The Influence of Leadership and Work Culture on Software Cost Effort

Khaled Hamdan
UGRU, UAE University, Al Ain, UAE
khamdan@uaeu.ac.ae

Boumediene Belkhouche
College of Information Technology, UAE University
Al Ain, UAE
b.belkhouche@uaeu.ac.ae

Peter Smith
University of Sunderland, Sunderland
peter.smith@sunderland.ac.uk

Abstract—**Globalization is adding more dimensions to software effort estimation process. The notions of leadership and culture carry with them highly variable assumptions, and thus, must be explicitly modeled. A new model that incorporates leadership and culture is proposed, elaborated and validated. A survey was undertaken to determine the impact of culture and its effect on the software development process in the areas of project team timeliness, collaboration and team work, leadership characteristics, cultural intelligence, motivation and communication.. The use of the Bootstrap method for estimating the effort involved in a given project, along with analogies using real historical data, demonstrates the effectiveness of this approach in surmounting difficulties in describing abstract quantitative variables. Our approach is tested on a cluster sample dataset of 41 cases (projects) collected in 2007 from more than 20 organizations. The results show that the inclusion of leadership and culture in the cost estimation model improves the accuracy of software cost estimation.**

Index Terms—**Effort estimation, Leadership, Team culture, CBR, Bootstrapping, Ontology**

I. Introduction

Software development teams are becoming less homogeneous and more distributed as a result of globalization. In this new setting, team members tend to possess diverse backgrounds, thus affecting the dynamics and quality attributes of the team. These attributes play a significant role in determining the cost and quality of software projects ([8]; [22]). Moreover, the behavior of an organization and its productivity depend highly on the culture and leadership among the members of the organization ([14]; [27]; [12]; [33]; [28]). Consequently, culture and leadership become critical parameters for software cost estimation.

There has been a continuous search for better models and tools to aid project managers in the cost estimation process ([18]; [17]; [13]). Commonly software cost estimates are based on various methods, such as: Algorithmic Estimation Models COCOMO [8]; SLIM [26], and Function Points [4], Expert Judgment [16] and Case-Based Reasoning ([24]; [1]; [31]). Jørgensen and Shepperd [17] identified over 300 papers on software cost estimation. Many attempts have also been made to identify the effect of individual differences in software developers [22]. However, not enough attention is explicitly given to leadership and cultural issues.

Given the multitude of cost estimation approaches, we chose to concentrate our research on analogy methods, such as those proposed by Shepperd and Schofield [32], and specifically on case-based reasoning (CBR). CBR is an approach used to improve effort estimation by understanding and measuring the similarity between cases ([31]; [5]; [23]). CBR tries to predict an outcome by finding similar cases to the current problem ([1]; [30]). The major strategy of CBR is capturing previous experiences into a case database in order to propose solutions to new problems (cases). The database of past projects is used as a reference point in order to combine actual costs of previous projects for the prediction of the costs of a new project with similar attributes. CBR can be applied either at the project level as a whole, or at the sub-system level.

Our research investigates the hypothesis that organizational culture and project leadership are significant factors in determining accurately the cost of software projects development within the Arab Gulf States. In this region, there is a rapid expansion of IT infrastructure and services, and a generalized use of expatriate labor. These states share key similarities, but differ significantly from the rest of the world [2]. Besides culture, a major difference between Arab leadership and leadership in other nations is to be found in Arab authority values. Our ultimate goal was to develop a

Contact: khamdan@uaeu.ac.ae

doi:10.4304/jsw.8.6.1353-1367

CBR-based cost estimation model that incorporates leadership and culture parameters. Our approach consisted of several stages. In the first stage, a survey of software development projects within government departments in United Arab Emirates (UAE) was undertaken [15]. In stage two, the analysis of the survey highlighted several parameters, and specifically leadership and culture, which have an impact on cost estimation in this area. In stage three, software cost estimation ontology was developed to provide guidance and to reduce ambiguities. In stage four, a cost estimation model was proposed and its implementation was elaborated. The final stage encompassed the evaluation of our cost estimation model.

Our paper is structured as follows. Section 2 provides a review of the role of leadership and culture in software development. We detail the surveys undertaken, and provide a statistical analysis of the data collected from various organizations. Based on this analysis the model parameters are identified. Section 3 presents the new software effort estimation model. This model augments the CBR model by incorporating culture and leadership. The results and evaluation of the model are presented and discussed. Section 4 uses an example to illustrate our method. Section 5 describes the implementation of the model.

II. CULTURE AND LEADERSHIP

The research literature indicates that leadership and culture are strongly related. Culture plays an important role in people's lives in general, and in organizations in particular. It comprises factors such as knowledge, beliefs, values, traits, experiences, language and religion that make up a community, lifestyle and its way of thinking ([29]; [25]). Numerous authors ([14]; [33]; [28]; [29]) emphasize the role of culture on the organization's propensity for learning. Team performance, behavior and attitudes within and outside the organization are affected by culture. Similarly, leadership impacts productivity. Gardner [14] defines leadership as the process of influencing others to achieve a task by providing purpose, direction and motivation. Hence, to achieve consistent success, it is necessary for organizations to promote a positive, inclusive culture and a supportive leadership.

In an attempt to identify attributes characterizing culture and leadership, we interviewed administrators, managers and project leaders and we surveyed the relevant literature. Figure 1 summarizes our synthesis.

