet been supported by evidence from research [8].

3.2 Early estimation

To support early estimation an approximate version of the COSMIC-FFP method can be used. In the Measurement Manual this is described in detail [9]. The manual also states that the approximate version might be different for different environments. To check this statement we have derived our own approximate version, based on the data of the first ten projects that were sized to derive the conversion formula (see table 1).

In the very early stages of software development only the number of functional processes is known. The approximate version provides an average value for the size of a functional process. To estimate the size of an application the number of functional processes can be multiplied by the average size of a functional process. In the example in the Measurement Manual, based upon development of avionics of a military aircraft, the average size of a functional process is 8 cfsu. From Rabobank data we have calculated the average size of a functional process to be 7.3 cfsu.

In a later stage of the development process there is sufficient information about the functional processes to classify them into different categories. The method described in the Measurement Manual to classify functional processes uses four categories:

- small (e.g. retrieval of information about a single object of interest)
- medium (e.g. storage of a single object of interest with some checks)
- large (e.g. retrieval of information about multiple objects)
- complex

These categories can be assigned average values by dividing the number of functional processes – ordered by size - into four quarts and computing the average size of a functional process in each of the quarts. In table 2 we present the comparison between data from the Measurement Manual and our own data.

Quart	Manual	Rabobank
small	3,9	3,6
medium	6,9	4,4
large	10,5	6,3
complex	23,7	14,9

Table 2:
Comparison of averages

The fields from which both sets of data originate are very different and not surprisingly the results are not comparable. This is in contrast with one of our earlier publications [8]. After the publication of that article we discovered that there was a difference between the method described in the measurement manual and the method used to calculate the values in the measurement manual.

Therefore we decided to test both methods of deriving an early estimation method:

[A] Dividing the total size into four quarts of equal size

[B] Dividing the total number of functional processes into four quarts of equal numbers

In table 3 the outcome is presented for the range of the quarts of each method. Method B gives ranges for small, medium and large that are so close together that it is not a practical method to use for early estimating: the ranges are so close together that there is a large risk of misqualifying a functional process into the wrong quart while making an early estimate. On the other hand the penalty for misqualifying small or medium functional processes is only 0.8 cfsu.

Quart	Range	Avg.
Method [A]		
small	≤ 5	4,0
medium	5-8	6,2
large	8-14	10,8
complex	≥ 14	24,7
Method [B]		
small	≤ 4	3,6
medium	4-5	4,4
large	5-8	6,3
complex	≥ 8	14,9

Table 3: Comparison of quarts

To test the predictability of these methods we recalculated the size of the 11 projects in section 3.1 by substituting the real value with the average value of the corresponding quart of the early estimation method.

Method	Precision overall	Precision per project
[A]	6%	13%
[B]	9%	17%

Table 4: Comparison of precision

We calculated the precision of the methods in two ways: the precision of the total size of all 11 projects and the average absolute precision for each project. Both calculation methods show the same trend that the method described in the Measurement Manual is the better predicting estimation method. Because we only tested the predictability within one organization further research will be necessary to conclude if there is a general truth about the best predicting estimation method.

3.3 Maintenance

Most software projects are enhancements to existing software. In the early nineties of the last century a working group of NESMA first proposed a method for measuring enhancement using function point analysis [10]. In 1998 this method was published as a professional guide, not as a part of the NESMA standard. This method distinguishes between project size (which can have a fractional value) and application size (which is always a whole number). Rabobank used this method before the conversion to COSMIC-FFP. This method has substantial acceptance in the Netherlands, but very little acceptance in the rest of the world, where the IFPUG view on measuring enhancement projects is most common that does not distinguish between project and product size.

In COSMIC-FFP measuring changed functionality is part of the method. Section 4.3b of the Measurement Manual describes that the size of a changed functional process is an aggregation of the number of modified data movements (added, modified and deleted). As with new functionality this results in a size of a whole number of cfsu. Dividing the size of the changed functional process by the original size results in a factor that is usually in range with the maintenance factor from the NESMA method for measuring enhancements.

Next to new functionality and changed functionality enhancement projects usually also contain deleting existing functionality. Strict application of the description on how to deal with changed functionality means that deleting a functional process has the same impact on the functional size as creating new functionality. For the application size this is obviously true, but for the project size it will overestimate the corresponding work effort. To deal with this problem COSMIC-FFP offers the possibility of using local extensions to the method. A more appropriate way to deal with this problem is to use a different product delivery rate for disconnecting existing functionality. COSMIC envisages producing guidelines for each domain which gives detailed guidance on dealing with this kind of questions [11].

4 Implementing COSMIC-FFP

Three questions had to be answered positively before COSMIC-FFP could be accepted as a replacement for function point analysis (see section 2.3):

- Is it possible to convert function point analysis size to COSMIC-FFP size?
- Can COSMIC-FFP be used for early estimation?
- How well can maintenance projects be estimated with COSMIC-FFP?

The answers were convincing enough and the management of the bank gave permission to start the transformation. Sogeti was asked to draw up a plan for the implementation of COSMIC-FFP as a replacement for function point analysis.

