Applicability of COSMIC Full Function Points in an administrative environment
Experiences of an early adopter

Frank Vogelezang, Arlan Lesterhuis
Sogeti Netherlands B.V.
frank.vogelezang@sogeti.nl, arlan.lesterhuis@sogeti.nl

Rabobank is reshaping its systems portfolio of Rabobank from dedicated product systems to a network of generic services with a shared data source. In this environment Function Point Analysis no longer fits the sizing needs. An alternative was found in the COSMIC Full Function Points method. Because of the absence of benchmark data a conversion formula was derived for projects that were measurable in both COSMIC Full Function Points (End User Viewpoint) and Function Point Analysis. This conversion formula now reads as:

$$Y\ (cfsu) = -87 + 1{,}2\ X\ (fp)$$

The correlation coefficient for this conversion formula is 0,99 and the standard deviation in the difference in the Y-value is 59.

To support the estimating process in early stages of systems development the locally calibrated approximate version of COSMIC Full Function Points was derived from the first set of measurements. Our version shows very good resemblance to the version presented in the Measurement Manual. Because these figures were derived in a very different environment this might be an indication that these figures have a more general applicability.

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. This group has one subsidiary company for common activities (leasing and factoring, investment, securities and insurance) and support (purchase, maintenance, ICT). The ICT-support is organized in four ICT-departments: Rabobank ICT for office and infrastructure applications, RNE to support securities, IF to support loans and finance and IBS to support payments and savings.

Sogeti Netherlands

This paper descibes aspects of the introduction of the COSMIC Full Function Points method (CFFP) within two ICT-departments of Rabobank (IBS and IF). To be able to understand the situation it is also relevant to know what Sogeti Netherlands is and how Sogeti is involved.

Sogeti Nederland B.V. is a Dutch software services company with 1700 employees. In august 2002 IQUIP Informatica, Gimbrere & Dohmen and Twinsoft merged to form Sogeti. Since 1988 IQUIP is known in the Netherlands as a promoter and initiator of functional size measurement. Sogeti continues the leading role of IQUIP by means of the Expertise Centre Metrics of the Engineering & Projects division. Sogeti plays an active role in the promotion and further development of CFFP by participating in working groups of the NESMA (Netherlands Software Metrics Association) and the Measurement Practices Committee of COSMIC. For over four years Sogeti supplies the know-how and the manpower for the functional sizing of software projects for the two mentioned ICT-departments by means of a local Expertisegroup Metrics [1].

The need for a new sizing method

The ICT-departments IBS and IF develop information systems for the payment-, savings- and finance departments of Rabobank. Until recently, each product of one of these departments was supported by its own information system with dedicated data. If a client wanted more than one product, various information systems needed to be activated and synchronized. To gather management information specialized information systems are needed to collect, synchronize and combine data from a whole range of product systems.

The changing market demands more complex products with a shorter time-to-market which made Rabobank look for a new ICT-strategy. Information systems should be able to focus on the client (a client with one or more products) instead of on the product (each product has its own clients). This meant a change from an architecture with dedicated product systems to an architecture which contains a shared data source for shared (client) data.

The *set* of dedicated product systems are migrating to a *net* of generic service components organized in (front-end) distribution systems, client systems and product systems. New information systems now usually contain links between various existing systems together with new service components. This means Function Point Analysis (FPA) no longer gives appropriate sizing figures for this architecture, since one of the basic principles of this technique is the coupling of data and functionality within the information system to be sized [2].

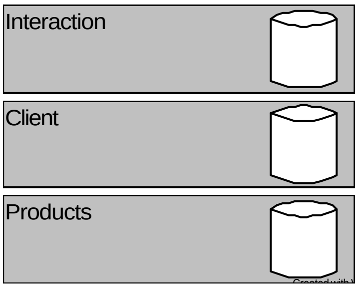

Figure 1 : Rabobank architecture

The chosen architecture is designed to uncouple the (product) data from the (client) functionality. It is therefore not fit to be sized with a technique which uses the basic principle that all data and functionality are within the information system to be sized. Figure 1 is a simplified representation of the Rabobank architecture:

- The interaction layer contains the functionality to interact with the outside world. The data contained in this layer only holds information corresponding to the channel (e.g. internet, terminal, ATM, mobile phone) of interaction with the outside world.
- The client layer contains the functionality to gather or store information at client level. The data contained in this layer holds information about the client itself, the channels he uses or is allowed to use and about the products he has. Details about the exact use of channels and products is contained in the other layers.
- The products layer contains the data about the products portfolio of Rabobank and the product details. Functionality in the products layer only responds to events from the client layer.

