A COSMIC Measurement Procedure for BPMN Diagrams

Beatriz Marín, José Quinteros
Facultad de Ingeniería
Universidad Diego Portales
Ejército 441, Santiago, Chile
{beatriz.marin, jose.quinteros}@mail.udp.cl

Abstract—In recent years, the estimation of effort, budget and time has been a fundamental step in the software project life cycle. Having this knowledge it is possible to manage and control a software project. So that, it is important to have these estimations as early as possible in order to produce a sound schedule for the project. Taking into account that BPMN diagrams are widely used in industry for analysis, in this paper we present a functional size measurement procedure based in the COSMIC method that measure these diagrams. With this size it is possible to obtain estimations at early stages of software cycle.

Keywords—BPMN; COSMIC; Estimation, Functional Size; Functional Size Measurement; Business Process

I. INTRODUCTION

In the last years, the estimation of effort, budget and time has become an important input in the software project management [1], since having this knowledge it is possible to better planning and control a software project. Thus, it is important to have these estimations as early as possible in order to produce a sound schedule for the project.

Nowadays, there are several methods that allows the estimation of budget, effort, and time; which are mainly based on COCOMO [2]. These methods present late estimations since they used lines of code (LOC) as input in the measurement process [3], resulting in measures that are dependents of the programming language. Other estimation methods are based on Function Points Analysis (FPA) [4]. FPA proposes the functional size measurement, which use design artifacts as input in the measurement process, so that FPA-based methods provide more early measurements than COCOMO-based methods, and also FPA-based methods are independent of the programming language [5].

To measure the functional size of software applications, there are five standard measurement methods [6], [7], [8], [9] and [10]. These methods have been illustrated in the measurement of the functional size of final applications. However, project leaders need indicators in early stages of software cycle for a better management of software projects. Thus, it is necessary to define how the standards can be applied to the diagrams that are used in early stages of software cycle. A measurement procedure corresponds precisely to this specification [9].

We have selected the COSMIC measurement method to specify a measurement procedure since, in contrast to other standard functional size measurement methods, COSMIC can be applied to any type of software (including MIS, real-time software, embedded software, and hybrid software), and it also allows the measurement of multi-layer applications.

Taking into account that BPMN diagrams [11] are widely used in industry for analysis, in this paper we present a functional size measurement procedure based in COSMIC that measure the functional size of BPMN diagrams. With this size, project leaders can calculate productivity indicators, the price to be charged to clients, effort [12], etc.

The rest of the paper is organized as follows: Section 2 presents the background of this work. Section 3 presents the design of a measurement procedure. Section 4 presents the automation of the procedure and its application to an example. Finally, Section 5 presents some conclusions and further work.

II. BACKGROUND

This section presents the main characteristics of BPMN and COSMIC, and some relevant related works.

A. BPMN

The Business Process Model and Notation (BPMN) [11] was recognized as standar by OMG in 2011. The main goal of BPMN is to provide a notation that will be understandable by all business users, such as the analysts that creates the processes, the technical developers that provides the technical platform for these processes, and the business people who will manage and will monitor these processes. Thus, the main advantage of using BPMN is that it creates a standardized bridge between the design and execution of business processes.

The main elements of BPMN diagrams are the following:

- Flow objects are used to define the behaviour of business process. They can be events (start, end), activities (processes, tasks), or gateways.
- Data is represented with four elements: data objects, data inputs, data outputs, and data stores.
- There are four connecting objects that allow the connection among flow objects and other information: sequence flows, message flows, associations, and data associations.
- There are two ways of grouping the primary modeling elements through swimlanes: pools and lanes.
• Artifacts are used to provide additional information about the process. They are groups and text annotations.

B. COSMIC

The COSMIC measurement method version 3.0.1 [13] has three phases: the measurement strategy phase, the mapping phase, and the measurement phase.

In the strategy phase, the purpose, the scope and the level of granularity of the measurement must be determined. In addition, functional users must be identified. Functional users are defined as a type of user that send or receive data to/from the functional processes of a piece of software.

In the mapping phase, functional processes and the data groups involved in these functional processes must be identified. A functional process is a basic component of the set of requirements that describe what the software shall do. A functional process comprises a single, cohesive, and independent set of data movements. An optional step of this phase is the identification of data attributes that are related to the data groups.

