A survey of known experience of using COSMIC functional size measurement for Real-Time software

October 2017

Real-time software and functional size

The use of functional size as a measure to make like-for-like comparisons between different software development or maintenance contracts has been common practice in administrative software for decades. In real-time software this practice is now developing. One of the reasons for these different practices in contracting, benchmarking and improvement programs is due to the fact that a lot of so-called first generation functional size measurement methods were difficult to implement in the real-time software environment [1]. This is mainly due to the fact that most of these methods largely rely on the presence of data structures in the software, while transactions are the dominant characteristic of real-time software. This document describes a number of public examples where the COSMIC method has been used successfully in the real-time software domain.

Background: The COSMIC Method

The COSMIC method of measuring a size of software from its requirements is the only ISO standard method designed to be able to measure:

- real-time and infrastructure software, as well as business application software;
- software requirements at the component or user story level, and/or at intermediate levels, e.g. an iteration, and to seamlessly aggregate these sizes up to the size of a whole application according to well-defined aggregation rules;
- approximate sizes early in a project's life-cycle before requirements are known in detail, using a variety of well-defined methods.

Sizes measured in units of COSMIC Function Points (CFP) depend only on the requirements; they are totally independent of the technology or processes used to develop the software.

The method has been successfully applied to measure real-time software in the aerospace, automotive, security and telecoms domains. Data have been published on various studies showing an excellent correlation of CFP sizes with development effort, resulting in regression curves that can be used to estimate effort. In the case of embedded software, CFP sizes have also been used to estimate the required hardware memory and the processor load. CFP size measurement has been automated for several situations, e.g. from specifications and from executing code.

This paper summarises cases of organizations producing real-time software that have published papers describing their experience, or that are known by the contributors to have used CFP sizing.

There may well be many other organizations using CFP sizing in these software domains but the COSMIC organization has no means of knowing the extent of the method's usage. COSMIC is a not-for-profit, voluntary organization with no formal membership. Many organizations prefer to keep their software project performance data private, so we never hear about them. Further, although we know of specific studies, we do not know in most cases if the organizations concerned continue to use the method.

All COSMIC documentation is available for free download from cosmic-sizing.org. This documentation includes guidelines on applying the method for real-time software and in agile environments.

COSMIC use in Aerospace and Avionics software

BAE SYSTEMS

In the year 2000, when the COSMIC method was still in the very early stages of its development BAE SYSTEMS participated in the method's field trials. After initial training, BAE staff measured the avionics software of two main aircraft sub-systems, of approximate sizes 3000 and 8250 CFP, i.e. these were very large sub-systems. The data yielded some very interesting results. Notably, BAE measured the 'number of lines of algorithm' (or NOLA) associated with the movements of data for each 'functional process' (a software process triggered by an external event). The results showed that most data movements had few NOLA. These results provided strong evidence in support of the COSMIC method's design assumption that for most software the proportion of functionality devoted to *manipulating* data versus *moving* data (in and out of the software and to and from storage) is a constant. The count of the data movements therefore provides a good measure of functional size.

Eurocopter

In 2006, Eurocopter (now Airbus Helicopters) published an account of their work on applying CFP sizing to the software of the NH90 military helicopter [2]. They monitored 2-3 years of development of the 'application portfolio' on the mission computer and tracked the absolute change of CFP's from release to release. They also tracked effort (from requirements through to qualification test) and established productivity figures for three areas. Sizes were measured 'largely automatically' from requirements held in DOORS, a requirements definition and management tool.

Airbus

The UK subsidiary is investigating the application of virtual reality capability for use to help engineers in the aircraft design stage and for training maintenance engineers, etc. Sizes of enhancements to the standard VR environment are being measured in units of CFP. The aim of this work is to develop a cost-estimation method for VR developments and applications, taking CFP sizes as input. At least part of this work will be published in graduate student theses.