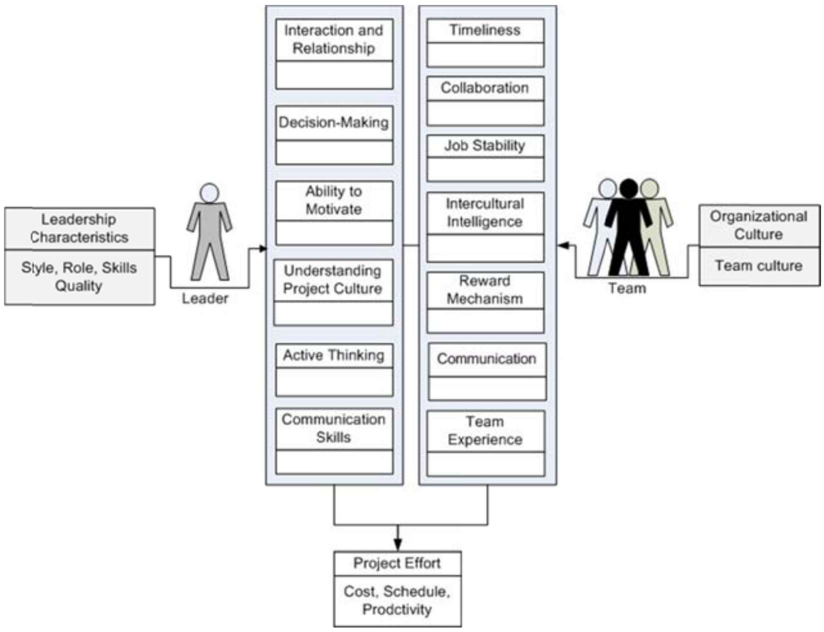


Figure 1. Culture and Leadership Parameters

Based on these preliminary findings, a survey to assess leadership and culture was carried out. Each leadership characteristic was categorized into four sub-items that were rated 1 to 9 by the respondents. Then for each of the six main characteristics the average of the sub-items rating was calculated. Project team culture characteristics were similarly assessed. Thirty eight projects were analyzed. A student's t-test and a one-way analysis of variance (ANOVA) were used to determine significant differences in project attributes according to type of project and organization. Associations between attributes were assessed using Pearson's correlation for quantitative data and the Chi-square test of independence for qualitative data. The Kolmogorov-Smirnov test was used to assess the normality of data. Descriptive statistics for leadership and culture variables are shown below (Tables 1 and 2).

TABLE I.
LEADERSHIP CHARACTERISTICS

	Mean	Median	Std. Deviation	Minimum	Maximum
Interaction and Relationship	6.9500	7.0000	1.2015	2.5000	9.0000
Decision-Making	6.8950	7.0000	1.2564	3.8000	9.0000
Ability to Motivate	6.5250	7.0000	1.3985	3.5000	8.8000
Understanding Project Culture	6.7150	7.0000	1.3552	2.8000	8.3000
Active Thinking	7.2450	7.5000	1.1507	2.0000	8.5000
Communication skills	7.2150	7.6500	1.0890	3.8000	9.0000

TABLE II.
TEAM CULTURAL CHARACTERISTICS

	Mean	Median	Maximum	Minimum	STD Deviation
Timeliness	6.6658	7.0000	8.5000	1.7500	1.4575
Collaboration	6.9750	7.0000	8.5000	3.0000	1.3447
Job Stability	6.7303	7.0000	8.7500	3.2500	1.2058
Intercultural Intelligence	7.0539	7.1250	8.2500	5.0000	1.0335
Reward Mechanism	6.4298	7.0000	8.0000	2.3333	1.2566
Communication	7.1184	7.5000	9.0000	3.5000	1.3187
Team Experience	6.6461	7.0000	8.2500	3.8000	1.1155

In general, the means and the medians of all the leadership and cultural characteristics are quite high indicating their importance. Regarding the correlations between the leadership characteristics, Table 3 shows that all these characteristics correlate highly ($p<0.001$). The cultural characteristics also show strong correlations (see Table 4). The only exception is 'Team Experience' which seems to correlate only with 'Reward Mechanism' and 'Communications'. It is also interesting to see the correlation between leadership and cultural characteristics (see Table 5). 'Team Experience' correlates only with 'Decision-Making' and 'Communication Skills'. Some leadership and cultural characteristics appear to be more important than others. These characteristics were believed by the respondents to be significant attributes in most cases. This is probably due to the fact that these are innate attributes which are part of the individuals' characters which have been shaped by interaction with others and by life experience in the community.

TABLE III.
CORRELATION BETWEEN LEADERSHIP CHARACTERISTICS

		Interaction and Relationships	Decision Making	Ability to Motivate	Understanding Organisation Culture	Active Thinking	Communication skills
Interaction and Relationships	Pearson Correlation	1	.743**	.677**	.760**	.750**	.789**
	Sig. (2-tailed)		.000	.000	.000	.000	.000
Decision-Making	Pearson Correlation	.743**	1	.714**	.717**	.743**	.884**
	Sig. (2-tailed)	.000		.000	.000	.000	.000
Ability to Motivate	Pearson Correlation	.677**	.714**	1	.779**	.607**	.645**
	Sig. (2-tailed)	.000	.000		.000	.000	.000
Understanding Organisation Culture	Pearson Correlation	.760**	.717**	.779**	1	.688**	.648**
	Sig. (2-tailed)	.000	.000	.000		.000	.000
Active Thinking	Pearson Correlation	.750**	.743**	.607**	.688**	1	.793**
	Sig. (2-tailed)	.000	.000	.000	.000		.000
Communication skills	Pearson Correlation	.789**	.884**	.645**	.648**	.793**	1
	Sig. (2-tailed)	.000	.000	.000	.000	.000	

**. Correlation is significant at the 0.01 level (2-tailed).

TABLE IV.
CORRELATION BETWEEN CULTURE CHARACTERISTICS

		Timeliness	Collaboration	Job Stability	Intercultural Intelligence	Reward Mechanism	Communication	Team Experience
Timeliness	Pearson Correlation	1	.700**	.791**	.693**	.520**	.598**	.286
	Sig. (2-tailed)		.000	.000	.000	.001	.000	.073
Collaboration	Pearson Correlation	.700**	1	.694**	.667**	.686**	.666**	.170
	Sig. (2-tailed)	.000		.000	.000	.000	.000	.295
Job Stability	Pearson Correlation	.791**	.694**	1	.752**	.449**	.476**	.133
	Sig. (2-tailed)	.000	.000		.000	.004	.002	.413
Intercultural Intelligence	Pearson Correlation	.693**	.667**	.752**	1	.443**	.453**	.110
	Sig. (2-tailed)	.000	.000	.000		.004	.003	.498
Reward Mechanism	Pearson Correlation	.520**	.686**	.449**	.443**	1	.584**	.303
	Sig. (2-tailed)	.001	.000	.004	.004		.000	.057
Communication	Pearson Correlation	.598**	.666**	.476**	.453**	.584**	1	.467**
	Sig. (2-tailed)	.000	.000	.002	.003	.000		.002
Team Experience	Pearson Correlation	.286	.170	.133	.110	.303	.467**	1
	Sig. (2-tailed)	.073	.295	.413	.498	.057	.002	

