Automated Size Measurement of Embedded System based on XML using COSMIC FSM

Thandar Zaw, Swe Zin Hlaing, Myint Myint Lwin, and Koichiro Ochimizu

University of Information Technology, Myanmar

thandarzaw@uit.edu.mm,swezin@uit.edu.mm, myintmyintlwin@uit.edu.mm, ochimizu@jaist.ac.jp

Abstract. Nowadays, function size is an important role for measuring software before the development phase in software development life cycle. Industries need well-defined different notations and measurement method. Thus, COSMIC FSM is one of the International Standards of FSM for measuring the size of embedded software. Manual measurement is time consuming and complicated to measure the size of large system. This paper proposes the automated measurement tool that accepts the XML documents from three different notations such as the UML, SysML and Petri net. Then, the XML documents of these diagrams have been translated to COSMIC FSM by using mapping rules with the case study of cooker system. This paper shows the result of software size by using the prototype tool that is the same with the result of manual measurement.

Keywords Common Software Measurement International Consortium (COSMIC FSM), Sequence diagram, UML, SysML, Petri net

1 Introduction

Software size estimation is an important input for estimating the effort required to develop the measured software. Software sizing is the prediction of the software size to build the system. Several size measurement methods have been proposed to measure the size of software. The most popular size estimation methods are source line of code (SLOC) and functional size measurement (FSM). Size-related measure is based on some output from the software process and it is language dependence. And function-related measure such as functional measurement (FSM) is based on an estimate of the functionality of the delivered software. Function size measurement methods are technology independent and can be estimated early in analysis and design.

Functional Size Measurement (FSM) is a great method in industrial development as it supports the input to estimate the effort. Software functional size measurement methods that have been recognized as various ISO FSM: IFPUG FPA, MKII, NESMA, FISMA, and COSMIC FSM[1]. The other traditional methods are applied only in business software but they are hardly applied in real-time software and embedded software. But COSMIC can be applied in both software.

Many researchers proposed the software estimation methods which are not applicable for various modeling notations. To address this limitation, this paper proposed the automated software size measurement tool for different modeling notations to estimate the functional size of software. These notations have been translated to COSMIC by using mapping rules with a simple case study of cooker system. This paper is organized as follows: the second section provides related work; the third section presents the proposed system; the fourth section explains the evaluation and final section is conclusion and future work.

2 Related Work

In [2], Symons, C. described the COSMIC concepts that can be applied in any real-time software requirements to measure the functional size of real-time software to understand clearly for any software engineer with alarm example. In [3], Soubra, H., et al. proposed the design of the FSM procedure based on the documentation of the mapping of the Simulink concepts to COSMIC concepts for the embedded real-time software system. In [5], Luigi Lavazza., et al proposed the UML that can be used to build models according to the COSMIC measurement rules. Asma Sellami et al. [6] proposed the measurement method for sizing of sequence diagram that can be measured both the functional and structural size at different level of granularity. In [4], the author proposed the automated functional and structural measurement of software size from XML structure of sequence diagram to calculate COSMIC CFP. This paper proposed the automated FSM tool which is based on the XML documents from different design model notations. Then, the mapping rules are defined between these notations and COSMIC. It is expected that the result of the functional size of software can be provided by many industries to increase effort and productivity.

3 Proposed System

This section describes the proposed method to calculate the functional size of software by using the XML structure of three different notations. The proposed system of three phases is shown in Fig 1. In measurement strategy, the proposed system is analyzed XML document from the popular notations such as UML and SysML sequence diagram and Petri net. In mapping phase, after analyzing the design models with COSMIC concepts, the design models have been translated into COSMIC by using the mapping rules. In measurement phase, the actual size of software is calculated by using COSMIC method.

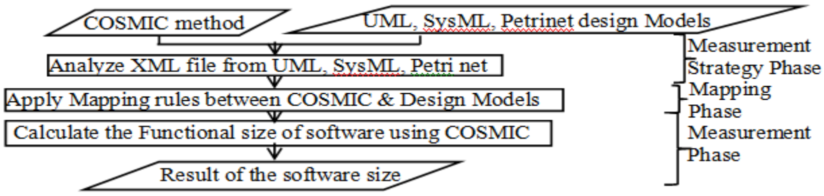

Fig. 1. Proposed System based on different design notations

3.1 UML Use Case Model of Cooker System as Functional User Requirements

This paper discusses the specification of a simple version of the cooker system which is used as a case study to determine the counting of COSMIC [6]. Before developing the UML, SysML and Petri net representation, the specification of the cooker system must be defined. The functional user requirements of this system are as follows:

1. The cooker software can get the input from a door sensor and start button. Then, it can show the light and heater on/off when the power is switched on.
2. When the start button is pressed and the door is closed, the cooking starts. If the door is open, the start button has no effect.
3. Either the door is open while the cooking is in progress or when cooking is completed, the timer signals will stop.

