Costing of COTS-Based Systems: An Initial Framework

Hakan Erdogmus
Institute for Information Technology,
National Research Council Of Canada
Montreal Road, M-50
Ottawa, Ontario
Canada K1A OR6
Hakan.Erdogmus@nrc.ca

Carol A. Sledge
Software Engineering Institute,
Carnegie Mellon University
Pittsburgh, PA 15213-3890, USA
cas@sei.cmu.edu

Michael Looney
DERA, Ministry of Defence
DERA Portsdown West
Portsdown Hill Road
Fareham, PO17 6AD, UK
mjlooney@dera.gov.uk

Shonnag Allison
Specialist Procurement Services,
Defence Procurement Agency
Elm 1b#187
MOD Abbey Wood
Bristol BS34 8JH, UK
SPS-CF14e@dpa.mod.uk

ABSTRACT
Costing is an important part of COTS-based system development. Many development process activities and costing activities are intricately related, and thus best be addressed in tandem. Like the development process itself, costing should be treated as a continuous activity in COTS-based development. The necessary infrastructures must be in place for its support and monitoring. In this paper, we address the main features of a rudimentary costing model for COTS-based systems with these features, and identify some additional areas for improvement.

Keywords
Cost estimation, COTS-based systems, cost models

1 INTRODUCTION

Software development based on Commercial Off-The-Shelf, or COTS, products is an emerging paradigm that is gaining tremendous popularity (Dean and Vigder, 1997; Carney and Oberndorf, 1998). Savings and shorter development times are two primary driving forces for it. Since the COTS approach may not make economical sense when these two forces are weak (Erdogmus, 1999), costing is a critical issue for its success.

However, costing for COTS-based systems (CBS) is particularly difficult. One reason is the need to consider whole-life costs rather than just the initial development costs, due to the highly iterative, continuous nature of the CBS development process. Technology refreshes and COTS product upgrades force the initial development activities to be repeated to varying degrees during the operational sustainment phase, making development, and thus costing, an ongoing activity.

Additional cost factors that may not be prevalent in custom development must be accounted for in a CBS. An exhaustive list of these factors is out of the scope of this paper. Critical factors include product assessment, evaluation and selection, market and technology watch, technology forecasting, vendor and product maturity, engineering of evolvable architectures, reintegration, use of risk mitigation strategies, as well as management of licenses, warranties, and data rights for the COTS products used.

Moreover, costing is intricately intertwined with the development process in that cost estimates and the outcomes of other costing activities feed into many other process activities. Costing is an inseparable part of the CBS development process. For example, product selection and COTS architecture development cannot ignore the underlying cost issues. Similarly, during operational sustainment, upgrade and technology refresh decisions must depend on cost inputs.

So far existing COTS cost models, such as COCOTS, primarily focuses on initial development, and view costing as a one-time activity. This is a good start, but does not address a critical aspect COTS-based system development. The need for models that can support continuous development from a whole-life viewpoint is increasing.

2 RUDIMENTARY SEI MODEL

Software Engineering Institute at Carnegie Mellon University is currently developing a cost model (hereby called “the model”) under the sponsorship of the U.S. Department of Defense within the context of its COTS-Based Systems initiative. In this section, we point out to some important features of this model. In the next section, we propose some modifications that we believe will improve its effectiveness.

2.1 Features of the Model

The model revolves around the following tasks:

- identification of new or changed cost factors;
- data collection and market watch;
- cost estimation;
- cost and marketplace monitoring; and
- calibration of model parameters based on data collected during monitoring activities.

The cost estimation task itself builds on a multi-level model that is open and robust. The model is bottom-up in that estimates are derived based on individual process activities. Both parametric submodels and average values are proposed for estimation. To estimate activity effort and schedule that have variability, parametric submodels will be used. To estimate activities that are constant in effort and schedule, average values will be used.

The different levels of the estimation model differ in the amount of detail they address. The three levels that are being proposed, starting with the highest level are, the Concept Model, the High-Level Model, and the Detailed Model. Deeper levels require an increasing amount of knowledge about the end system. The Concept Model is intended for the concept exploration. The High-Level Model will aid in business case analysis and prototyping. Finally, the Detailed Model will be used in the actual production (initial development) and operational sustainment.