**. Correlation is significant at the 0.01 level (2-tailed).

TABLE V.
CORRELATION BETWEEN CULTURE AND LEADERSHIP CHARACTERISTICS

		Timeliness	Collaboration (Interpersonal Relation)	Job Stability	Intercultural Intelligence	Reward Mechanism	Communicatior	Team Experience
Interaction and Relationships	Pearson Correlation	.692**	.690**	.738**	.571**	.458**	.585**	.165
	Sig. (2-tailed)	.000	.000	.000	.000	.004	.000	.322
Decision-Making	Pearson Correlation	.677**	.849**	.628**	.529**	.571**	.706**	.338*
	Sig. (2-tailed)	.000	.000	.000	.001	.000	.000	.038
Ability to Motivate	Pearson Correlation	.598**	.790**	.610**	.472**	.666**	.487**	.092
	Sig. (2-tailed)	.000	.000	.000	.003	.000	.002	.583
Understanding Organization Culture	Pearson Correlation	.714**	.773**	.741**	.644**	.690**	.449**	.132
	Sig. (2-tailed)	.000	.000	.000	.000	.000	.005	.430
Active Thinking	Pearson Correlation	.609**	.750**	.558**	.398**	.538**	.591**	.242
	Sig. (2-tailed)	.000	.000	.000	.013	.000	.000	.144
Communication Skills	Pearson Correlation	.580**	.776**	.501**	.469**	.611**	.701**	.345*
	Sig. (2-tailed)	.000	.000	.001	.003	.000	.000	.034

**. Correlation is significant at the 0.01 level (2-tailed). *.Correlation is significant at the 0.05 level (2-tailed).

III. AN AUGMENTED CBR MODEL

The significance of our approach to CBR is the inclusion of culture and leadership in our model (Figure 2).

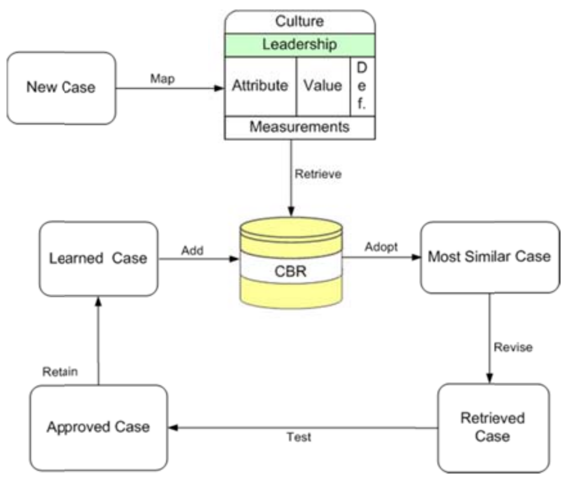


Figure 2. The Augmented CBR

The augmented CBR incorporates specific cultural and leadership characteristics, factors, and issues that were identified as influencing cost estimation. The identified parameters have been categorized into seven groups (see Figure 3): organization line of business, application type, organization type, organizational culture, project leadership, project technical environment, and year of project completion.

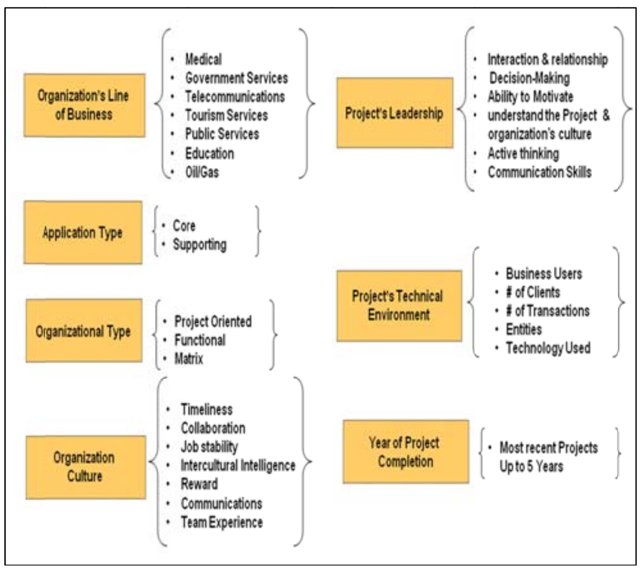


Figure 3. Model Parameters

The cultural and leadership characteristics are measured using a nine-point type scale, where 1 means Not Influential at all and 9 means Highly Influential. The remaining parameters are coded as follows. The Organization Line of Business was measured by a seven-point type scale (code 1 to 7): Medical, Governmental Services, Communication, Public Services, Tourism Services, Education and Oil and Gas. The Application Type was measured on a two-point scale (code 1, 2) with 1 being Core systems and 2 being Support systems. The Organizational type was measured on a three-point type scale (code 1 to 3) with 1 being Project Oriented (project manager has the highest power in making decisions), 2 being Matrix (Project Manager has moderate power in making decisions), and 3 being Functional (project manager has lowest level of power in making decisions). The Project Technical Environment parameter is measured according to the number of Core Users (Backend Users), number of Clients, number of Transactions, numbers of Entities, and Technology (Hardware and Software Infrastructure). The Year of Project Completion measures the duration of the project.

A component of our CBR approach is the Estimation by Analogy method (EBA). The key idea behind the EBA is that similar input data vectors have similar output values [5]. A number of nearest neighbors is sought according to a distance metric to determine the output approximation. The estimation of the outputs is calculated by using the average of the outputs of the neighbors (analogies). It is a procedure consisting of three steps. First, the new class for which the project effort is to be estimated is characterized by a set of attributes common to the ones characterizing previous projects in a historical database. Second, one or more similar projects (neighbors or analogies) from the dataset are identified. Similarities and differences between the different projects' features and the source case that is nearest the target are identified by measuring the distance between cases. Finally, the values of the neighbor projects are used to produce the estimate (usually by computing their mean). A sample data structure for representing the cases is shown in Figure 4.