The use case diagram is illustrated for the common process of the cooker system as shown in Fig. 2. This system consists of two main functionalities: Start Cooking and End Cooking. The functional users of the input side of the cooker system are DoorSensor, Start Button and the output sides of the functional users are Light and Heater. Timer is on both sides of functional user.

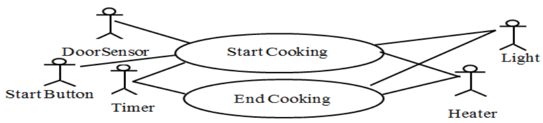

Fig. 2. Use case diagram of Cooker System

3.2 COSMIC FSM in UML , SysML Sequence Diagram and Petri net

The measurement process of cooker system mainly comes from the basic functional requirements. In this system, there are two functional processes of UML and SysML sequence diagrams for cooker system as shown in Fig. 3 and 4. In Fig. 3, the cooker checks that the door is open or closed. When the door is closed, it sends the signals to heater, light and timer for one minute of cooking. The XML structure of UML and SysML sequence diagram is extracted from Papyrus and several related tags to message notation is identified and analyzed to calculate CFP. SysML borrowed from UML are reused without modification. So, the concepts and features of SysML are the same as UML. The <message> tag is an important to calculate the data movements based on COSMIC method. There are several types of message. In the XML structure, “Cooker Software” is defined as the boundary. If “messageSort” attribute in <message> tag is “asynchCall”, it will identify Entry and Exit data movements. If it is synchronous call and move the data between the Cooker Software and other object, it will identify Read and Write data movements. Entry data movement is defined by extracting the value in “coveredBy” attribute in< lifeline> tag that is matched from value in “receiveEvent” attribute in <message> tag and value in “xmi:id” attribute in fragment tag. Exit data movement is also defined by extracting the value in “coveredBy” attribute in <lifeline> tag that is matched from value in “sendEvent” attribute in <message> tag and value in “xmi:id” attribute in <fragment> tag.

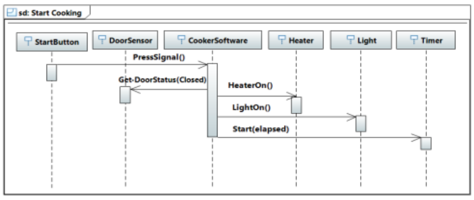

Fig. 3. Start Cooking of UML and SysML

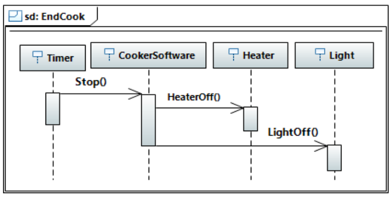

Fig. 4. End Cooking of UML and SysML

The XML structure of Petri net is extracted from HiPS. Places are drawn as ellipses and held multi-sets (bags) of tokens. A place models a local state given by its tokens. Transitions model behaviors are drawn as boxes. A transition is connected to input places and output places by arcs. Each data movement corresponds to the transition. The <transition> is a key to define the data movements. The Petri net diagrams of cooker system are shown in Fig.5 and Fig.6.

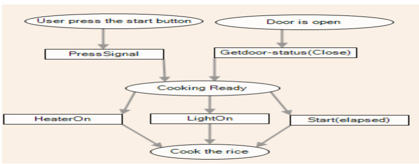

Fig.5. Start Cooking in Petri net

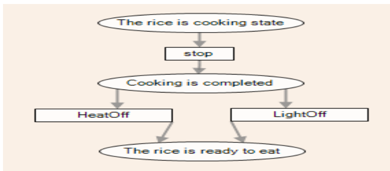

Fig. 6. End Cooking in Petri net

3.3 Mapping Phase

In this section, the key concepts of COSMIC are mapped to the different notations such as: UML, SysML and Petri net. The mapping rules are described as follows:

Rule 1: The boundary represents in use case diagram. It shows the application border that is established by identifying the external elements and application system.

Rule 2: The functional user is an active object that does not stop until either the complete behaviour is executed or the object is terminated by some external object.

Rule 3: The functional process identifies use cases in the system.

Rule 4: The data groups identify the trigger event that carries data between objects.