The Concept Model relies on a set of rules of thumb to estimate effort and schedule for a given development activity. The rules of thumb are based on the COTS product types to be used in the system. The estimates' accuracy can be supplied through a confidence interval, for example, within one unit of standard deviation of the estimated value. Here COTS product type refers to general product category, such as database, GUI builder, compiler, browser, spreadsheet, and so on, based on a maintained list and the associated data. The overall estimate can then be calculated from the estimates of the individual product types and activities. Adjustments may be required to the overall estimate in cases where overlaps exist.

The High-Level Model is also centered around COTS product types and activities. However, the product list used may be more refined. For example, the general categories may have been broken down to more specific categories. In addition, cost estimates are adjusted with respect to the COTS cost factors applicable to the activity being considered. For example vendor maturity and team integration experience may be accounted for by corresponding multipliers in the same spirit as COCOTS.

2.2 Adjustments to the Model

2.2.1 Focus of Concept Model

The main feature of the Concept Model is that its rules of thumb are based on COTS product types. Reliance on COTS product types for cost estimation at this level pose some problems.

First, such a fine level of granularity may not be desirable. In this case the complexity of the model grows with the list of COTS product types considered, which can be extensive. To get a quick, ballpark estimate, a relatively small number of rules of thumb would serve better. Second, the Concept Model should be a black box model; it should be as implementation independent as possible. For example, it may not be clear at the concept level, whether a separate database component is necessary to implement the required functionality, therefore this information should not be required.

Thus at the concept level, the focus is best placed on the end system, the application type, or the domain. The rules of thumb should be at the application or domain level rather than at the subsystem level. This provides a much higher granularity than the one provided by COTS product types. An example is a flight monitoring system for a specific aircraft, which can be implemented using different COTS products. The concept level cost estimate, however, can still focus on main application characteristics and domain features, necessitating only a relatively small number of rules of thumb. These rules may have been derived from experience with similar past projects. The COTS product information can be added at next level, or at the High-Level Model.

2.2.2 Additive Cost Factors

Another possible adjustment is concerned with the use of risk mitigation strategies during development. An example of such a strategy is to hedge against vendor demise by adopting a flexible architecture into which the COTS components can easily fit. Such strategies often require an up-front investment, and are likely to pay off in terms of future cost savings. Unlike other conventional cost factors, such as team experience, the underlying cost factor here is additive rather than multiplicative. They can be accounted for by a negative, or downward, adjustment to the original cost estimate. The amount of adjustment, however, may not be possible to calculate with traditional techniques. Valuation techniques such as real options analysis (Erdogmus, 1999) are used for this purpose.

2.2.3 Time Value of Money (TVM)

The last improvement concerns the time value of money. Cost estimation has traditionally ignored time value of money, partially because of its focus on initial development costs rather than whole-life costs. However, in long-lived systems where the whole-life costs are spread across the timeline, time value of money is important for costing. The same effort that costs a certain amount in today's dollars is unlikely to cost the same amount in five years. This is a fact of general economic behavior, but can also be influenced by other considerations that are specific to the government or military domain. Typically, different situations will necessitate different interpretations of TVM. For example, the interpretation of TVM for contract pricing will in general be different from its interpretation for budget allocation.