System Non-Functional Requirements (NFR)

Abran, Al Sarayreh and others published a series of papers demonstrating how CFP sizing could be applied to measure the functionality arising from system NFR such as for maintainability, portability, configuration, security, system operations, etc. [See 3 for an example of these papers]. The NFR were all taken from ECSS (European Co-operation for Space Standardization), ISO and IEEE standards. The work demonstrated that in many cases system NFR evolve into software functional requirements that can be measured using CFP sizing, in the same way as any other software functional requirements.

COSMIC use in Automotive software

Renault

Renault, with Nissan, is now one of the world's largest automotive suppliers. It has published its experience [4] of using CFP sizing to measure real-time, safety-critical software specifications held in the Matlab Simulink tool for the Electronic Control Units (ECU's) of its vehicles. ECU's control the engine, brakes, air-conditioning, lighting, etc. Individually, these may be small components, but there are 50 – 100 ECU's in a modern car; the size and complexity of these will increase extremely rapidly with increasing vehicle infotainment and for autonomous driving vehicles.

Renault produces the CFP sizes automatically from specifications in Simulink and has published the rules for this [5]. The data it gathers are used to control price/performance of ECU component suppliers and also to predict, at the design stage, the required ECU memory size. Saab (the Swedish car manufacturer, at the time part of GM) has also used CFP sizing to predict ECU memory sizes. Additionally, Renault has used CFP measurements to help predict processor load.

Some component manufacturers in Renault's supply chain are also using CFP sizing.

Other vehicle manufacturer

Another global vehicle manufacturer uses CFP sizing to estimate effort for the software maintenance requests for its ECU's and to control the price and performance of this work [6,7]. They use COSMIC, because it is the only method that can precisely measure small software maintenance changes.

COSMIC use in Telecom software

Alcatel and Nokia

These companies participated in the initial field trials of the COSMIC method. Nokia used the method extensively, e.g. to measure the packing density of functionality in its mobile phones [8].

China Mobile

Currently this is the world's largest operator of mobile telecoms networks. It is deploying CFP sizing across the whole group to help estimate the cost of a software as the basis for establishing projects and to control work on the maintenance of its software. The method is used in more than ten provinces to measure software size, in both real-time and business application domains.

EXFO

EXFO is a Canadian-based supplier of telecoms fault detection equipment. It mandates use of the COSMIC method to measure the software output at its development centres around the world.

NTT Japan

NTT from Japan published data from its telecoms software showing a good correlation of CFP sizes with development effort [9].

Turkcell

Turkcell operates fixed and mobile telecoms networks in Turkey and many nearby countries. It is one of the largest integrated communications and technology services companies in the region. It uses the COSMIC method in its software development outsourcing projects since 2015. This includes real-time software such as complex-event processing applications, real time charging applications, messaging applications, etc. Turkcell has published articles concerning selection, roll out, usage and automation and standardization of the method [10, 11, 12].

COSMIC use in Security software

Canadian supplier of security and surveillance software systems

This supplier to world-wide clients uses Agile/Scrum methods to handle requests for new or changed functionality. Each change request is broken down into 'tasks' which are allocated to iterations lasting from 3 to 6 weeks. Effort for each task is estimated using a 'Planning Poker' process in units of Story Points, which are then translated directly into estimated work-hours.

The sizes of 24 tasks within nine iterations were re-measured in units of CFP. CFP sizes correlated very well with actual task effort, very much better than the relationship between Story Points and effort [13]. The study concluded that COSMIC sizes can help understand software project performance and be used to build estimating models more accurately and without the subjectivity of Story Points.

Chinese National University of Defense Technology

It is noteworthy that the first Chinese translation of the COSMIC 'Measurement Manual', published in 2009 was made by staff from the National University of Defense Technology, Beijing. The translators worked in software engineering and in particular in software cost estimating. A COSMIC China Chapter was established in 2016 to deploy COSMIC methods. The COSMIC method will become a Chinese national standard soon.

Automated CFP sizing and Current Research

To embed the sizing into the software development process, automation is important. Automated CFP sizing has been achieved from specifications in tools such as Simulink and tools for holding UML specifications. Semi-automated sizing has been achieved for executing and static code [6].