The search for a new sizing method

First an attempt was made to modify FPA 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 (ILF) or External Input Files (EIF). The rules were tested with a few projects and it became obvious that the drawn up rules for both the overlying as the undelying layer were not unambiguous. In addition there was a lot of discussion over the relation between the weight of real ILF's and EIF's from the counted layer and defined ILF's and EIF's from the underlying layer. Also a discussion started about the impossibility to compare the sizing values. At this point the main issue was that we would have four different kinds of sizing values: real function points for complete systems and derived 'layer points' for each type of layer. Bending the rules of FPA to fit an architecture that used different basic principles appeared to be impractical.

In 2001 a reference model was proposed by Dekkers and Kammelar [3]. With this model a functional sizing method could be designed to fit any kind of architecture or environment. Early 2002 Sogeti designed a functional sizing method for an object oriented environment of Rabobank ICT [1]. In this method the following estimating elements were identified:

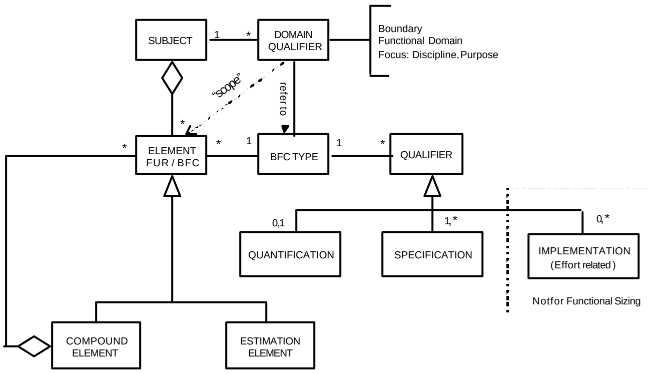

Figure 2 : Functional size reference model

- man-machine interface
- processes
- model
- services

Experiments with this tailor-made functional sizing method showed promising results. But the discussion about comparability of this method started anew. This method would be a good method for estimating internal projects, but external benchmarking would not be possible. In a period where management wanted to compare the productivity of the ICT-departments with that of external parties there was no support for a non-standard new functional sizing method. The solution had to be a standard functional sizing method. Rabobank is not exceptional in adopting contemporary architectural views and systems development methods. The difficulties in finding the right functional sizing method should therefore not be unique.

The choice of a new sizing method

In the late nineteen nineties the Common Software Measurement International Consortium (COSMIC) was formed aiming to develop, test, bring to market and seek acceptance for a new software sizing method to support estimation and performance measurement in contemporary environments. This initiative resulted in CFFP [4]. Within Sogeti it was verified that this method met the technical requirements of Rabobank for a functional sizing method. Some questions remained that could not be answered beforehand:

- Can the historic data from function point analysis be converted to CFFP?
- Can CFFP be used for early estimation?
- How well can maintenance projects be estimated with CFFP?

Sogeti and Rabobank together financed a research project to investigate the above questions and to rearrange the sizing and estimation process. In this paper we will deal with the technical aspects. The organisational aspects are described in a paper on the ESEPG-conference by Dekkers and Vogelezang [1].

How to use a new sizing method for estimating new projects

With FPA Rabobank had a complete support of the sizing and estimation process, with two levels of appoximate estimating in the early stages of a software development project [2] and a functional sizing method for enhancement projects [5], supported by a measurement database with evaluation data from within the organization. For CFFP all three questions mentioned above could not be answered beforehand or were non-proven. Before CFFP could be accepted these aspects should be filled in.

The chosen approach was the following:

- Choose a number of projects already sized in function points and size them again using CFFP.
- Evaluate if there is a correlation between the size in function points and the size in COSMIC functional sizing units (cfsu) and derive a conversion formula.
- Convert the measurement database with the derived conversion formula.
- Use the measurement data in cfsu to derive the approximate versions of CFFP according to the guidelines in the Measurement Manual [4, chapter 7].

Conversion of function points to cfsu vice versa

To determine a possible correlation between function points and cfsu only those projects were selected that had made an unadapted use of FPA.

Within the Rabobank organization, all sizing reports contain a section that describes assumptions about the described functionality or interpretations of the counting rules that had to be made to be able to size the reported project.

Assumptions about the described functionality are usually the result of ambiguous documentation. For a new sizing using CFFP the same assumptions must be used to get comparable results.

Interpretations of the counting rules usually are the result of a mismatch between the principles of the functional sizing method and the development method used to design the software. Interpretations therefore are small adaptations of FPA. If this was the case a project would be dismissed from the conversion project because of adapted use of FPA.

NESMA 2.0	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 Rabobank

If the preconditions were met a project could be resized using CFFP. The End User Viewpoint was used because this viewpoint uses a definition of the user that is the most similar to that of Function Point Analysis. Up to now eleven projects have been sized with both methods (see table 1 and figure 3).

