Measuring Software Costs: A New Perspective on a Recurring Problem

Diane Jamieson
Curtin Business School
Curtin University of Technology
Perth Western Australia
JamiesonD@cbs.curtin.edu.au

Kevin Vinsen
ADI Limited, IS3
Perth Western Australia
kevin.vinsen@adi-limited.com

Prof Guy Callender
Curtin Business School
Curtin University of Technology
Perth Western Australia
CallenderG@cbs.curtin.edu.au

Abstract

Regular outsourcing of software development signifies a need for a common understanding between procurement specialists, project managers and software engineers of individual system requirements costs, to facilitate accord on price.

Dependable cost estimates are usually required by customers long before detailed analysis and design activities would normally produce this information. A number of estimation techniques have evolved to produce cost information at an early stage in the software life-cycle, however estimation continues to contribute to budget blowouts.

Contemporary techniques for costing specifications described as use cases are increasingly challenged as the size and complexity of the system expands. In addition object-oriented representations of requirements fail to directly map into structures used by project managers, leading to ongoing comparisons of value that are subjective and often unrepresentative of final project expenditure. Flaws in the current application of cost estimation methodologies have been identified and a possible solution is proposed.

1. Current Interests

Curtin University of Technology, ADI Limited, Defence Materiel Organisation (DMO), Total Metrics, and other partners have applied to the Australian Research Council (ARC), as part of the November 2003 Linkage Project round, for funding for a project to take a multidisciplinary approach to the costing of software development. The results of this application are not yet known, but the partners are keen to progress this research in advance of the ARC decision, for the proposed project: "Optimising Value for Money (VfM) in software development through standardised cost metrics in the procurement, development and project management life-cycles". This project aims to develop the principles by which the ongoing costs can be estimated from specifications based on use cases and can then be precisely mapped into a Work Breakdown Structure (WBS) used by the project management to control a software engineering life-cycle.

The team believes that this research will allow organisations to set budgets with greater confidence and more accurately determine progress during software contracts. It has significant implications for procurement specialists purchasing software development. Problems encountered during a contract by customer contract managers and supplier project managers have often been introduced into the contract through suspect decisions on budget, scope, schedule and choice of supplier made at early stages of the procurement process. The validity of many VfM decisions would come under serious scrutiny if an identical analysis were to be repeated subsequent to contract closure. A reduction in the frequency of invalid cost estimates and inappropriate budgets would contribute to more successful software projects and improve the VfM proposition for bespoke software development in comparison to the purchase of packaged software. Any 'buy' decision over a 'build' decision should be based upon the business merits rather than any associated contract management problems.

2. Past Work

Curtin University researchers have had previous experience in the area of software engineering economics in the private sector, and are leading, coordinating, and undertaking preliminary research on this topic.

3. Issue Statement

Early functional techniques for estimating the cost of system requirements have been largely superseded by object-oriented techniques [1] forcing an evolution of cost estimation methodologies.

The introduction of object-oriented software engineering, has improved programming efficiency but has not generally led to improved cost estimation of large, complex systems. In fact it could be argued that since many estimation techniques are still based on functions, not objects, estimation has not yet fully adapted.

This research examines one of the problems associated with producing an initial accurate cost estimate for a project with extensive and complex use case based specifications, namely managing the combinatorial complexity of such systems. It also examines the ability of project managers to use these estimates.

Specifically the research aims to build upon and extend the existing techniques for costing contemporary software systems to develop a link between cost metrics at the requirements definition phase and the Earned Value of completed requirements during the development phase of the project. Analysis of the scholarly literature, supported by interviews with a number of leading practitioners, indicates that little theoretical work has been done in this area. The result is a chronic inability to consistently and reliably determine the ongoing value of the developed system against the initial specifications.

Type of Issue

Multimedia Victoria [2] highlights the outcomes of The Standish Group's CHAOS Report [3], which "found that 32% of projects terminate before delivery and only 11% are completed on budget. Of the remaining 57% of projects that are completed, the average budget overrun is 87%."

Lederer and Prasad [4] report that only one in four systems development projects in a study group completed within cost estimates. Lederer and Prasad researched cost estimation techniques and searched for the causes of inaccurate costs estimates, recommending that estimators follow an estimating methodology. The findings included a recommendation to "Reassess the project control system for accountability and management commitment" (Lederer & Prasad, [op. cit.]), demonstrating the link between the representation of software cost at the requirements phase and that used by the project managers.

