Complementing Approaches in ERP Effort Estimation Practice: an Industrial Study

Maya Daneva
University of Twente
Drienerlolaan 5, Enschede 7500,
The Netherlands
++31 53 489 2889
m.daneva@utwente.nl

ABSTRACT
Projects implementing enterprise resource planning (ERP) solutions are characterized by specific context factors such as high level of reuse, scope of the ERP modules, interdependent functionality, and use of vendor-specific standard implementation method, all of which impose risks known to cause various degrees of project failure. We suggest a remedy to this issue by tackling it from a portfolio management perspective. Our solution rests on earlier work by other authors and is a combination of a classic cost estimation method (COCOMO II), a Monte Carlo simulation process, and a portfolio management model. We report on the results of a case study done in a company site in the telecommunication sector.

Categories and Subject Descriptors
D.2.8 [**Software Engineering**]: Management – *cost estimation, time estimation.*

General Terms
Measurement, Economics.

Keywords
Enterprise Resource Planning, Effort Estimation, Portfolio Management.

1. INTRODUCTION
Effort estimation models exist to provide ERP adopters, as early as the stage of requirements, with predictions of the resources needed for their ERP projects. In practice, however, these models are inadequate and ERP projects still tend to have a very high frequency of schedule and cost overruns, quality problems, and outright cancellations. As studies [3,9] indicate, ERP implementation projects still experience a shortage of proper methodologies to evaluate size, effort, productivity and other cost factors. This is coupled with high level of uncertainty regarding the factors that drive the effort for a number of project activities, for example joint RE and architecture design (AD). Traditional software effort estimation methods do not yield accurate results in ERP context because they account for factors which only partially describe this context, and they let vendors, consultants and outsourcing partner companies incorporate their bias and intuition into the estimate. Even established approaches, such as the COCOMO family of models [1], offer only a partial fit with the ERP projects [9]. In this paper, we make a step to improve the existing practice of ERP effort estimation as part of the ERP requirements engineering (RE) process.

We propose an integrated approach which complements COCOMO II with the concepts of portfolio management and Monte Carlo simulation. In the remainder of this paper, we present our approach in Section 2. In Section 3, we report on a case study in which we applied it, and in Section 4, we draw some early conclusions.

PROMISE'08, May 12-13, 2008, Leipzig, Germany.

2. THE SOLUTION APPROACH
Our solution rests on four types of sources: (i) the COCOMO II reference model [1] that lets us account for ERP adopter's specific cost drivers, (ii) the Monte Carlo simulation [6] which lets us approach the cost drivers' degrees of uncertainty, (iii) the effort-and-deadline-probability-based portfolio management concept [8] 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 [3,4]. We chose the combination of (i), (ii) and (iii), because other researchers already experimented with it [5] and found it encouraging. Unlike these researchers, who used complementarily the three methods for the purpose of custom software contract bidding, we adapt each of the methods to the context of ERP projects and then, we adopt their joint use therein.

2.1 COCOMO II
COCOMO II is a well-known algorithmic model for objectively estimating software project costs [1]. It comprises (i) five scale factors, which reflect economies and diseconomies of scale observable in projects of various sizes, and (ii) 17 cost drivers, which serve to adjust initial effort estimations. In ERP project settings, at least three of the scale factors are directly related to the joint RE and architecture design (AD) activities, and thus raise the role of architects in reducing project costs. COCOMO II allows ERP teams to include in their estimates (i) the maturity level of the ERP adopting organization, (ii) the extent to which

requirements' and system architecture's volatility is reduced before ERP configuration, and (iii) the level of team cohesion and stakeholders' participation. In COCOMO II, the degrees of both the scale factors and the cost drivers vary from extra low, very low, low and nominal to high, very high and extra high. Suppose ERP project stakeholders assign a degree to each scale factor and cost driver, the estimation of project effort and duration will result from the two equations below:

$$\text{Effort} = A \times (\text{Size})^{E} \times \prod_{i=1}^{17} EM_i \quad (*)$$

$$\text{and} \quad \text{Time} = C \times (\text{Effort})^{F} \quad (**)$$

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

$$E = B + 0.01 \times \sum_{j=1}^{5} SF_j \text{ and}$$

$$F = D + 0.2 \times (E - B)$$

Because our model is targeted for application in the RE stage, in our research we let size be expressed in Function Points (FP) [4], which can be calculated on the basis of the limited information that is known at requirements. We chose this measure of functional size because it is applicable to any ERP package and not to a specific package's context [3]. Next, the effort multipliers *A, B,* and *EM*, and the scale factors *SF* were calibrated by using ERP effort data collected between 1997 and 2004 in the case study company. (For confidentiality reasons, the author could not donate the project data to the PROMESE database).