- *Time Value of Money as Return on Investment.* One way of looking at time value of money is as the minimum required return on investment. This can be phrased in terms of cost avoidance or savings. Let's say that one unit of effort spent today will pay off by saving 1+d units of effort in a year. Theoretically that 1+d units of savings could be re-invested back in to the project, paying off by saving $(1+d)^2$ units in two years. In N years, the effort will be worth $(1+d)^N$ years. In effect, d here is an annual compounded effort-to-benefit ratio that can be applied just like a discount rate to calculate the present value of future costs. The question of how to determine the appropriate rate, however, remains.
- *Time Value of Money as Opportunity Cost*. This is the traditional economic interpretation. Seen as an opportunity cost, TVM in a publicly funded project could be the government's borrowing rate, which is captured by the ongoing short-term risk-free interest rate. The reasoning behind this thinking is as follows: the government would save an amount equivalent to this rate in interest payments on its debt by withholding funds and using them for dept reduction. Again, this interpretation is a theoretical one, and it is economically sound. However, it may not be appropriate or applicable in all situations.
- *Time Value of Money for Contract Pricing*. For non-competitive contract pricing, TVM could be the average discount rate used by the government or military contractors. That way it reflects a fair market value for the bidder. This discount rate may be determined and supplied by a central body.
- *Time Value of Money for Budgeting*. It may be unsuitable to use TVM in its traditional sense for budgeting purposes in the public sector. Normally time is assumed to have positive value, that is, money is expected to grow. However, certain funding practices may nullify this hypothesis. In other situations, money will shrinks with passage of time, and time will have negative value. Suppose that a public project is allocated a fixed lump sum payment at the beginning of a 5-year budget based on projected costs, rather than receiving the funds in installments according to a cost schedule. Since a public project cannot invest the funds that are not immediately needed to accumulate interest, the project would be stuck with a shortfall of funds in the end if its future costs are discounted using a positive discount rate when seeking funding. In this example, time has no value. Suppose now that the government is planning to introduce across the board cuts, say 2% per year, in the next year's budget. To avoid funding shortfalls, the project's post first-year costs must be upward adjusted to account for the upcoming cuts. The end effect here is that time has negative value, and the underlying discount rate is negative.

These examples demonstrate that the correct interpretation of time of value of money in cost estimation is situational and may be contrary to the prevailing economic intuition.

2.3 Monitoring

One of the proposed features of the SEI model is monitoring. We addressed monitoring in the context of the COTS process; see the companion paper by Looney et al. Another important facet of monitoring pertains to costing. This facet involves the tracking of actual costs versus estimated costs.

The output of monitoring activities feed directly into cost estimation and model calibration. A continuous and systematic monitoring capability is instrumental for keeping the model parameters up to date in a fast-changing technology environment. For this capability to be possible, a suitable data collection infrastructure must be in place. The right kind of data will improve the quality of decision making within and across projects, but it is of value only when it is current, relevant, and used properly. For example, cost and schedule data collected during the early upgrade and refresh cycles may be very useful for making future product upgrade and technology refresh decisions.

Project management processes such as Earned Value Management (Fleming and Koppelman, 1996) can support and implement the cost monitoring capability. We feel that such processes will be most effective when the organization of the underlying activities mirrors the architecture of the system. This can be achieved, for example, by determining the underlying Work Breakdown Structure with a strong technical focus or based largely on a technical decomposition. For the use of EVM in COTS-based systems, see Stanley, Oberndorf, and Sledge (2000).

3 DISCUSSION

Costing is an important part of CBS development. Many development process activities and costing activities are intricately related and best be addressed in tandem. Like the development process itself, costing should be treated as a continuous activity in CBS development and the necessary infrastructures must be in place for its support and monitoring.

A model suitable for CBS must address not only whole-life costs in a satisfactory manner, but also account for new cost factors that are not normally present in traditional custom development. The rudimentary model that is under development at the Software Engineering Institute is a first step towards this direction. In this paper, we have summarized the main features of this model and identified some areas for improvement.

REFERENCES

Carney, D. and Oberndorf, P. (1998). Commandments of COTS: still in search of the promised land. Technical Report, Software Engineering Institute, Pittsburgh, PA.

COCOTS. http://sunset.usc.edu/COCOTS/cocots.html

Dean, J.C. and Vigder, M.R. (1997). System implementation using COTS software. In *Proceedings of the 1997 Software Technology Conference* (Salt Lake City, Utah,199).

Erdogmus, H. (1999). Quantitative approaches for assessing the value of COTS-centric development. In Proceedings of the Sixth International Symposium on Software Metrics (Fort Lauderdale, FL, November 1999)

Fleming, Q.W. and Koppelman, J.M. (1996). *Earned Value Project Management*. Project Management Institute.

Staley, M.J., Oberndorf. P., and Sledge, C.A. (2000). Earned Value for COTS-Based Systems. To appear in *STC' 2000, The 12th Software Technology Conference*, April 30–May 5, Salt Lake City, Utah.