Research undertaken by the investigators to date has uncovered little evidence that the academic or practitioner communities have scientifically investigated the link between the costs associated with use cases and those costs used to represent requirements in WBSs. A common approach is to directly map use cases into the WBS but the investigators argue that this is not valid for large projects, particularly if there are significant numbers of 'include' and 'extend' use cases. In these situations there exists a many-to-many relationship between use cases and functionality.

Context

Diverse representations of individual requirement cost by software developers and project managers, means that there is usually lack of accord between procurers, project managers and software developers on an appropriate contract price and whether value is being obtained. This project will develop principles to connect existing academic and practitioner research in procurement (for example, southernSCOPE method (Multimedia Victoria, [op. cit.]), project management (WBS and Earned Value) and software metrics (FPA, COCOMO, use cases) to develop a comprehensive and rigorous representation of software costs that is consistent with all three lifecycles. These principles will form the basis of a new methodology that will facilitate a reduction in the cost of software development through greater reliability in cost estimation. This is particularly valuable in the case of multi-million dollar, complex and mission critical software systems, used by organisations such as the Australian DoD.

This will allow VfM procurement decisions to be more rigorous, variations between actual and budgeted costs of requirements (Earned Value) to be more accurately calculated, project problems to be identified and actioned more rapidly, and rework of costing to be minimised. Most importantly a common view on costs will reduce contract conflict between customers and suppliers, resulting in lower management costs. Essentially it will contribute to a reduction in expenditure on software development through fewer bad purchases and reduced problems during development. In addition the professionalism of procurers (buyers), and software developers (suppliers) should improve.

Stakeholders

Glass [5] describes Westpac's Core Systems Redevelopment project, which was terminated in 1992 at a cost of AUD$150m, almost twice the original estimate. Problems faced by procurers include determining appropriate budget allocations, understanding whether offered solutions have been priced correctly, choosing the best VfM solutions, and evaluating whether the budget is on track during the contract. Project managers have problems in aligning their project based cost and schedule systems with the object-oriented representations of use case costs, both at the project start-up and during project operation, giving them difficulties in discovering cost crisis points. Software engineers are required to supply cost estimates to procurers and project managers long before they have been able to undertake sufficient analysis to accurately determine these costs.

Information Needs

Long term historical data on initial project costing techniques and associated estimates must be examined in the context of project delivery and final costs for projects where specifications were given as use cases. Ideally this data would include a variety of cost techniques that are based upon a number of techniques. Careful examination of this data would identify the impact of the estimation process on the final budget situation.

A longitudinal study over an extensive period would be required to test the effectiveness of alternatives proposed by this research with current techniques, from the initial project concept phase to the completed delivery. A four year timeframe has been proposed to the ARC for this project.

4. Proposed Approach

Jacobson [6], the architect of the use case driven approach to object-oriented software engineering, would not recommend estimating the cost of a system at the use case or requirements phase. An accurate cost requires information that can only be obtained after detailed analysis and a reasonable amount of design is undertaken. It is not until the design phase that implementation issues are uncovered. However customers need firm estimates of system costs long before design can be undertaken and cost estimation techniques become very important in defining budgets.

Jacobson [op. cit.] also understands the potential mismatch between functionality and objects.

> *Since it is often hard to draw clear borders between functionalities in a system, in practice the developers are forced to make many judgements about where to split functionality between objects.*

A direct mapping of use cases into a project WBS can therefore introduce problems in assigning costs to functionality.

Function Point Analysis (FPA), which counts the individual functions of a system, has a long history of success with functionally specified requirements and has now been applied to object-oriented specifications. However the authors argue that where use cases describe large or complex systems there is no simple functional output from use cases. FPA has been applied successfully to small scale use case systems (Multimedia Victoria, [op. cit.]), but encounters difficulties with large scale systems because not all functional requirements are initially defined.

What is missing is a mapping layer that maps the use cases to all of the functional requirements. This second layer of functional decomposition of use cases could potentially map directly into the WBS and provide the basis of function point counting. This would allow the complexity of a use case to be described rather than implied and map elements of disparate use cases to single WBS elements. Then each of the functional elements could be estimated as accurately as possible using FPA. This would give as an additional benefit a cost breakdown at the level understood by the procurers, project managers and software engineers. Thus Earned Value of requirements could be determined, and the VfM implicit in the procurement of software development becomes more explicit.