	Effort	*Attribute 1*	*Attribute 2*	…	*Attribute k*
Case 1	E_1	X_{11}	X_{12}	…	X_{1k}
Case 2	E_2	X_{21}	X_{22}	…	X_{2k}
….	…	…	…	…	…
Case n	E_n	X_{n1}	X_{n2}	…	X_{nk}
New Case	Unknown	Y_1	Y_2	…	Y_k

Figure 4. The data structure [5]

A. Effort Estimation Procedure

The following 3-stage procedure (Figure 5) is used to estimate the amount of effort for the new project.

1. Extract historical cases that are most similar to the current one according to a selected metric.
2. Estimate the response variable "effort" for the new case based on the extracted cases.
3. Estimate the precision and carry out validation.

a. Use the Bootstrap method to get an estimate of the standard deviation :

- Select (with replacement) the bootstrap samples.
- Determine the bootstrap replicates of the median (or the mean).
- Compute the standard deviation of the bootstrap replicates of the median (or the mean).

b. Use the Bootstrap method to get an estimate of the bias and carry out validation.

- Compute the bootstrap replicates of the bias for the median (or mean) as the difference between each bootstrap replicate of the median (or mean) and the sample median (or mean) of the dataset used.
- Compute the bootstrap estimate of the bias as the average of the bootstrap replicates of the bias. The mean bias either shows the over estimation (+) or underestimation of the effort (-). A positive value of the mean bias represents overestimation and a negative value represents underestimation.
- Validate by correcting for the bias according to its sign (+/-).

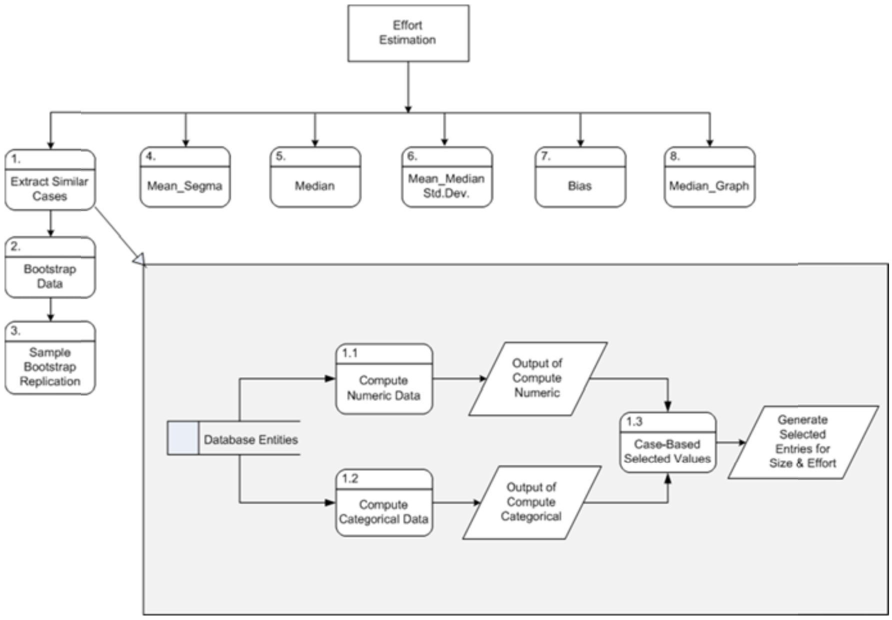


Figure 5. Effort Estimation Procedure

B. Similar Cases Extraction

Step 1 uses the k-nearest neighbor algorithm to identify similar cases [3]. The unweighted Euclidean distance measure is the most popular and straightforward distance measure that has been previously used with encouraging results in software engineering cost estimation studies ([6]; [32]). When there are mixed (numerical and categorical) data, a dissimilarity measure is also computed. An advantage of this measure is that it takes into account the missing values of the projects as well. In such a case, the distances are usually calculated using only the available attributes.

In the following, it is assumed that the new project is represented by a vector of attributes $(Y_1, Y_{2,} \dots Y_n)$ and every project i by the vector $(c_{i1}, \dots c_{ik})$ as shown in Figure 4.The distance is then computed using the formula shown in Figure 6.

$$\mathrm{Sim}(c_1, c_2, F) = \frac{1}{\sqrt{\sum j \in P \text{ Feature\_Dissimilarity}(c_{1j}, c_{2j})}}$$

$F = \text{Number of features}$
$c_1, c_2 = \text{Cases}$

$$\text{Feature\_Dissimilarity } (C_{1j}, C_{2j}) \begin{cases} (C_{1j} - C_{2j})^2 & \textit{Features are numeric} \\ 0 & \textit{Features are categorical} \quad \textit{and } C_{1j} = C_{2j} \\ 1 & \textit{Features are categorical} \quad \textit{and } C_{1j} \neq C_{2j} \end{cases}$$

Figure 6. Nearest Neighbour Algorithm [32]

Determining similar projects that operate in a similar organizational culture and ascertaining an acceptable level of similarity among projects with different organizational cultures is challenging. For example, projects A and B may be similar in decision-making, ability to motivate team members, communications, and project manager experience, yet they may be different in organizational culture. In such a case, the difference in understanding culture can be substituted by a buffer time. The magnitude of impact needs calibration after further studies. For example, at 4-6 on the scale of variables, 10% more man-days need to be added to the new project to accommodate individual differences [20]. The weight on the scale is assigned according to the effectiveness of each parameter in the process.

C. Response Variable Estimation

Once similar cases are identified, the unknown effort (of the new case) is estimated by a location statistic (mean, median) of those “neighbor” cases. The use of the median is motivated by two reasons:

1. The data we are using here exhibit generally skewed distributions (possibly caused by outliers). This in turn gives biased estimates, and the median is known to be affected less than the mean by this phenomenon.
2. The use of the median for inference purposes requires fewer assumptions than that of the mean.

D. Bootstrapping

The Bootstrap method is used to implement the third step in order to adjust the bias of the EBA estimate of the actual effort. This technique is used to estimate the standard error (SE), bias, confidence intervals, and other measures of statistical accuracy [11]. It provides re-sampling methods that are widely used to estimate parameters and evaluate the biases and the errors of estimation with little or no assumptions about the underlying distribution.

