



# Towards an accurate functional size measurement procedure for conceptual models in an MDA environment

Beatriz Marín<sup>a,\*</sup>, Oscar Pastor<sup>a</sup>, Alain Abran<sup>b</sup>

<sup>a</sup> Centro de Investigación en Métodos de Producción de Software, Universidad Politécnica Valencia, Camino de Vera s/n, 46022 Valencia, Spain

<sup>b</sup> Department of Software Engineering and Information Technology, École de technologie supérieure, Université du Québec, 1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

## ARTICLE INFO

### Article history:

Received 13 October 2008

Received in revised form 27 December 2009

Accepted 5 January 2010

Available online 21 January 2010

### Keywords:

Conceptual modeling

Object orientation

Functional size measurement

COSMIC

MDA

## ABSTRACT

The accurate measurement of the functional size of applications that are automatically generated in MDA environments is a challenge for the software development industry. This paper introduces the OO-Method COSMIC Function Points (OOMCFP) procedure, which has been systematically designed to measure the functional size of object-oriented applications generated from their conceptual models by means of model transformations. The OOMCFP procedure is structured in three phases: a strategy phase, a mapping phase, and a measurement phase. Finally, a case study is presented to illustrate the use of OOMCFP, as well as an analysis of the results obtained.

© 2010 Elsevier B.V. All rights reserved.

## 1. Introduction

The MDA [50,51] approach separates application and business logic from the platform technology, allowing code generation by means of model transformations. In such MDA contexts, the software production processes are based on conceptual models that are used as inputs to the code generation process. These conceptual models must have the required semantic formalization to specify all the functionality of the generated application and to avoid different interpretations for the same model. The adoption of MDA-based technology has presented new challenges, such as the capability of measuring the size of the products generated in the early phases of the software development.

The Functional Size is defined as the size of the software that is derived by quantifying the functional user requirements<sup>1</sup> [32]. The functional size measurement (FSM) was defined in the late 1970's mainly through the Function Point Analysis (FPA) proposal [9]; later, FPA was adapted to object-oriented models aligned with the UML standard [41,61–63]. However, these FPA-based approaches present limitations for the measurement of the functional size of conceptual models in MDA environments. Even though we found approaches that measure the functional size in MDA environments in the literature [4,1], these approaches are also FPA-based and have limitations, too. One such limitation is that FPA approaches only allow the measurement of the functionality that the human user sees, ignoring all the functionality that is needed for the correct operation of the application (which must be built by the developer, even though it is not seen by the human user). Another significant limitation is that the size of any elementary process (an elementary unit of functional user requirements [32]) within a model is limited to

\* Corresponding author. Tel.: +34 96 387 73 50; fax: +34 96 387 73 59.

E-mail addresses: [bmarin@pros.upv.es](mailto:bmarin@pros.upv.es) (B. Marín), [opastor@pros.upv.es](mailto:opastor@pros.upv.es) (O. Pastor), [abran.alain@etsmtl.ca](mailto:abran.alain@etsmtl.ca) (A. Abran).

<sup>1</sup> The functional user requirements are defined in the ISO 14143-1 as a sub-set of the user requirements. The functional user requirements represent the user practices and procedures that the software must perform to fulfil the users' needs. They exclude Quality Requirements and any Technical Requirements.

only three intervals of classifications for DET (Data Element Types), RET (Record Element Types) or FTR (File Types Referenced). These intervals assign a specific functional size for the elementary process in the complexity tables of IFPUG FPA [27,28]. Therefore, the functional size of a model will not vary even if some elementary processes have a very large number of DET, RET or FTR [21,39,40].

To overcome the limitations of the initial design of the FPA measurement method, the COSMIC measurement method [7] was defined in the late 1990s as the second generation of the functional size measurement method. It has been adopted as an international standard: ISO 19761 [34]. One of the advantages of COSMIC over FPA is that it allows the measurement of the functional size from different points of view, for instance, the functionality that the users see and the functionality that the developer has to build. Another advantage is that COSMIC uses a mathematical function that is not limited by maximum values to measure the size of conceptual models. This mathematical function aggregates the functional size of the functional processes (an elementary unit of functional user requirements [5]) that are specified in the conceptual models. This helps to better distinguish the size of large conceptual models. Another advantage of COSMIC is that it allows the measurement of applications that are generated in layers, which permits the measurement of the whole application or the measurement of each layer of the application.