Process or Solution

Cost estimation of small systems based on use cases is manageable if a structured approach is followed, but as the number of use cases increases the task becomes more difficult. The authors suggest that this is because there is a non-linear relationship between the summed complexity of the use cases and the complexity of the total system. A complex multi-million dollar system would have over 100 use cases. At a high level the complexity of these use cases and their interactions can easily be overlooked and the costs underestimated. For large and complex systems the “devil is in the detail”.

Function points and WBSs are both based upon functional requirements, and although both use cases and function points view a system from a user perspective, use cases are deliberately not functionally decomposed. The estimation of ‘functions’ from use cases is therefore fraught with problems.

It is proposed that a mapping between the use cases and the functional requirements be undertaken through a functional decomposition of objects.

Research Methods

In terms of the aims of this research and the background to the research problem, the researchers have developed a six-stage methodology, centred around a “collective case study” approach [7], with triangulation being developed on the basis of results arising from the literature review, the deconstruction of a number of small cases, and concept testing of static and live data.

1. Initial Project Scoping, Literature Review and Preliminary Publications
2. Search for Project Exemplars
3. Refinement of the Problem Definition and Prototype Development
4. Development and Administration of Test Sets to the Earned Value Model

5. Live Testing against a Completed Software Project
6. Live Predictive Testing on a Forthcoming Software Development Project
7. Completion of Results, Journal Publications and Development of Training Manuals
8. Preparation of Collaborative Workshop for Researchers and Practitioners
9. Dissemination of Results and Elaboration of Future, Related Research.

Previous Work

Several important concepts drawn from the fields of object-oriented software engineering, software cost estimation and project management provide the foundation for this project and are described below.

This project aims to build upon and extend the existing techniques for costing contemporary software systems to develop a link between cost metrics at the requirements definition phase and the Earned Value of completed requirements during the development phase of the project.

Use Case

Use cases, formalised by Jacobson [op. cit.], provide a popular technique for capturing requirements for object-oriented software analysis and design. A use case is commonly described as a sequence of actions an Information Technology (IT) system performs that yields an observable result of value to a particular Actor. An Actor is someone or something outside the system that interacts with the system. Use cases capture who (Actor) does what (Interaction) with the system, for what purpose (Goal), without dealing with system internals. A complete set of use cases specifies all the different ways to use the system and behaviour required of the system, thus bounding its scope. Overall, the contribution of Jacobson [op. cit.] rests with the capacity it demonstrates to facilitate the capture of user' requirements, control iterative development, and formulate system tests.

Function Point Analysis

Function Point Analysis (FPA) was first published in 1979. "In 1984, the International Function Point Users Group (IFPUG) was set up to clarify the rules, set standards, and promote their use and evolution. FPA provides a standardised methodology for measuring the various functions of a software application. FPA measures functionality from the user's point of view, that is on the basis of what the user requests and receives in return. The function points (FPs) are obtained by measuring the software application from two distinct perspectives:

1) The functional size, calculated by assigning weights to *each individual function*…
2) The value adjustment factor (VAF), calculated using predefined general systems characteristics (GSC) to asses the environment and processing capacity of the software application *as a whole*." [8]

Work Breakdown Structure

The Work Breakdown Structure (WBS) is a key component of project management methodologies, as evidenced by the Project Management Institute's [9] production of a WBS Standard. The WBS concept, as defined by its originator, the US Department of Defense [10] is:

- "A product-oriented family tree composed of hardware, software, services, data, and facilities…
- A WBS displays and defines the product, or products, to be developed and/or produced. It relates the elements of work to be accomplished to each other and to the end product.
- A WBS can be expressed down to any level of interest..."

Earned Value

Earned Value is defined by Fleming and Hoppelman [11] as "What you physically got for what you actually spent; the value of the work accomplished …" It is further described by DMO [12] as "a set of Best Practice Project Management Principles that integrate Cost, Schedule and Technical Performance…" and "establishes objective measures of the actual work achieved compared to the plan for that work…" DMO [op. cit.] also states that Earned Value Management "Provides the Commonwealth and Company Management with the same performance data that reflects an objective measure of contract progress, aids in analysis of the future cost and schedule outcomes, and aids in informed and timely decision making by providing early indication of possible problem areas."

5. Results, Status, Prospect, and Needs

Status

This project is in its early stages. Historical data going back to 1999 from a project being undertaken by one of the partner organisations will be examined as soon as sufficient time can be devoted. Funding is required to free key personnel before serious effort can be expended.