In the Bootstrap method, the original sample with a distribution F is sufficiently replicated and the expanded sample with a distribution $\hat{F}$ is used as the new population. The success of this method is partially due to the fact that $\hat{F}$ is consistent with F. A sample drawn from the population is used to test the estimators. E.g., an unknown parameter $\theta = t(F)$ is estimated by $\hat{\theta} = s(x)$ on the basis of a random sample $x = (x_1, x_2, \ldots x_n)$ from the probability distribution F. The procedure used by the Bootstrap method to estimate the standard error of s(x) is shown in Figure 6. Given the observed dataset $x = (x_1, x_2, \ldots x_n)$, a statistic of interest $s(x)$ to compute, a B bootstrap replication of s, say $s(x^*_1), s(x^*_2), \ldots s(x^*_B)$, where B , the number of replications, is a some large number, e.g., around 1000.

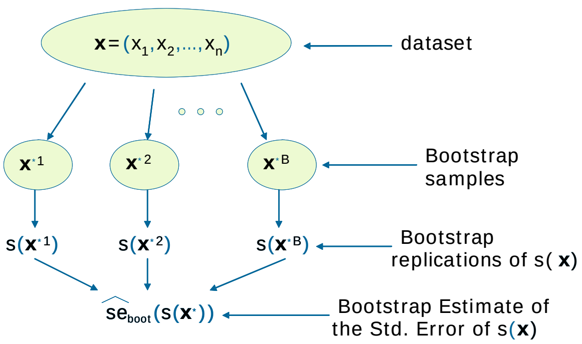


Figure 7. Standard Bootstrap

Here $\widehat{se}_{boot}(s(x^*))$ is the sample standard deviation of $s(x^{*k})$, $k = 1,2 \ldots, B$. Generally, the Bootstrap method gives adequate results for B between 25 and 200 [11]. A bootstrap sample $x^* = (x^{*1}, x^{*2} \ldots x^{*B})$ is a random sample of size n drawn from $\widehat{F}$. For the Bootstrap standard error estimate, it is known that $\lim_{B\to\infty} \widehat{se}_{boot}\, s(x^*)) = Se_F(\hat{\theta}^*)$. The limit is an ideal, though not fully accurate, estimate of the S.E. of $s(x^*)$. In theory, Bootstrap estimates the standard deviation of the sample median. This is normally given by the standard error of the sample medians(x), i.e., using the distribution of the median which is not obvious. The Bootstrap estimate of the standard error (calculated for the sample median) gives an easy and practical answer.

IV. Measure Accuracy and Validation

The model was tested on a number of governmental development projects in order to determine its accuracy and appropriateness. Results suggest that closer estimates are obtained when cultural and leadership attributes are included in the estimation model. Specifically, the estimation of actual effort improved in 90% of the support system projects and in 50% of the core system projects, when leadership and cultural attributes were added.

We used the jack-knife method to evaluate the predictive accuracy for our approach [21]. This validation method is an effective useful tool for assessing the error of the prediction procedure. Given a set of completed cases, one of the cases (say the ith case) is removed from the dataset and the remaining cases are used as a basis for the estimation of the removed case.

Two measures of local error (Table 6) are calculated:

1. The magnitude of relative error (MRE) [9].
2. The magnitude of relative error to the estimate (MER) [19].

TABLE VI.
Local Accuracy Measures

$MRE = \frac{\lvert E_A - E_E \rvert}{E_A}$	$MER = \frac{\lvert E_A - E_E \rvert}{E_E}$

The local measures are the basis for the estimation of the global predictive accuracy measures MMRE, predmre25, MMER and predmer25 (Table 7).

TABLE VII.
Global Accuracy Measures

$MMRE = \frac{1}{n}\sum_{i=1}^{n} \frac{\lvert E_{Ai} - E_{E_i} \rvert}{E_{Ai}}$	$MMER = \frac{1}{n}\sum_{i=1}^{n} \frac{\lvert E_{Ai} - E_{E_i} \rvert}{E_{E_i}}$
$pred_{mre}25 = \frac{\#(\text{cases with } MRE \leq 0.25)}{\#(\text{cases})}$	$pred_{mer}25 = \frac{\#(\text{cases with } MER \leq 0.25)}{\#(\text{cases})}$

If the MMRE (or MMER) is small, then these are a good set of predictions. A usual criterion for accepting a prediction method as good is that it has a $MMRE \leq 0.25$ (similarly for MMER). The opposite is the case for the predmre25 (or predmer25) accuracy measure. A standard criterion for considering a method as acceptable is $pred_{mre}25 \geq 0.75$ (similarly for predmer25).

In order to select the appropriate number of analogies the jack-knife technique is applied from one up to ten analogies and the MMRE, MRE, and pred25 accuracy measures are calculated for each of the cases in the whole dataset. It was decided to use one analogy for the predictions, i.e., a number that minimized the MMRE and gave relatively reasonable results for the measures.

The dataset was split according to application type: “Supporting” applications are the systems which support the internal (shared) services in any organization. These applications are not linked directly to the organization mission and vision; rather they enhance the efficiency, effectiveness, and the performance of the supporting services. Those systems share similar features across the government departments.

“Core” applications exist to help to achieve the mission and vision of the organizations and to satisfy their core purpose. The features of these applications are unique. Organizations with a similar line of business could share similar features.

Next, the (core-support) models were intended to measure predictive accuracy (MMRE, Pred25) with and without cultural and leadership characteristics in the split cases. The functionalities of these systems are different and should be treated separately. The analogy showed significant differences between cases for the support systems of the cases including cultural and leadership characteristics which improved the analogy. The core applications improved the analogy by 50 percent when the two highest effort cases were removed. There are 19 core projects and 17 support projects in which there are no missing values for the dependent variable.

In the presence of correlated independent variables, the regression coefficient may not be meaningful. The negative coefficients in equations do not reflect the true effects of independent variables. The fitting accuracy of the model is presented in Table 8 and Table 9. In order to evaluate the predictive accuracy, the jack-knife procedure was used. Then two different MMRE were calculated:

1. The "fitting" MMRE: this is calculated by the regression procedure in SPSS. The "Unstandardized" predicted values are computed for the data that were used to fit the model and these are in fact the predicted logarithm of the efforts. The computed MREs are therefore given by:

$$\left| \frac{e^{predicted} - e^{LN(effort)}}{e^{LN(effort)}} \right|$$

Next, the mean of all MREs gives the MMRE.