Rule 5: The four data movements are identified as follows:

Rule 5.1: The Entry data movement identifies from Rule 2 to Rule 1.

Rule 5.2: The Exit data movement identifies from Rule 1 to Rule 2.
Rule 5.3: The Read/Write data movement identify messages that send into or out of the internal persistent storage.
Rule 6: Apply the COSMIC measurement function.
According to COSMIC measurement, each of the data movement in each functional process is added to get the functional size of that process.
Rule 7: Aggregate the functional size measurements.
Aggregate all of the data movements of the functional processes of the whole system into a single functional size value to obtain the functional size of the system.

3.4 Measurement Phase

After defining the mapping rules, the data movements of each functional process have been identified as shown in Fig. 3 to 6. According to COSMIC standard, 1CFP is defined as the size of one data movement. In the functional process of Cook End, it has 1 Entry data movement that calculated the Stop attribute from timer. It also identified 2 Exit data movements which also counted the HeaterOff attribute to Heater, the LightOff attribute to Light respectively. The subtotal of functional size for that functional process is 3CFP. The total size from each function is 8CFP by adding all number of data movements. The data movement of each sequence diagram in this system is as shown in Table 1.

Table 1. Measurement of data movements for cooker system

Process	Message	Functional User	Data Movement	CFP
Start Cooking	PressSignal()	Start button	Entry	5 CFP
	Getdoor-status(Close)	Door Sensor	Entry	
	HeatOn()	Heater	Exit	
	LightOn()	Light	Exit	
	StartCooking()	Timer	Exit	
End Cooking	Stop()	Timer	Entry	3 CFP
	HeatOff()	Heater	Exit	
	LightOff()	Light	Exit	
			Total Size	8 CFP

3.5 Automated Measurement for UML, SysML and Petri net

The prototype tool for the automatic measurement of the functions is developed as an example in Java. The measurement result of the cooker system is shown in Fig7.

4 Evaluation

The automated function size measurement for cooker system is successfully implemented in the three different notations. The manual measurement of FSM is time consumption because different measurers need to measure the manual count of

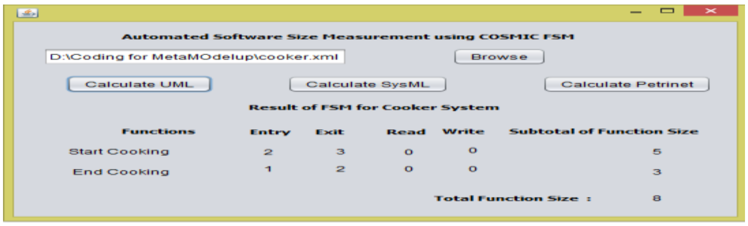

Fig. 7. Automated Functional Size measurement results of Cooker System

software size correctly. But the measurers who are not experts use the automated measurement of FSM to get more accurate size of software. The result of the manual measurement of case study is the same result with the automated result.

5 Conclusion and Future Work

Software function size can be measured throughout the design phase in the software development process. Like the other FSM methods, COSMIC FSM is the estimation method for measuring the size of embedded software. In this paper, the proposed tool applied the mapping rules between the XML structure from three different notations and COSMIC FSM to estimate the functional size of software. The automated tool contains the COSMIC rules for a simple case study of cooker system and helps to estimate the effort in the early stages of development. It has been intended to propose the large case studies by extending the COSMIC rules.

6 References

1. ISO 19761, Software Engineering – COSMIC : A Functional Size Measurement Method, International Organization for Standardization, ISO, Geneva, 2011.
2. Symons, C.: Sizing and Estimating for Real-time Software – the COSMIC-FFP method. In: DOD Software Tech News', Editor: Data & Analysis Center for Software, USA DOD, Rome NY, vol. 9(3), pp. 5–11 (2006).
3. Soubra, H., Abran, A. , Stern, S. , Ramdan-Cherif, A., "Design of a Functional Size Measurement Procedure for Real-Time Embedded Software Requirements Expressed using the Simulink Model", IWSM-MENSURA, 2011.
4. Meiliana etal. ,"Automating Functional and Structural Software Size Measurement based on XML Structure of UML Sequence Diagram ", 2017 IEEE International Conference on Cybernetics and Computational Intelligence 20-22 Nov. 2017.
5. Luigi Lavazza and Vieri Del Bianco, "A Case Study in COSMIC Functional Size Measurement: the Rice Cooker Revisited", IWSM/Mensura 2009.
6. A. Sellami. etal, "A measurement method for sizing the structure of UML sequence diagrams", Information and Software Technology 59, 2015.