4.1 Planning

The implementation plan, based on the MOUSE method for implementing a metrics program in an organization [4] addressed the following main issues:

- decision which viewpoint(s) to use
- preparing information for the organization about COSMIC-FFP (summary of the method, intranet, presentations)
- changing guidelines and procedures
- converting historical function point analysis data to COSMIC-FFP

In the implementation plan 720 hours were planned for the implementation of COSMIC-FFP, excluding attendance to presentations where the new method would be presented. Details of this plan are described in [2].

4.2 Transformation

Most activities related to functional size measurement at Rabobank are performed by the Metrics group (see section 1.3). The transformation from function point analysis to COSMIC-FFP had a large impact on this group because it had to use a different technique to do its work, but the impact for the rest of the organization was very small. The actual replacement of function point analysis by COSMIC-FFP consumed 837 hours, 16% more than estimated. (for details, see [2])

4.3 Lessons learned

The transformation of the functional sizing method only impacted a small number of people and went relatively smooth. Still some lessons were learned with a more general applicability than this implementation.

ⓘ (regression) analysis

To gain benefits from the investments in function point analysis in the past, the possibility was investigated to reuse the measurements done in the past (see section 3.1). Selection of projects appeared difficult, because in a substantial number of projects assumptions about the described functionality or interpretations of the sizing rules had to be made in order to be able to size the reported project, which disqualified those projects for the conversion project (see also section 3.1). The effort for the selection process was underestimated.

ⓘ information to project managers

To project managers it does not matter whether they get function points or Cosmic functional sizing units as a sizing measure. Both represent size and the estimation process does not change; only the values of some variables change. Their only interest is the estimate and not the way such an estimate is produced. They only needed to know the highlights of COSMIC-FFP and the new values for their estimation variables. The effort to inform this group was overestimated.

ⓘ transformation

This was the real underestimated activity. It was more difficult for experienced function point analysts to learn to apply COSMIC-FFP. Because of the different approach to data (groups) COSMIC-FFP requires a different mind-set. This is an important lesson learned and this has to be taken into account for future transformations.

5 First results

The results from analysis of the first projects sized with COSMIC-FFP prove that COSMIC-FFP is useful and can be applied in the same domain as function point analysis. Projects that were difficult to size with function point analysis could well be sized with COSMIC-FFP. The range of software (specifications) that can be sized with COSMIC-FFP is definitely wider then the range covered by function point analysis.

Time spent on sizing the 10 pilot projects with function point analysis was on average 30 to 35 hours per project. The effort to size with CFFP was on average 21 hours. This matches the feeling that sizing with COSMIC-FFP is easier. The accelerating effect of resizing the same application was absent because another analyst did the second sizing.

The lack of well-documented sizing guidelines was felt clearly when sizing with COSMIC-FFP. Although the concepts are described accurately, practicing the concepts when sizing projects is not always easy. As a consequence one of the team members, who played a leading role in the described implementation, became an important participant in the conception of the first COSMIC guideline for the sizing of business application software [11].

6 Conclusions

It is possible to implement COSMIC-FFP as a replacement for function point analysis. The range of software (specifications) that can be sized is wider then the range covered by function point analysis. In our case function point analysis data could be converted to COSMIC-FFP, although further research is necessary to prove general applicability of this conversion.

We've shown that COSMIC-FFP can be used in the early estimation of projects and that it also can be used to size enhancement projects.

An important lesson from this implementation is that changing the mind-set from function point analysis to COSMIC-FFP can be hard for experienced function point analysts.

References

1. Leeuwis, W., 'Functiepuntanalyse bij IBV', NESMA spring conference 2001
2. Dekkers, A.J.E., Vogelezang, F.W., 'COSMIC Full Function Points: Additional to or replacing FPA', Proceedings of the 8th European Systems and Software Engineering Process Group Conference (ESEPG 2003), London, June 2003
3. Dekkers, A.J.E., 'The Practice of Function Point Analysis', Proceedings of the 8th European Software Control and Metrics conference (ESCOM 1997), Berlin, 1997
4. Vogelezang, F.W., 'Implementing metrics in an organization', NESMA Yearbook, 2004
5. NESMA, 'Definitions and counting guidelines for the application of function point analysis - A practical manual', version 2.0, 1996
6. Dekkers, A.J.E., Kammelar, J., 'A functional sizing meta model', Proceedings of the 4th European conference on Software Measurement and ICT Control (FESMA – DASMA 2001), Heidelberg, 2001
7. Abran, A., Meli, R., Symons, C., 'COSMIC-FFP (ISO 19761) Software size measurement: State of the art 2004', Software Measurement European Forum (SMEF 2004), Rome, January 2004
8. Vogelezang, F.W., Lesterhuis, A., 'Applicability of COSMIC Full Function Points in an administrative environment - Experiences of an early adopter', Proceedings of the 13th International Workshop on Software Measurement (IWSM 2003), Montréal, September 2003
9. COSMIC MPC, 'Measurement manual – The COSMIC implementation guide for ISO/IEC 19761', Version 2.2, January 2003
10. Engelhart, J.T., Langbroek, P.L., Dekkers, A.J.E., Peters, H.J.G., Reijnders, P.H.J., 'Function point analysis for software enhancement, A professional guide of the Netherlands Software Metric Users Association', NESMA 2001 (in 1998 the dutch version was already published)
11. Lesterhuis, A., 'The COSMIC-FFP Business Application Software Guideline', Proceedings of the 14th International Workshop on Software Measurement (IWSM 2004), Berlin, November 2004