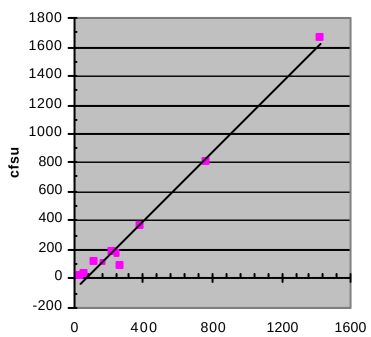

Figure 3 : Correlation between fp and cfsu

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 funtional size as seen from the perspective of the end user it seemed reasonable to expect a linear correlation between the two methods. With linear regression the conversion formula from function points to cfsu at this moment reads as:

Y (cfsu) = -87 + 1,2 X (fp)

The correlation coefficient is 0,99 and the standard deviation[1] in the difference in the Y-value is 59 so we can conclude that in the Rabobank environment there is a fairly good correlation between the size in function points and in cfsu. We hope that when

[1] For each application there are real sizing figures X (in fp) and Y (in cfsu). Using the formula each value X leads to a calculated value Y'. The standard deviation is the difference between Y and Y'. With a smaller standard deviation the formula better predicts the outcome of the real value Y.

more projects become available the correlation will increase and the standard deviation will decrease. Note that with more projects available also the values of the gradient and the offset might change.

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. CFFPoints counts the use of data: If some entity is not fully maintained (which is often the case in our set of projects) this leads to less data movements per entity and thus to a negative offset in the conversion formula from function points to cfsu. As far as we are aware this theory has not yet been supported by evidence from research.

IFPUG 4.1	COSMIC 2.0
31	29
40	38
49	51
56	52
77	81

Table 2 : Sizing results Fetcke

In 1999 Fetcke has written a case study about different methods of functional sizing [6]. This study was not intended to find conversion formulae, but from the data of this study conversion formulae between the various functional sizing methods can be calculated. The data that most resembled our own conversion data were that for IFPUG 4.1 and COSMIC 2.0.

With linear regression the conversion formula from function points to cfsu reads as:

Y (cfsu) = -6,2 + 1,1 X (fp)

The correlation coefficient is 0,99 and the standard deviation in difference in the Y-value is 2,6. We can conclude that the data from the study by Fetcke has an even better predictive value for the size in cfsu than the Rabobank figures. The value for the gradient has the same order of magnitude.

The fact that the offset is much smaller than with the Rabobank figures might be explained as a support for the theory that the offset is a translation of the influence of the partial maintenance of data. The applications studied by Fetcke were applications with a very small number of entities and these entities were fully maintained.

The differences between IFPUG 4.1 and NESMA 2.0 and between COSMIC version 2.0 and 2.2 are of such a nature that without detailed knowledge of the underlying data there is no real certainty whether the results can be combined. The differences between both sets of methods generally leads to results that are very similar. If we assume that both sets contain comparable results than the conversion is:

Y (cfsu) = -52 + 1,2 X (fp)

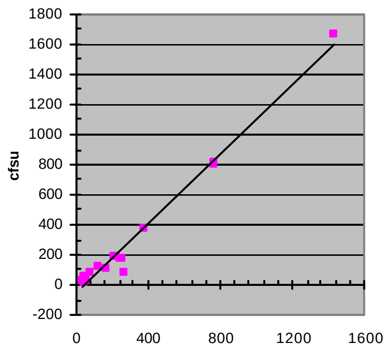

Figure 4 : Correlation between fp and cfsu

The correlation coefficient is 0,99 and the standard deviation in the difference in the Y-value is 59. Incorporating the Fetcke results into the data from Rabobank only influences the offset from –87 to –52, but has no effect on the gradient, the correlation coefficient and the standard deviation.

Conversion of maintenance function points to cfsu vice versa

At this point we do not have enough direct data to prove that maintenance function points [5] can be converted to CFFP counts for enhancement projects. Up to now we have to work with indirect data to support such an assumption. To do so we have compared the relationship between functional size and effort for both function points (new development projects) and maintenance function points (enhancement projects). The exact results cannot be disclosed because thay contain traceable productivity figures, but they give very similar gradients. This may indicate that function points and maintenance function points are similar quantities in relation to effort [7]. With CFFP there is one method for sizing both new developments and enhancement projects. If function points can be converted to cfsu and if function points and maintenance function points are similar quantities in relation to effort there is a good ground for the assumption that maintenance function points can be converted to cfsu.