In the measurement phase, the identification of data movements must be carried out, and the measurement function must be applied. A data movement moves one or more types of data attributes belonging to a only one type of data group. There are four types of data movements, which are defined as:

- **Entry**: It is a type of data movement that moves a data group from the functional user across the boundary within the functional process that is required.
- **Exit**: It is a type of data movement that moves a data group from a functional process across the boundary to the functional user that needs it.
- **Read**: It is a type of data movement that moves a data group from the storage to the functional process that requires it.
- **Write**: It is a type of data movement that moves a data located in a functional process inside of a storage.

The measurement function assigns 1 CFP (COSMIC Function Point) to each data movement. Then, the results are aggregated in order to obtain the functional size of the software measured.

C. Related works

There are some works that measure the functional size of software from business models. In [14] an approach for measuring the functional size of ERP systems is presented by applying IFPUG FPA to business models. Therefore, in contrast to approaches based in COSMIC, it has the limitation related to the precision of measurements of IFPUG FPA.

In [15], a measurement approach based on COSMIC was presented. This approach uses as input UML use-case, activity and class diagrams. However, it does not present a mapping among business and COSMIC concepts, so that it is difficult to apply to BPMN diagrams.

A recent COSMIC measurement procedure for BPMN diagrams was presented in [16]. It presents rules that map two different concepts of COSMIC to the same concept of BPMN, which can produce ambiguities. The application of this work is manual, which is an error-prone and time consuming task.

III. DESIGN OF E-FSMP

This section presents E-FSMP, meaning Early Functional Size Measurement Procedure, which is a procedure based in COSMIC that allows the measurement of the functional size of BPMN diagrams. Since COSMIC is comprised of three phases, the E-FSMP is also comprised of three phases.

A. Measurement strategy phase

In this phase, the strategy of the effort of measurement is defined by the identification of the purpose, the scope, the level of granularity, and the functional users.

The **purpose** of E-FSMP has been defined as: To measure the functional size of the requirements specified in BPMN diagrams. The **scope** corresponds to the BPMN diagrams contractually agree as the requirements.

The **level of granularity** is related to the requirements and the natural evolving of some requirements specified in the software artifacts to be measured. The standard level of granularity is as at which individual functional processes have been identified and their data movements identified.

The **functional users** correspond to any type of user that interacts with the software, which is specified in the FUR. In BPMN diagrams, these users correspond to the role defined to execute the processes in the intended software system. These users are functional users because they send (or receive) data to (from) these processes.

The **boundary** is defined as a conceptual interface between the software being measured and its functional users. In BPMN diagrams, this boundary corresponds to the pool. To avoid mistakes in the identification of the functional users and the boundaries of a BPMN diagram, Table 1 shows the rules that have been defined to identify the functional users (Rule 1) and the boundary (Rule 2).

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 1</td>
<td>Identify 1 functional user for each role in the BPMN diagram.</td>
</tr>
<tr>
<td>Rule 2</td>
<td>Identify 1 boundary between a functional user and a pool of a BPMN diagram.</td>
</tr>
</tbody>
</table>

B. Mapping phase

When the measurement strategy has been established, the mapping phase is carried out. In this phase, the functional processes and data groups must be identified.

A **functional process** is a set of functionalities of the software system that allows the achievement of a functional requirement. Generally, in BPMN diagrams, the functional requirements are presented as activities. Thus, the activities of the BPMN diagram are considered as functional processes.

However, the functional processes can be identified several times if they are accessed from more than one role. To avoid duplicity, each activity is identified as a functional process only for the first role that access it.
Later, the **data groups** that participate in the functional processes must be identified. Data groups correspond to the incoming and outgoing documents of each activity of the BPMN diagram. Table 2 shows the rules that have been defined to identify the functional processes (Rule 3), to avoid duplicity in the identification of the functional processes (Rule 4) and to identify data groups (Rule 5).

**TABLE II.** E-FSMP MAPPING RULES

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 3</td>
<td>Identify 1 functional process for each activity that can be accessed by a role of the BPMN diagram.</td>
</tr>
<tr>
<td>Rule 4</td>
<td>Identify an activity as a functional process only once.</td>
</tr>
<tr>
<td>Rule 5</td>
<td>Identify 1 data group for each data input or data output of the activities specified in the BPMN diagram.</td>
</tr>
</tbody>
</table>

**C. Measurement phase**

In this phase, data movements of each functional process are identified. Then, the measurement function is applied, and the results are aggregated to obtain the functional size of each functional process, which later are aggregated to obtain the functional size of software.