2.2 The Monte Carlo simulations

To obtain more realistic estimates, we approached the inherent uncertainty of the cost drivers by applying a Monte Carlo simulation technique, as suggested by the THAAD Project Office [6]. This is a problem-solving technique used to approximate the probability of certain outcomes by running multiple trial runs, called simulations, using random variables. Its purpose was to help us use a range of possible values for our estimates, instead of single stakeholders' guesses. This allowed us to feed randomly-selected values into the COCOMO II model and, then, see how likely each resulting outcome was. More in detail, the Monte Carlo simulation enabled us to ascribe a particular distribution type to an input variable in a model. When we run the model, the distribution attached to the input variable was randomly sampled and the result entered into the model. Repeatedly running the model many times and collecting samples of the output variables for each run helped use produce an overall picture of the combined effect of different input variables distribution on the output of the model can be produced. This was plotted as a histogram and showed the likelihood of obtaining certain output values for the set of input variables and attached distribution definitions. We took as inputs the COCOMO II factors and uncertainty values and, then, generated a population mean, standard deviation, and confidence intervals. For each uncertain factor, we obtained possible effort and duration estimation values.

2.3 The portfolio management concept

For the purpose of this case study, we deployed the portfolio management concept by Fewster and Mendes [8]. It rests on an effort and deadline probability model that allows us to quantify the uncertainty associated with a project estimate. We chose it because: (i) it is applicable at the stage of requirements or project bidding [8], (ii) its only input requirement is a record of previous projects; although it does require an effort estimate for every project, it need be nothing more sophisticated than a subjective opinion [8]; and (iii) it fits with the ERP adopters' project realities suggesting that an ERP project is implemented as a portfolio of interdependent subprojects [3,4]. Each subproject is a piece of functionality (or an ERP module) linked to other pieces (or modules). For example, the Material Management functionality in a package is tightly linked with the Sales and Distribution module. Suppose we have a set of interdependent subprojects, the effort estimation model will yield (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 the ERP adopter engages in total *E* people in 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:

$$\begin{pmatrix} Y_1 \\ Y_1 + Y_2 \\ \dots \\ Y_1 + Y_2 + ..Y_n \end{pmatrix} \leq E \begin{pmatrix} d_1 \\ d_2 \\ \dots \\ d_n \end{pmatrix} \qquad (***)$$

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 $Y_1, ...,Y_n$ is satisfying all conditions, then we say that the portfolio ***Y*** succeeds. The portfolio's probability of success is equal to the ratio of the number of successes in the set ***Y*** to the number of trials in the simulation.

3. THE CASE STUDY

3.1 Planning

The solution approach was applied in a setting of a large organization-wide ERP roll-out that included eight functional modules of one ERP package (namely SAP) and covered three locations of a North American telecommunication company [4].

We modeled the uncertainty of the five scale factors and the 17 cost drivers by means of a probability distribution, that is, we identified for each factor its distribution type and its parameters. This was done based on previously published experiences [5,6] and uncertainty assessments provided by our project stakeholders. Based on the observation that COCOMO II provides time estimation as in (**), we formulated the following condition for portfolio management in terms of time constraints:

$$\begin{pmatrix} T_1 \\ T_1 + T_2 \\ \dots \\ T_1 + T_2 + \dots T_n \end{pmatrix} \le \begin{pmatrix} m_1 \\ m_2 \\ \dots \\ m_n \end{pmatrix} \qquad (****)$$

where T_i is the ERP implementation time in months for subproject i. In this condition, we did not include the number of people E, because COCOMO II assumed an average number of project staff [1] which was accounted for in (**). Furthermore, as recommended in [4], 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. Finally, our case study plan included assessment of how much the probability of success increased when treating ERP projects as a portfolio. We expected that the subprojects with high uncertainty ratings would benefit more from portfolio management, than the projects with low uncertainty ratings would do.

3.2 Project data

Our data came from 13 SAP projects implemented in the case study company. The projects were carried out between November, 1997 and October, 2003. In this period, the author was employed by the case company as a SAP process analyst and was actively involved in the projects. The ERP implementation process model adopted in the context of the projects was the AcceleratedSAP (ASAP) RE process [3]. It is a project-specific process, engineered and standardized by SAP, and provided to clients by ASAP-certified consulting partners .