2. The "predictive" MMRE: this is computed when the jack-knife procedure is applied and it can be also be computed in SPSS by the "deleted" residuals. These residuals (say r) are computed as the differences r =ln(effort) - predict; but here the prediction is made for each case when this is deleted from the data. So by computing first: Predicted = ln(effort) - r

The jack-knife MRE is:

$$\text{MRE=} \left| \frac{e^{predicted} - e^{LN(effort)}}{e^{LN(effort)}} \right|$$

The mean and median of all MRE is the predictive MMRE. After calculating both MMRE and MMER, the corresponding pred25 measure for them.

TABLE VIII.
ACCURACY MEASURES FOR THE LINEAR REGRESSION MODEL
(SUPPORT SYSTEMS)

		MMRE	MdMRE	MMER	MdMER	predMRE25	predMER25
Regression	**Fitting Accuracy**	20%	17%	21%	19%	79%	68%
	Predictive Accuracy	56%	47%	57%	45%	16%	26%
EbA best model (n=8)	**Predictive Accuracy**	129 %	120%	69%	68%	16%	17%

TABLE IX.
ACCURACY MEASURES FOR THE LINEAR REGRESSION MODEL
(CORE SYSTEMS)

		MMRE	MdMRE	MMER	MdMER	predMRE25	predMER25
Regression	**Fitting Accuracy**	24%	17%	25%	21%	74%	74%
	Predictive Accuracy	42%	28%	44%	30%	37%	42%
EbA best model (n=9)	**Predictive Accuracy**	102%	101%	62%	62%	22%	22%

The comparison of the two models shows that the linear regression model outperforms EBA for all support and core systems measures. On the other hand, analysis of the completed projects, including leadership and cultural attributes appears to provide better results. Regression and analogy performed better when cases were split and selected as core and support systems.

Two methods for estimating the actual effort and total cost both for core and support system projects were presented and their accuracy was evaluated. Results suggest that better estimates are obtained when cultural and leadership attributes are included in the estimation model. Specifically, the estimation of actual effort and cost accuracy improved drastically for both support and core systems, when leadership and cultural attributes were added. Total cost may be used as alternative evaluation for software effort estimation due to its importance and significance in predicting the cost model.

The fitting accuracy of the model is presented in Table 10. In order to evaluate the predictive accuracy, the jack-knife procedure was used. After applying linear regression on the project's leadership characteristics and project team culture attributes separately, it was concluded that a representative model for the dependent variable LNTotalCost could not be built.

TABLE X.
ACCURACY MEASURES FOR THE LINEAR REGRESSION MODEL (LNTOTALCOST)

		MMRE	MdMRE	MMER	MdMER	predMRE25	predMER25
Regression	**Fitting Accuracy**	57.37%	39.32%	53.53%	41.36%	29.00%	24.00%
	Predictive Accuracy	72.07%	43.35%	68.33%	49.37%	21.95%	21.95%
EbA best model (n=7)	**Predictive Accuracy**	79.65%	49.13%	76.55%	52.96%	24.39%	34.15%

Estimation by analogy (EBA) is another technique for the prediction of a dependent variable. Various neighbors were tried out and the results of the jack-knife procedure are presented in Table 11. As observed, the optimal number of neighbors varies according to the accuracy measure that needs to be optimized. It would appear that '7 neighbors' is a good choice for the construction of an EBA model.

TABLE XI.
PREDICTIVE ACCURACY MEASURES FOR THE EBA MODEL (LNTOTALSKILLCOST)

No of Neighbors	MMRE	MdMRE	MMER	MdMER	predMRE25	predMER25
1	101.21%	67.34%	168.48%	63.68%	17.07%	17.07%
2	86.26%	52.34%	95.37%	52.99%	31.71%	26.83%
3	87.44%	55.17%	96.74%	58.34%	24.39%	21.95%
4	80.52%	56.65%	87.55%	53.40%	24.39%	24.39%
5	82.16%	55.41%	82.12%	57.97%	17.07%	19.51%
6	84.63%	54.43%	80.54%	62.73%	26.83%	24.39%
7	79.65%	49.13%	76.55%	52.96%	24.39%	34.15%
8	83.03%	53.23%	78.60%	52.89%	24.39%	24.39%
9	90.94%	58.60%	78.09%	49.24%	19.51%	21.95%
10	101.20%	58.33%	74.24%	45.16%	24.39%	21.95%
11	125.62%	64.80%	78.99%	52.94%	19.51%	21.95%
12	137.85%	78.91%	81.32%	55.49%	19.51%	21.95%
13	141.94%	79.94%	80.95%	55.19%	21.95%	19.51%
14	149.54%	80.75%	82.68%	56.84%	17.07%	17.07%
15	147.01%	80.36%	80.06%	56.82%	14.63%	19.51%
16	144.00%	73.10%	78.36%	55.52%	17.07%	19.51%
17	148.19%	71.72%	79.45%	56.80%	14.63%	17.07%
18	159.23%	73.38%	78.28%	60.85%	12.20%	17.07%
19	161.07%	75.69%	75.63%	62.47%	12.20%	14.63%
20	170.66%	75.14%	78.18%	62.33%	7.32%	9.76%

The comparison of the two models shows that the linear regression model outperforms EBA for MMRE, MdMRE, MMER and MdMER, whereas the opposite is true for the remaining measures. On the other hand, the parametric and non-parametric tests do not provide a statistically significant difference between these measures.

Assume y is a new case with actual effort 582. Regarding the evaluation of the predictive accuracy for EBA method, the jack-knife procedure was adopted [21]. First of all, take the absolute value of (actual – estimate) / actual. After applying analogy to estimate the last project (jack-knife y), analogy finds case 1 to be the most similar and reports that 320 is the estimate. However, the true value is 582. So, the relative error for analogy is (320 – 582) / 582. The MER will be calculated based on the procedure (actual – estimate) / estimate. So, the MER is abs((582 – 320) / 320) for the first project (see Table 12).

In order to select the appropriate number of analogies the jack-knife technique was applied from one up to ten analogies and the MMRE, MRE, and pred25 accuracy measures were calculated for each of the cases in the whole dataset. It was decided to use one analogy for the predictions, i.e. a number that minimized the MMRE and gave relatively reasonable results for the measures. The values of the effort for the selected cases were: 320, 105, 138, 324, 600, 750, 1250, 1295, and 1300. It appeared that '9 neighbors' is a reasonable choice for the construction of the EBA model.