Currently, there are some approaches that apply COSMIC for the purpose of estimating the functional size of future software applications from high-level specifications [36,23]. As the functionality to be measured using these proposals is not sufficiently detailed to generate the final application automatically in MDA environments, some FSM procedures have been designed to measure the functional size of the application from its conceptual models, such as Diab's proposal [16] and Poels' proposal [57]. Both FSM procedures were defined by establishing a mapping between the COSMIC concepts and the primitives of the Real-Time Object Oriented Modeling (ROOM) specification [59] or the primitives of the MERODE conceptual models [15], respectively. The Diab and Poels proposals are based on conceptual models that allow the generation of the applications, but that do not allow the generation of fully working applications from these conceptual models. This generates a traceability problem between the generated application and the conceptual model. Moreover, the conceptual models of these proposals do not have enough expressiveness for the total specification of the application in an abstract way. Therefore, these procedures do not allow the accurate<sup>2</sup> [30] measure of the functional size of the applications generated from the conceptual models.

To avoid this problem, we have selected the OO-Method approach [54]. This approach is an object-oriented method that is based on model transformations. It provides the semantic formalization needed to define complete and unambiguous conceptual models, allowing the automatic generation of software products using an MDA-based technology. Although presentation and interaction modeling is accepted to constitute an essential part of any software development method, it is remarkable that most of the current model-based approaches do not include an explicit presentation model. OO-Method does this, and, consequently, a complete view of the software product to be built from the model is provided at the conceptual level. This method has been implemented in a suite of industrial tools by the company CARE Technologies [13] (a software development company). Currently, CARE Technologies has a procedure to measure the functional size of conceptual models [4], but this procedure is limited to the IFPUG FPA with the limitations explained above.

In general, the Functional Size of applications is used to successfully apply prediction models [4,63], such as effort and budget models. In an MDA context, the effort of programming the applications is substituted by a modeling effort. Thus, in this context, dealing with the cost of the applications should mean dealing with how to measure cost from the involved models. Therefore, it is very important to know the accurate functional size of the conceptual models that are built with these MDA-based approaches since this functional size is necessary to adjust the budget in order to estimate the cost of the software product that is generated automatically.

In addition, since the functional size measurement is independent of the technological platform, it allows the generation of indicators in early stages of the software development cycle. Thus, knowing the accurate functional size of software from the conceptual models is critical for evaluating risks and for obtaining early project indicators (such as productivity). Moreover, given that a functional size measurement procedure analyzes all the elements of the conceptual model that fulfil the functional user requirements, the measurement procedure can be used as a very valuable tool to identify defects in the conceptual models.

The contribution of this work is the systematic design and application of an FSM procedure that measures the accurate size of the generated applications from their conceptual models. Thus, this paper presents the OOmCFP proposal, which is a procedure based on COSMIC to measure the functional size of the OO-Method applications from their conceptual models. The holistic view of the OO-Method conceptual model (including the structure of the system, the behavior, the interaction between the users and the system, and the interaction among the components of the system) allows OOmCFP to take into account all the functionality of the generated applications, including the functionality related to the presentation features that other measurement procedures do not consider. Therefore, we infer that OOmCFP contributes substantially to improving the measurement of applications generated from conceptual models.

The rest of the paper is organized as follows: Section 2 presents the main concepts of the COSMIC measurement method, the related work, the OO-Method approach, and a process model for software measurement. Section 3 presents the design of

<sup>2</sup> The accuracy of measurement is defined in the International Vocabulary of Basic and General Terms in Metrology (VIM) as closeness of agreement between a quantity value obtained by measurement and the true value of the measurand.

Download English Version:

<https://daneshyari.com/en/article/379048>

Download Persian Version:

<https://daneshyari.com/article/379048>

[Daneshyari.com](https://daneshyari.com)