For each of the 13 projects, we got (i) project size data, (ii) reuse levels, (iii) start and end dates, and (iv) scale factor and cost driver ratings. Size was measured in terms of IFPUG FP [4]. Reuse levels were formed by using a reuse indicator that included reused requirements as a percentage of total requirements delivered [4]. 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 [5]. We opted to use a lognormal distribution for functional size, which was motivated by the observations of Chulani et al [2]. These researchers studies the size distribution and found that its skew is positive and that log(size) is likely to be a normal distribution. With this input data, we run Monte Carlo simulations [6] which gave us samples of (i) effort, expressed in man-month, and (ii) time, expressed in months. Generally, a Monte Carlo simulation consists of many - often thousands of, trials, each of which is an experiment where we supply numerical values for input variables, evaluate the model to compute numerical values for outcomes of interest, and collect these values for later analysis. In this case study, we used 10000 trials and generated the samples of effort and time, as presented in Figure 1 and Figure 2, respectively. In these figures, the Y-dimension shows the frequency with which a value was observed in the sample of 10000 trials. The X-dimension shows the value range. Because the average subproject involved four professionals (two business users, one external consultant and one internal IS team members), we adopted the assumption for E to be 4.

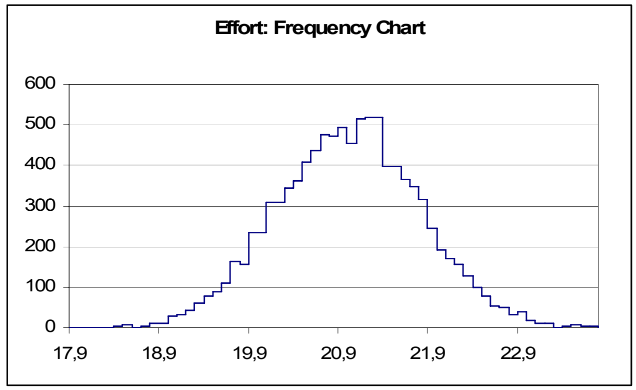

Figure 1. The Monte Carlo histogram of the probability distribution of effort (in person/months).

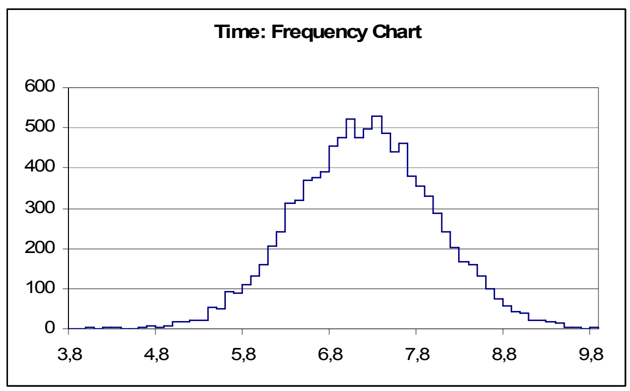

Figure 2. The Monte Carlo histogram of the probability distribution of time (in months).

3.3 Results

We summarize the results in three groups:

(1) findings regarding the use of portfolio management,

(2) findings concerned with adjusting cost drivers, and

(3) findings about the probability of success of highly-uncertain projects when managing them as a portfolio.

In the first group of findings, we observed that under effort constraints and under time constraints, the probability of success was 99.11% and 87.76%, respectively. The observation, that the success probability under effort constraints is greater than the success probability under time constraints, is due to the linear relationship between effort and time we assumed under effort constraints in (***). We make the note that, in contrast to [1], we did not assume a linear relationship between time and effort under time constraints.

Furthermore, to derive our second group of findings, for each cost driver and scale factor, we constructed two portfolios. The first one had this driver/cost factor rated as very high for all projects and the other portfolio had it rated as very low for all projects. For example, we found that when selective reuse [4] was practiced in ERP projects, the probability of success was higher under both time and effort constraints. For the purpose of illustrating this point, we report on the results (see Table 1) yielded when constructing two portfolios of subprojects, namely the first one with the factor of REUSE rated as *very high* for all subprojects and the second one with REUSE rated very low for all subprojects. (We make a note that *low* level of reuse in an ERP project indicates massive customization of the standard components and that a high level of reuse indicates limited customization [4]). Table 1 suggests that when a project is composed of subprojects all of which have REUSE rated *very high*, the probability of success is greater under both time and effort constraints. We observed that 13 out of the 17 factors from the COCOMO II model can be adjusted in a way that maximizes the probability of success.

Table 1. Analysis of the probability of success for the factor REUSE under effort constraints and time constraints.

REUSE rating	Probability of success	
	Under effort constraints	Under time constraints
Very low	68.78%	76.52%
Very high	96.87%	98.88%

Regarding our third group of findings, we observed that bundling ERP projects as a portfolio had the advantage over managing projects separately in terms of ability to explicitly and systematically approach uncertainty. Table 2 and Table 3 compare the probability of success for projects under effort constraints and for projects under time constraints, respectively. They indicate that the probabilities of success for projects with high uncertainty ratings are greater when those projects are managed as a portfolio.