The authors have drawn upon experience and research in the fields of software engineering, project management and procurement to highlight potential flaws in the application of cost estimation methodologies to object-oriented software engineering projects. A modification is proposed that could contribute to increased estimation accuracy and improved project outcomes. A multi-disciplinary research project is being developed to progress this research, including experts in use case requirements, function point analysis, project management and procurement. A number of organisations that produce or use large complex use case based software systems have agreed to participate in the project, however funding is currently an issue.

Challenges

One of the most significant challenges in this project is the volume of data to be analysed in the time available. The data will be sourced from a number of significant use case projects.

The use cases upon which the original cost estimations were based must be analysed and mapped to the delivered functionality of the systems and actual cost of each component. In addition, attention must be given to hidden costs deriving from: components that were developed but not delivered, re-engineering, project inefficiency and scope creep.

The next major challenge will be to test the new principles on the exemplars and, if possible new projects. This will require: a full analysis of the selected exemplars, detailed work decomposing the use case to the functional elements, and parallel execution of existing techniques and the principles developed through this project.

6. References

[1] Jalote, P. (2002). 'Software Project Management in Practice'. Boston, Addison-Wesley.

[2] Multimedia Victoria. (2003). SouthernSCOPE, Retrieved: 2 February 2004 from http://www.mmv.vic.gov.au/southernscope.

[3] The Standish Group (1995). 'The Standish Group Report -CHAOS'. Retrieved: 15 December 2003 from http://scs.carleton.ca/~beau/PM/Standish-Report.html.

[4] Lederer, L. and Prasad, J. (1995). 'Causes of Inaccurate Software Development Cost Estimates.' Journal of Systems Software, Vol 31, page 132, pp 125-134.

[5] Glass, R.L. (1998). *Software Runaways – Lessons Learned from Massive Software Project Failures.* Upper Saddle River, NJ, Prentice Hall.

[6] Jacobson, I. (1992). Object-Oriented Software Engineering - A Use Case Driven Approach, page 143. Reading, MA., Addison-Wesley Publishing.

[7] Stake, R. (2000). ‘Case Studies.’ In Denzin, N. and Lincoln, Y. (Eds.) *Handbook of Qualitative Research (2nd Ed),* page 437. Sage. London.

[8] Abran, A. and Robillard, P.N. (1996). 'Function Point Analysis: An Empirical Study of Its Measurement and Processes.' IEEE Transactions on Software Engineering, Vol. 22, No. 12, page 895, pp 895-910.

[9] Project Management Institute (2001). Practice Standard for Work Breakdown Structures, available online at: http://www.pmi.org/prod/groups/public/documents/info/pp_wbsexcerptsfile.pdf

[10] Department of Defense (1998). MIL-HDBK-881, Department of Defense Handbook, Work Breakdown Structure, Chapter 1, page 4. Retrieved 2 February 2004 from http://www.acq.osd.mil/pm/newpolicy/wbs/mil_hdbk_881/mil_hdbk_881.htm.

[11] Fleming, Q.W. and Hoppelman, J.M. (1996). *Earned Value Project Management*, page 30. Upper Darby, Penn., Project Management Institute.

[12] Defence Materiel Organisation (DMO) (2003). Earned Value Management. Retrieved 2 February 2004 from http://www.defence.gov.au/dmo/esd/evm/index.cfm.

7. Biography

Diane Jamieson

Diane Jamieson is a Senior Research Fellow at Curtin University of Technology, Perth, Western Australia, working for the Chair of Leadership in Strategic Procurement.

Diane has a strong interest in the described area of software engineering economics, having worked as a procurement specialist, project manager and software engineer. She has tertiary qualifications in science, computer science and business.

Kevin Vinsen

Kevin Vinsen holds the position of Principal Architect for Commercial Systems with the IS3 group of ADI Limited. He has specialist skills in object-oriented analysis and design and a special interest in cost estimation metrics for OO systems. Kevin has a detailed knowledge of use case theory and the application of use cases to large scale software systems.

Kevin also lectures on a part time basis at Murdoch University, Perth where he teaches software metrics

Professor Guy Callender

Professor Callender is the industry-funded foundation Chair of Leadership in Strategic Procurement at the Curtin Business School, Curtin University of Technology.

His principal areas of recent research relate to the relationship between procurement and logistics systems, and their relationships and interfaces with Information Systems and other areas of operational management