Amongst these various automation efforts, two prototype COSMIC-based tools have been developed (available for free download [14]) to measure the size of aerospace real-time embedded software modelled using the SCADE commercial tool and the size of ECU application software designed following the AUTOSAR (AUTomotive Open System Architecture) standard.

It is noteworthy that much research currently underway involving measurements of functional size now seems to depend on using COSMIC FSM rather than 1st Generation FSM methods. As an example, a proposal has been made [15] to use CFP sizing in metrics that will help assess the test intensity and defect density of autonomous real-time testing. Such testing is needed in cases of the IoT, e.g. in autonomously-driven vehicles where the tests must be expanded in real-time to deal with interactions with other vehicles or objects. Measuring test intensity and predicting the remaining defect density for the expanded system in real-time is necessary to understand the effect of actions taken by the expanded system.

Acknowledgements

The information in this note has come from many sources, in particular Alain Abran, Selami Bagriyanik, Thomas Fehlmann, Dylan Ren, Hassan Soubra, Charles Symons, Sylvie Trudel, and Frank Vogelezang.

References

[1] thepriceofit.blogspot.com/the-first-generation-FSM

[2] 'COSMIC FFP: an application in a military avionics system', 2006. (Notes: 'FFP' has now been dropped from the name of the COSMIC method. This is an internal presentation, which we have permission to use.)

[3] 'A Standards-based Model for the Specification and Measurement of Maintainability Requirements', Al-Sarayreh, Khalid T.; Alain Abran and Juan J. Cuadrado-Gallego, Proceedings of the 22nd International Conference on Software Engineering and Knowledge Engineering (SEKE 2010), Redwood City, California, USA, July 2010, *(ISBN 1-891706-26-8).*

[4] 'Manage the automotive embedded software development cost & productivity with the automation of a Functional Size Measurement Method (COSMIC)', Alexandre Oriou, Eric Bronca, Boubker Bouzid, Olivier Guetta, Kevin Guillard, International Workshop on Software Measurement, 2014 Rotterdam.

[5] 'Verifying the Accuracy of Automation Tools for the Measurement of Software with COSMIC – ISO 19761 including an AUTOSAR-based Example and a Case Study', Hassan Soubra, Alain Abran, Amar Ramdane-Cherif, International Workshop on Software Measurement, 2014, Rotterdam.

[6] 'Software Projects Estimation & Control: the Versatility and Contributions of COSMIC Function Points' Alain Abran, Charles Symons, Christof Ebert, Hassan Soubra, Frank Vogelezang, ICEAA 2017, Professional development and Training Workshop, Portland, USA

[7] 'Functional size estimation technologies for software maintenance', Christof Ebert, Hassan Soubra. IEEE Software, November/December 2014.

[8] 'Defining measures for memory efficiency of the software in mobile terminals', Hannu Toivonen, International Workshop on Software Measurement, Magdeburg, Germany, October 2002.

[9] 'Validation of application results of COSMIC-FFP to switching systems', Shin-ichi Nagano, Ken-ichi Mase, Yasuo Watanabe, Takaichi Watahiki, Shigeru Nishiyama, NTT, Australian Conference on Software Metrics, 2001.

[10] 'Selection of a functional sizing methodology: A telecommunications company case study', Global Journal on Technology, (7), 2015.

[11] 'Benefits and Challenges of Measuring Software Size: Early Results in a Large Organization', Collegium of Economic Analysis Annals, (43), 2017.

[12] 'Automated COSMIC Function Point measurement using a requirements engineering ontology', Information and Software Technology, (72), 2016.

[13] 'Effort Estimation with Story Points and COSMIC Function Points - An Industry Case Study' Christophe Commeyne, Alain Abran, Rachida Djouab, Software Measurement News, Vol. 21, No. 1, 2016.

[14] estaca.fr

[15] 'Autonomous real-time software and systems testing – a Six Sigma approach', Thomas Fehlmann and Eberhard Kranich, International Workshop on Software Measurement, 2017, Gothenburg.