Integrating Portfolio Management and Simulation Concepts in the ERP Project Estimation Practice

Maya Daneva

University of Twente
m.daneva@utwente.nl

Abstract. This paper presents a two-site case study on requirements-based effort estimation practices in enterprise resource planning projects. Specifically, the case study investigated the question of how to handle qualitative data and highly volatile values of project context characteristics. We counterpart this challenge and expound upon the integration of portfolio management concepts and simulation concepts into a classic effort estimation model (COCOMO II).

1 Introduction

Business-requirements-based effort estimation is a practical part of the early stage of any Enterprise Resource Planning (ERP) project. Though, how to construct realistic schedules and budgets so that an ERP-adopting organization can achieve cost-effective and timely project delivery is, by and large, unknown. Researchers [3,10] indicate that existing project estimation practices (e.g. [1]) are limited by their inability to counterpart the challenges which the ERP project context poses to estimation analysts, for example, how to account for the uncertainties in cost drivers unique to the diverse configurations, system instances and versions [3] included in the solution. Here we explore one possible approach as a remedy to this situation. In case study settings, we complementarily deployed the COCOMO II effort estimation model [1] at the requirements stage, and portfolio management (PM) and Monte Carlo (MC) simulation concepts. In what follows, we provide a background on the approach and, then, we report on our case study plan, its execution, and our early conclusions.

2 Background

Our approach to uncertainties in ERP effort estimation rests on four types of sources: (i) the COCOMO II model [1] that lets us account for ERP adopter's specific cost drivers, (ii) the MC simulation [8] which lets us approach the cost drivers' degrees of uncertainty, (iii) the effort-and-deadline-probability-based PM concept [9] which lets us quantify the chance for success with proposed interdependent deadlines for a set of related ERP projects, and (iv) our own experience in ERP RE [4], which was used to incorporate the effort estimation process into the larger process of early RE (that is, at time of bidding). We chose the combination of (i), (ii) and (iii), because other researchers already used it [7] and found it encouraging. In marked contrast with these authors [7] who blended these techniques for the purpose of custom software

B. Paech et al. (Eds.): REFSQ 2008, LNCS 5025, pp. 147–152, 2008.

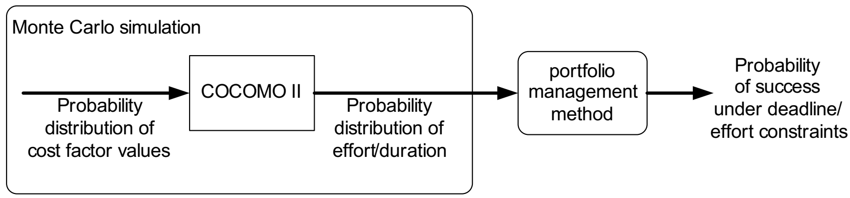

Fig. 1. A conceptual model of the solution approach

contract bidding, we adapt the techniques to the ERP project context and we use them jointly therein.

Fig. 1 presents how the three techniques fit. Because we want to incorporate ERP project context uncertainties into the project estimates, we suggest COCOMO II take as inputs the probability distributions of the COCOMO scale factors and cost drivers (instead of taking as inputs single values as in [1]). We use the MC simulation to get randomly-selected values into COCOMO II and, then, see how likely each resulting outcome is. Our approach yields as a result the possible effort and duration estimation values for each uncertain factor. Unlike COCOMO II, our output is the probability distributions of effort and duration and not the most likely effort and duration which COCOMO II creates. The probability distributions are fed into the PM method [9]. To run it, we first bunch projects into portfolios and, then, obtain the probability of successfully delivering the projects under both time and effort constraints. The application of this solution in context is described below.