Table 2. Increase in probability of success for low and high uncertain projects under effort constraints.

Uncertainty level	Probability of success		Ration of increase (a)/(b)
	Individual projects (a)	Portfolio (b)	
Low uncertainty	93.78%	98.81%	1.05
High uncertainty	84.31%	97.76%	1.16

Table 3. Increase in probability of success for low and high uncertain projects under time constraints.

Uncertainty level	Probability of success		Ration of increase (a)/(b)
	Individual projects (a)	Portfolio (b)	
Low uncertainty	15.76%	87.52%	5.55
High uncertainty	8.31%	75.91%	9.13

4. LIMITATIONS OF THIS WORK

This work is an initial step towards developing a solution to counterpart the ERP effort estimation challenges we addressed in the Introduction. Needless to say, for the findings of this study to be consolidated and transformed into recommendations to ERP project managers, a few replication studies must be done first. To this end, we are faced with the following validity [7] threats:

First, the major threat to external validity arises from the fact that the company's projects might not be representative for the entire population of ERP adopters. We however, believe that our project context is typical for the telecommunication companies in North America. We judge these settings typical because they seemed common for all SAP adopting organizations who were members of the American SAP Telecommunications User Group (ASUG). The ASUG meets on regular basis to discuss project issues and suggest service-sector-specific functionality features to the vendor for inclusion in future releases. The SAP components our case company implemented are the ones which other ASUG companies have in place to automate their non-core processes (accounting, inventory, sales& distribution, cell site maintenance).

Next, we deployed complementary three models of three types. However, we are aware that there are other promising effort estimation modeling techniques by each type. For example, there is a number of approaches using portfolio concepts [10,11] which might be good candidates for the ERP settings. In the future, we plan to investigate whether different modeling choices sustain our results or limit the validity of our findings to the subset of analyzed models.

5. CONCLUSIONS

This paper reported on a first experiment of approaching the problem of ERP cost estimation by using a combination of three concepts complementing each other's strengths. The targeted effect was to systematically cope with two aspects inherent to ERP project contexts: (i) uncertainty of cost drivers and (ii) strong bias by vendors and consultants in cost estimation.

We found this approach to be one good alternative to ERP-adopters as they no longer have to live with whatever estimates are given to them by ERP vendors and consultants. We also made an attempt to achieve an increased probability of success for highly uncertain ERP projects, a company may have to implement. Our early results suggest that when managed as a portfolio, these ERP projects have a greater chance to succeed under time and under effort constraints. This finding converges with the conclusions drawn by Jiamthubthugsin and Sutivong [5]. However, our results are preliminary only and we acknowledge that related validity concerns [7] remain our most important issue. We plan a series of experiments, action research and three case studies to test our approach. The results will serve to properly evaluate its validity and come up with an improved version of our method.

6. REFERENCE

[1] Boehm B., Software Cost Estimation with COCOMO II, Prentice Hall, Upper Saddle River, NJ, 2000.

[2] Chulani, S., Boehm, B. and Steece, B. Bayesian Analysis of Empirical Software Engineering Cost Models, IEEE Trans on SE, 25(4), 573-583.

[3] Daneva, M., ERP Requirements Engineering Practice: Lessons Learnt, IEEE Software, 21(2), 2004, 26-33.

[4] Daneva M., Measuring Reuse of SAP Requirements: a Model-based Approach, Proc.of Symp.on Software Reuse, ACM Press, NY, 1999, pp. 141-150.

[5] Jiamthubthugsin, W., D. Sutivong, Protfolio Management of Software Development Projects Using COCOMO II, Proc. of ICSE06, 889-892.

[6] McDonald, P., Giles, S. and Strickland, D., Extensions of Auto-Generated Code and NOSTROMO Methodologies, Proc. of 19th Int. Forum on COCOMO, Los Angeles, CA, 2004.

[7] Fenton N.E, S. L. Pfleeger, Software Metrics: A Rigorous and Practical Approach, PWS Publishing, 2nd ed. 1998.

[8] Fewster, R.M. and Mendes, E., Portfolio Management Method for Deadline Planning, Proc. of METRICS03, IEEE, 325-336.

[9] Stensrud E., Alternative Approaches to Effort Prediction of ERP Projects. Inf.&Soft Techn, 43(7), 2001, 413-423.

[10] Taudes A., M. Feurstein, A. Mild, Options Analysis of Software Platform Decisions: a Case Study, MIS Quarterly, 24(2), 227-243.

[11] Verhoef, C., Quantitative IT Portfolio Management, Science of Computer Programming, 45(1), 2002, 1-96.