For enhancement projects the ILF and EIF have little influence on the size of a project. Since we assume that the offset in the conversion formula is a result of the influence of the ILF and EIF on the project size, we believe that for converting maintenance function points to cfsu only the gradient of the conversion formula is relevant and the offset can be ignored. For enhancement projects the conversion formula thus becomes:

$$\mathbf{Y\ (cfsu) = 1{,}2\ X\ (mfp)}$$

Conversion of product delivery rate from function points to cfsu vice versa

We assume that a similar reasoning applies to the conversion of data for the product delivery rate (hours per cfsu). For the conversion of the product delivery rate only the gradient of the conversion formula is relevant and the offset may be ignored. For the product delivery rate the conversion formula reads as:

$$\mathbf{PDR_Y\ (hr\ /\ cfsu) = 0{,}83\ PDR_X\ (hr\ /\ fp)}$$

Using this conversion formula it is possible to benchmark projects sized with CFFP against projects sized with FPA which can be extracted from reference databases like ISBSG [8]. As long as there is no substantial set of reference data sized with CFFP this conversion formula is a working alternative. An initiative from COSMIC and ISBSG is underway to establish a CFFP reference database by the end of this year [9]. Results from Rabobank will be submitted to support this initiative.

Approximate version of COSMIC Full Function Points

To support early estimation an approximate version of the CFFP method can be used. In the Measurement Manual the process to do so is described in detail [4, chapter 7]. 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).

The approximate version gives an average value for the size of a functional process. In the very early stages of software development only the number of functional processes is known. 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 upun development of avionics of a military aircraft, the average size of a functional process is 8. From our data we have calculated the average size of a functional process to be 7,2.

In a later stage of the development process there is enough 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 extra checks
- large e.g. retrieval of information about multiple objects
- complex

These categories can be assigned average values by dividing the size of a number of projects into four quartiles and computing the average size of a functional process in each of the quartiles. In table 3 we present the data from the Measurement Manual and our own data.

Quartile	Avionics	Rabobank
small	3,9	4,0
medium	6,9	6,2
large	10,5	10,8
complex	23,7	24,7

Table 3 : Comparison of quartile averages

Although the fields which both sets of data come from could hardly be more different, the results are very similar. This might be an indication that these values have a more general applicability than was expected beforehand.

Conclusions

With a year of practical experience Rabobank is now using CFFP for administrative software as a replacement for FPA. For Rabobank it was some gamble as an early adopter in a field which is up to now largely dominated by FPA as a functional sizing method, but the first test has been passed. We are now at the point that we can say that for this environment CFFP works as well as FPA. The next test is the predictive value of CFFP for development projects that cannot be sized with FPA.

References

[1] Ton Dekkers & Frank Vogelezang (2003), "COSMIC Full Function Points: Additional to or replacing FPA", Proceedings of the European Systems and Software Engineering Process Group Conference 2003, London

[2] NESMA (1996), "Definitions and counting guidelines for the application of function point analysis A practical manual, version 2.0"

[3] Ton Dekkers & John Kammelar (2001), "A functional Sizing Meta Model", Proceedings 12th European Software Control and Metrics conference, London; Ton Dekkers & John Kammelar (2001), "A functional Sizing Meta Model", Proceedings 4th European conference on Software Measurement and ICT Control (FESMA - DASMA), Heidelberg

[4] COSMIC (2003), "Measurement Manual COSMIC Full Function Points 2.2, The COSMIC Implementation Guide for ISO/IEC 19761"

[5] NESMA (2001), "Function Point Analysis for Software Enhancement"

[6] Thomas Fetcke (1999), "The warehouse software portfolio, a case study in functional size measurement", Technical report no. 1999-20, Software engineering management research laboratory, Université du Quebec à Montréal, and Software engineering laboratory in applied metrics.

[7] Ton Dekkers (2003), "Functional size measurement methods are also applicable in enhancement projects", Proceedings of the 13th International Workshop on Software Measurement, Montreal

[8] ISBSG (2003), The ISBSG Estimation, Benchmarking & Research Suite", release 8

[9] COSMIC (2003), "A once-off Invitation from COSMIC and ISBSG Help establish the foundation COSMIC-FFP Benchmarks", http://www.cosmicon.com

Authors

Frank Vogelezang has been working as a practitioner and consultant within the area of software metrics for more than four years. Within this area he specialises in estimation and performance measurement within client organisations. Frank Vogelezang is a quality consultant for the Software Control division of Sogeti Nederland B.V. He is responsible for the Metrics Expertise Centre of Rabobank IBS. He is a member of the Measurement Practices Committee of COSMIC and a member of the COSMIC working group of NESMA.

Arlan Lesterhuis has been working as a consultant within the area of software metrics and testing for over twenty years. Arlan Lesterhuis is test consultant for the Software Control division of Sogeti Nederland B.V. He is responsible for the mathematical methods used within de Metrics Expertise Centre of Rabobank IBS. He is a major contribuant to the finetuning of the COSMIC Measurement Manual.