3 The Case Study Plan

Our case study was planned as per the guidelines in [14]. Our overall goal was to determine whether the use of PM increases the chance of success and, if so, to what extent. Our expectation was that the ERP projects with high uncertainty ratings of the COCOMO II scale factors and cost drivers would benefit more from PM, than the projects with low uncertainty ratings would. The scope of the case study covers two sites in a large North-American company. Each site represents an independent business unit running their own ERP projects based on a specific package. Prior to the case study, the units were independent firms which merged. While the first site implemented three modules of the PeopleSoft ERP package, the second site [5] rolled out a large organization-wide ERP solution that included eight functional modules of the SAP package. A condensed summary of the case study setup pertaining to the SAP site has been presented in a ESEM'07 poster [5].

The three techniques from Fig. 1 are summarized as follows:

COCOMO II: We used it because (i) it's a popular and comprehensive empirical parametric model [1] and (ii) both our sites had data allowing its use. COCOMO II produces estimates of effort and duration by using two equations as follows:

$$\text{Effort} = A \times (\text{Size})^{E} \times \prod_{i=1}^{17} EM_{i} \qquad \text{and} \qquad \text{Duration} = C \times (\text{Effort})^{F} \qquad (1)$$

Therein, E and F are calculated via the following two expressions, respectively:

$$E = B + 0.01 \times \sum_{j=1}^{5} SF_{j} \quad \text{and} \qquad F = D + 0.2 \times (E - B) \qquad (2)$$

In (1), 17 cost drivers (*EM*) serve to adjust initial effort estimations. In **(2)**, five scale factors (*SF*) reflect economies and diseconomies of scale observable in projects of various sizes. The degrees of these 22 context characteristics are rated on a seven-point scale, ranging from 'extra low' to 'extra high'.

Monte Carlo simulation: This is a problem-solving technique to approximate the probability of certain outcomes by running multiple trial runs, called simulations, using random variables. Here, we use it to obtain a range of possible values for our estimates, while taking the COCOMO II cost drivers and their degrees of uncertainty as inputs. We borrowed this idea from the THAAD Project Office [8] and the JLP NASA [6]. We run it according to these steps: (1) ascribe a particular distribution type to an input variable in COCOMO II; (2) repeatedly run the model 10000 times and collect samples of the output variables for each run so that we produce an overall picture of the combined effect of different input variables distribution on the output of the model; (3) plot a histogram showing the likelihood of obtaining certain output values for the set of input variables and attached distribution definitions.

Portfolio management: The PM method in [9] quantifies the uncertainty associated with a project estimate for a set of projects managed as a portfolio. It gives the opportunity to obtain a probability of a portfolio's success under effort and schedule constraints. We chose it because: (i) it is applicable at the stage of requirements [9], (ii) its only input requirement is a record of previous projects; and (iii) it fits with the ERP adopters' project realities suggesting that an ERP project is implemented as a portfolio of interdependent subprojects. Each subproject is a piece of functionality (or an ERP module) linked to other pieces. For example, the Asset Management functionality in a package is tightly linked with the Financial Accounting module and the Controlling module. Given a set of interdependent subprojects, the effort estimation model yields (i) the probability of portfolio's success with the proposed deadlines for each subproject in this portfolio, and (ii) a set of new deadlines which will result in a required probability of success. The portfolio success is judged by two conditions applied to any two subprojects ***a*** and ***b*** for which deadline_a is earlier than deadline_b. The conditions are that: (i) subproject ***a*** is to be over by deadline_a and (ii) subproject ***a*** and subproject ***b*** are to be over by deadline_b. In other words, the conditions require all subprojects planned with a deadline before deadline_b to be completed by deadline_b , rather than just project ***b***. This is the key to the portfolio approach, because uncertainty about completion of project ***b*** incorporated uncertainty from all previous projects. Suppose in total *E* people are on the project and let *d* be the number of work days it takes from start date to deadline, then the total available resources is *E*x*d*. So, suppose an ERP portfolio ***Y*** is made up by *n* subprojects, the success conditions are represented as follows:

$$\begin{pmatrix} Y_1 \\ Y_1 + Y_2 \\ \ldots \\ Y_1 + Y_2 + ..Y_n \end{pmatrix} \le E \begin{pmatrix} d_1 \\ d_2 \\ \ldots \\ d_n \end{pmatrix} \quad (3)$$

where Y_i is the estimated effort for subproject i to succeed. We check if, for any j, (j= 1..n), the sum of $Y_1,..,Y_j$ is greater of Exd_j. If this is true, then deadline d_j has failed. Success probabilities result from simulations in which $Y_1,...,Y_n$ are generated from a predetermined probability distribution. If we deem $Y_1, \ldots, Y_n$ is satisfying all conditions, then we say that the portfolio $\boldsymbol{Y}$ succeeds. The portfolio's probability of success is equal to the ratio of the number of successes in the set $\boldsymbol{Y}$ to the number of trials in the simulation.

4 Case Study Execution

We modeled the uncertainty of the 22 context factors by means of a probability distribution, which means identifying for each factor (i) its distribution type and (ii) its parameters. We did this based on proposed default choices by other authors [6,7,8], e.g. McDonald's [8] default 'high' levels of uncertainty associated to the ratings of the RESL, DATA, ACAP and PCAP cost drivers [1]. The level of uncertainty determines, in turn, the distribution type to be assigned to each cost driver: normal, triangular, and uniform for low, medium and high uncertainty, respectively.

Next, the matter that COCOMO II provides duration estimation (2), encouraged us to formulate the following condition for PM in terms of time constraints:

$$\begin{pmatrix} T_1 \\ T_1 + T_2 \\ \ldots \\ T_1 + T_2 + \ldots T_n \end{pmatrix} \le \begin{pmatrix} m_1 \\ m_2 \\ \ldots \\ m_n \end{pmatrix} \quad (4)$$

where T_i is the ERP implementation time in months for subproject i. We note that this condition does not include the number of people E, because COCOMO II assumed an average number of project staff [1] which was accounted in (2). Furthermore, as recommended in [7], we attempted to improve the chances for portfolio success by adjusting the cost drivers and scale factors. Hence, we adopted the assumption that for projects with two different ratings for the same factor, the probability of success for each project will be different too.

Project data: The data we used in the first site were collected from six PeopleSoft projects completed between May'98 and June'00 and the data in the second site - from 13 SAP projects carried out between Nov'97 and Oct'03. In this period, the author was employed by the case company as a SAP process analyst and was actively involved in the projects. In both sites, for each project, we got (i) project size data, (ii) reuse levels, (iii) start and end dates, and (iv) scale factor and cost driver ratings. Size (see equation (1)), was measured in terms of unadjusted Function Points (FP) [4]. We counted it by using the standard rules of the International Function Point User Group (IFPUG, www.ifpug.org). The first site employed a IFPUG-certified FP specialist who counted FP from requirements and architecture design documents delivered by the PeopleSoft consultants on board. The second site also followed the IFPUG standard, but used the counting rules specifically refined to the observable elements of functionality in the SAP business requirement documents [4]. The effort multipliers *A, B,* and *EM*, and the scale factors *SF* were calibrated for each site by using ERP effort data collected between 1997 and 2004 in the two business units. We note that in both sites, we did not have any knowledge about the uncertainty of the scale factors and cost drivers ratings and therefore, we used default levels proposed by other authors [6,7,8]. We opted to use a lognormal distribution for functional size, as this was motivated by Chulani's observations [2] that (i) the skew of the size distribution is positive and that (ii) log(size) is likely to be a normal distribution. With this input data, we run MC simulations (a total of 10000 trials) which gave us samples of (i) effort, expressed in person-months, and (ii) duration, expressed in months.

Results: To see how the change of uncertainty levels of a cost driver rating impacts the project success under effort and schedule constraints, we constructed two portfolios: the first one had this driver rated as 'very high' for all projects and the other portfolio had it rated as 'very low' for all projects. For each portfolio, we calculated the probability of success under time constraints and under effort constraints. For example, Table 1 indicates that – at both sites, when the ERP-specific tools (TOOL [1]) were used in the project, the probability of success was higher under both time and effort constraints.