TABLE XII.
THE EXAMPLE FOR THE DATASE

OrgNo	Project	Org Size	Org LOB	Org Type	Duration Mon	Tools Lang	DBMS	Decision Making	Commu-nications	Team Exp	Both C/L	Actual Effort	MRE	MER
29	Telematics system	60	4	2	16	1	2	7.0	7.8	6.0	1.00	320	45.0%	81.9%
28	E-Archive	4,000	2	3	11	1	2	8.0	8.3	6.0	0.50	1,300	123.4%	55.2%
38	Well Prognosis	2,296	7	2	7	3	3	7.8	7.8	6.8	0.48	600	3.1%	3.0%
4	Fuder (Data Manage	1,200	7	1	53	5	2	8.0	8.0	8.0	0.44	138	76.3%	321.7%
34	Al Ain Muncipality e	2,000	2	1	12	1	1	7.3	5.5	6.0	0.42	1,250	114.8%	53.4%
40	Daman Insurance A	2,296	7	2	4	2	2	7.8	8.0	6.8	0.40	324	44.3%	79.6%
30	Project Bus. Env.	200	8	1	24	2	1	7.3	7.8	6.0	0.40	1,295	122.5%	55.1%
15	Database Applicatio	16,000	2	2	6	2	2	7.0	8.0	7.0	0.39	750	28.9%	22.4%
1	Financial (JD Edwar	1,900	7	2	5	3	3	8.0	9.0	8.0	0.37	105	82.0%	454.3%
24	ERP	1,500	2	2	7	1	1	8.3	8.0	7.5	0.37	1,200	106.2%	51.5%
32	Planning and Devlop	167	2	1	16	1	1	7.5	7.5	7.5	6.0	582		

The execution of the macro using Minitab gave: s(x) = 600 and $\widehat{se}_{boot}(s(x^*))$ = 348. How good is this estimate? Compare it to $Se_{\hat{F}}(\hat{\theta}^*)$, which is obtained from the sampling distribution of $s(x^*)$ bootstrap replications (see Table 13).

TABLE XIII.
THE BOOTSTRAP ESTIMATE

	Bootstrap Estimate	
Row	median	of se
1	600	348.742

This section computes the standard of the bootstrap median based on all possible bootstrap samples, 9^9 samples [7].

$$p_i = P\{s(\mathbf{x}^*) = x_{(i)}\} = \sum_{j=0}^{4}\left[Bi\left(j; 9; \frac{i-1}{9}\right) - Bi\left(j; 9; \frac{i}{9}\right)\right]$$

The results are given in Table 14:

TABLE XIV.
THE SAMPLING DISTRIBUTION OF THE SAMPLE MEDIAN

x(i)	*p(i)*
320	0.0014493
105	0.0289240
138	0.1144725
324	0.2206611
600	0.2689862
750	0.2206611
1250	0.1144725
1295	0.0289240
1300	0.0014493

The previous distribution gives $Se_{\hat{F}}(\hat{\theta}^*)$ = 349.5 which is very close to the bootstrap estimate obtained earlier. To have more insight about these estimates, histograms were constructed of the 200 replications used in the bootstrap estimate and the histogram of 200 observations generated from the previous distribution of $s(x^*)$. These turned out to be similar (see Figure 8).

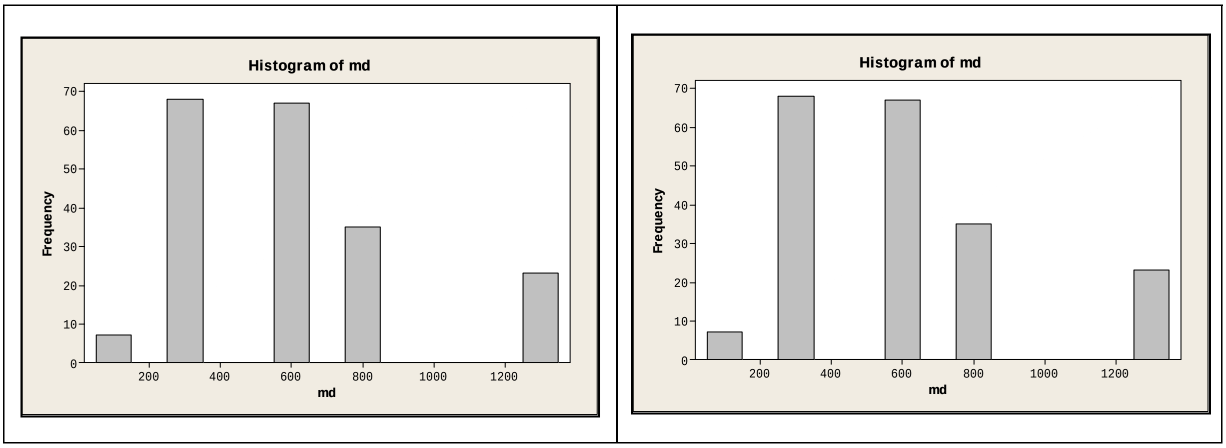


Figure 8. Simulated histogram and Bootstrap histogram of the median distribution

The Bootstrap estimate of the median bias was used to estimate the bias of the sample median (which is 600 here, as seen earlier). A Minitab macro (bootstrap bias), was written to display the bias bootstrap replications along with the sample median and its bootstrap estimate of the bias. The execution of the macro bootstrap bias gave $\widehat{bias}_{boot}(s(x^*))$ = 18.5 which is the bootstrap estimate of the median bias. The Bootstrap estimate of the bias is as shown (see Table 15):

TABLE XV.
THE BOOTSTRAP BIAS ESTIMATE

	bootstrap	
Row	estimate of the median	bias
1	600	18.52

This positive value suggests that the sample median (of 600) overestimates the actual median, and needs correction. A validation operation is then needed and the corrected median is then s(x) corrected = 600 – $\widehat{\text{bias}}_{\text{boot}}(s(x^*))$ = 600 – (18.5). So the estimate (prediction in fact) of the "effort" for the new project is: s(x) corrected = 581.5 (very close to the actual effort 582 and the median effort 600). This positive value suggests that the sample median (600) overestimates the actual median, and needs correction. The corrected median is then 582 man-days. The above computations show the extent of the simplicity and usefulness of the bootstrap method in estimating the median effort and its standard error.