Each functional process move two or more data movements. Each **data movement** moves one data group. Data movements can be entry, exit, read, and write. An entry data movement corresponds to the movement of an input data object to an activity. An exit data movement corresponds to the movement of an output data object from an activity. A read data movement corresponds to a movement of a data object from the database to the scope of an activity. A write data movement corresponds to a movement from an activity to the database.

When all the data movements have been identified, the **measurement function** is applied. This function assigns 1 CFP to each data movement identified. Once the measurement function has been applied, the measures can be aggregated to obtain the functional size of each functional process of the application as well as the whole application. Table 3 shows the rules defined for the identification of data movements (Rule 6, 7, 8, and 9); and the measurement rules (Rule 10 and 11).

**TABLE III.** E-FSMP MEASUREMENT RULES

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule 6</td>
<td>Identify 1 entry data movement for an input data object to an activity.</td>
</tr>
<tr>
<td>Rule 7</td>
<td>Identify 1 exit data movement for an output data object from an activity.</td>
</tr>
<tr>
<td>Rule 8</td>
<td>Identify 1 read data movement for each different data object from the database to an activity.</td>
</tr>
<tr>
<td>Rule 9</td>
<td>Identify 1 write data movement for each different data object that goes from an activity to the database.</td>
</tr>
<tr>
<td>Rule 10</td>
<td>Aggregate the CFP related to the data movements identified in each activity to obtain the functional size of that process.</td>
</tr>
<tr>
<td>Rule 11</td>
<td>Aggregate the CFP related to the functional processes identified to obtain the functional size of the software.</td>
</tr>
</tbody>
</table>

With the 11 rules presented it is possible to measure the functional size of software from their BPMN diagrams by the application of COSMIC FSM method.

**IV. AUTOMATION OF E-FSMP**

The manual measurement of functional size is generally very time-consuming and has many precision errors. For this reason, it is necessary to automate the measurement process to obtain a solution that can be applied in industrial contexts.

The tool that implements the E-FSMP was developed using Laravel [17], which is a PHP framework to develop web applications. This tool allows the measurement of BPMN or use case diagrams (see Fig. 1). The first step using the E-FSMP tool is to charge the XML file with the specification of a BPMN diagram. This XML file was automatically created by Bizagi tool from the BPMN specification.

Fig. 1. Home page of E-FSMP tool.

Then, the tool apply the rules defined by E-FSMP in order to obtain the functional size. To do this, the tool first verify the structure of the XML file. If the file does not correspond to a BPMN diagram, then the tool shows an error message. If the XML file corresponds to an specification of a BPMN diagram, the tool identify tasks in the XML description, and then, it identify data objects (inputs and outputs) and databases.

Later, the E-FSMP tool assigns 1 CFP to each data movement identified. Finally, the tool aggregates the results in order to obtain the functional size of the software that is measured.

Fig. 2 shows an example of BPMN diagram. This diagram correspond to the requirements specification for the sent process of a post office. The system is called SIGECO (Post Office Management System), which supports sent and reception of postal packages. Regarding the sent process (see Fig. 2), sender goes to the post office and requests to send a product. The post officer verify the product and the shipment data, and then register the shipment. Then, sender receives a code that will use to track the product. Later the post officer send the product, which is received in a different post office, which is responsible of the delivery process to carry out the product to the final receptor.

This example was manually measured applying the E-FSMP obtaining 24 CFP. Finally, the xml representation of the sent process of SIGECO was measured using the E-FSMP tool. The functional size obtained by the E-FSMP tool corresponds to 24 CFP, which is the same size obtained manually. The report of the E-FSMP tool shows the functional size measured, the tasks recognized as functional processes, and the data movements that can occur in each functional process.
V. CONCLUSIONS AND FURTHER WORK

In this paper, a functional size measurement procedure called E-FSMP has been presented. E-FSMP applies the standard COSMIC measurement method to BPMN diagrams by means of the application of 11 rules.

In addition, a tool that automates the application of E-FSMP has been presented. With this tool, measurement results can be obtained quickly and avoiding precision errors that commonly occur with manual measurements. Thus, the main contribution of this work is the specification of E-FSMP and the tool that automates the measurement of functional size at early stages in software development.

Further work includes the execution of real case studies that allows the validation of E-FSMP and verification of the scalability and the performance of the E-FSMP tool.

ACKNOWLEDGMENT

This work was funded by FONDECYT project TESTMODE (Ref. 11121395, 2012-2015).

REFERENCES