Table 1. Analysis of the probability of success for the factor TOOL under effort constraints

TOOL rating	Site	Probability of success	
		Under effort constraints	Under time constraints
Very low	PeopleSoft	46.33%	51.54%
Very high	PeopleSoft	97.99%	96.88%
Very low	SAP	49.27%	51.88%
Very high	SAP	98.01%	95.72%

Table 2. Probability of success for low/high uncertain projects under time constraints

Uncertainty level	Site	Probability of success		Ratio of increase (b)/(a)
		Individual projects (a)	Portfolio (b)	
Low uncertainty	PeopleSoft	21.45%	90.93%	4.23
High uncertainty	PeopleSoft	14.31%	86.77%	6.06
Low uncertainty	SAP	15.76%	87.52%	5.55
High uncertainty	SAP	8.31%	75.91%	9.13

In both sites, we observed that 13 of the 17 COCOMO II drivers can be adjusted in a way that maximizes the chance of success. Furthermore, we used 'the ratio of increase' [5,7] (i.e. the utmost right column in Table 2) to see whether the probability of success increases (and if so by how much) when projects are managed as a portfolio. Table 2 suggests that bundling ERP projects as a portfolio had the advantage over managing projects separately under time constraint.

5 Conclusions

Many issues arise when estimating ERP project costs from early requirements. This two-site case study applied an approach targeted to resolve the issue of volatile values of context factors which impact project outcomes. We learnt that: (1) to get a better estimate, we must be flexible and open enough to exploit the power of synergies among the three techniques and to learn from qualitative details of context; (2) examining the uncertainties in the context of each ERP portfolio clearly and from diverse sides helps us learn more about the effort estimation problem we face; (3) to ERP-adopters, this approach might be one good alternative over vendor-provided project estimates. However, our results are preliminary only. We are aware of related validity concerns [11] and plan a series of case studies to test our approach, to properly evaluate its validity and to come up with an improved version of it.

References

[1] Boehm, B.: Software Cost Estimation with COCOMO II. Prentice Hall, Upper Saddle River (2000)
[2] Chulani, S., Boehm, B.W., Steece, B.: Bayesian Analysis of Empirical Software Engineering Cost Models. IEEE Trans. on Software Eng. 25, 573–583 (1999)
[3] Daneva, M., Wieringa, R.: Cost Estimation for Cross-organizational ERP Projects: Research Perspectives. Soft Quality J. 16 (2008)
[4] Daneva, M.: Measuring Reuse of SAP Requirements: a Model-based Approach. In: 5th International Symposium on Software Reusability, pp. 141–150. ACM Press, LA (1999)
[5] Daneva, M.: Approaching the ERP Project Cost Estimation Problem: an Experiment. In: 1st International Symposium on Empirical Software Engineering and Measurement, p. 500. IEEE Press, New York (2007)
[6] Hihn, J.: Model-based Estimate. Technical report, JLP NASA (2004)
[7] Jiamthubthugsin, W., Sutivong, D.: Protfolio Management of Software Development Projects Using COCOMO II. In: 34th IEEE International Conference on Software Engineering, pp. 889–892. IEEE Press, New York (2006)
[8] McDonald, P., Giles, S., Strickland, D.: Extensions of Auto-Generated Code and NOSTROMO Methodologies. In: 19th International Forum on COCOMO, LA (2004)
[9] Fewster, R.M., Mendes, E.: Portfolio Management Method for Deadline Planning. In: 9th International Software Metrics Symposium, pp. 325–336. IEEE Press, Los Alamitos (2003)
[10] Stensrud, E.: Alternative Approaches to Effort Prediction of ERP Projects. J. Inf. Soft. Techn. 43, 413–423 (2001)
[11] Yin, R.: Case Study Research, Design & Methods. 3rd edn. Sage, Newbury Park (2002)