The bootstrap subsystem model was developed to enable a simple way of predicting the corrected median effort [10]. This validates the project estimation based on the entry of original analogy effort data, bootstrap sample replication size and effort values. It displays the median bootstrap replications along with the sample median and its standard error. The input interface of the system consists of:

- effort data (number of analogy),
- bootstrap sample replication size value,
- effort values.

The system outputs are replicates of data, mean, median and bias (see Figure 9).

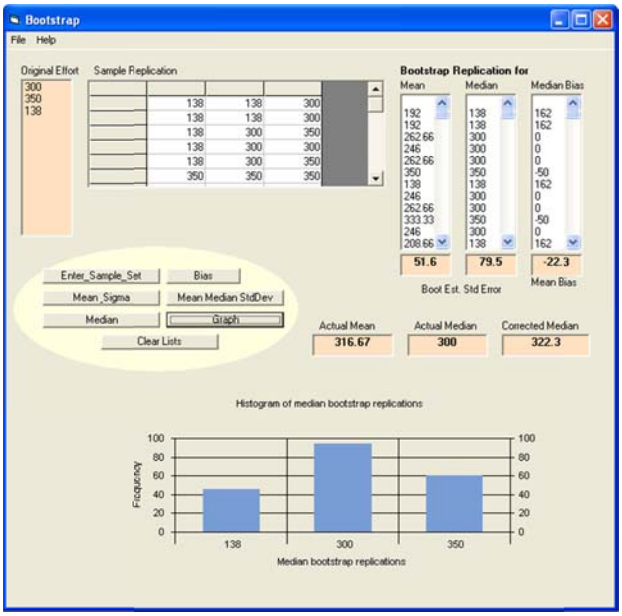


Figure 9. Bootstrap user interface

V. IMPLEMENTATION OF THE SYSTEM

A software tool, called SEEOS (software effort estimation ontology system) that supports the application of an analogy-based method was implemented. This tool provides a flexible interface that allows users to experiment with different project characteristic options. The main functions of SEEOS are the following: (1) defining comprehensive attributes for a project; (2) defining attributes, characteristics and measurement protocol; (3) providing the choice of options to be considered such as cultural factors and leadership; (4) determining which attributes are available to provide better accuracy; and (5) generating most similar projects for the required estimate. The structure of the query system is shown in Figure 10.

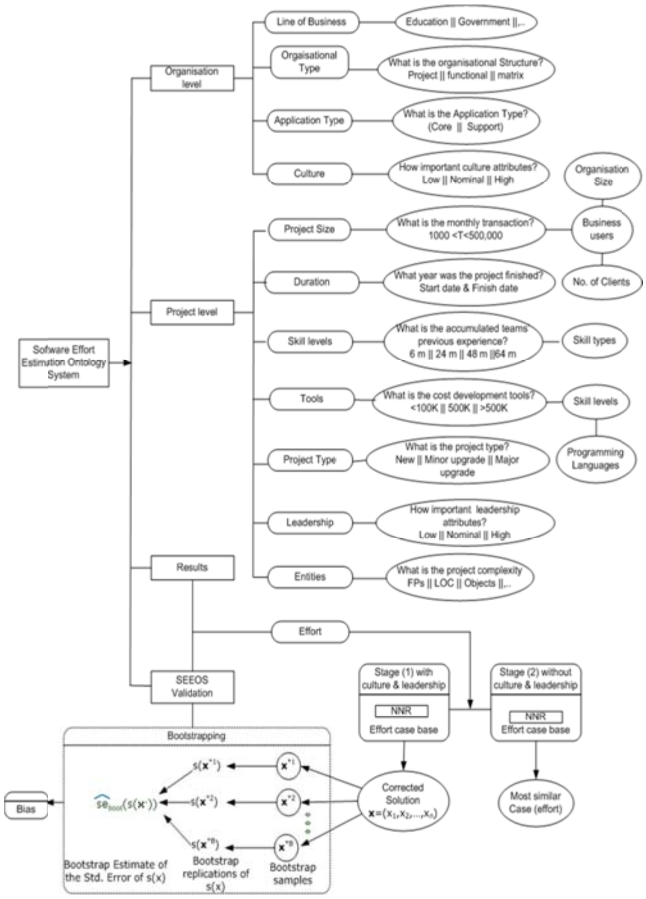


Figure 10. The structure of the query system

The SEEOS is modeled using a three-tier architecture (see Figure 11). This three-tier architecture consists of a presentation layer, a business logic layer, and a database layer. The SEEOS consists of three major subsystems: (1) the analogy subsystem to find the most similar projects; (2) the online subsystem used by different organizations to input projects data; (3) and the bootstrap subsystem to validate the project result.

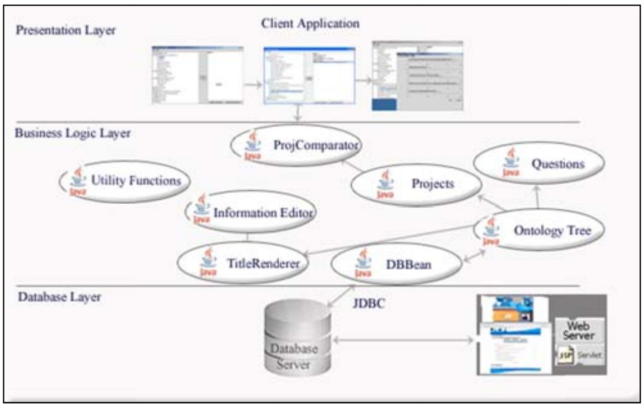


Figure 11. SEEOS architecture

A. SEEOS Presentation Layer

The main SEEOS presentation layer consists of two panels; the right side panel and the left side panel (see Figure 12). The left side panel shows the project's entities along with their attributes, descriptions and values. It consists of a tree structure of different selection keys related to the organization and projects. On the other hand, the right panel shows the selected entities to be estimated by the project manager. It consists of a list which will be populated by the selection keys on the left panel by pressing the "Right Arrow" button located between the two panels. In addition, the right panel has two buttons which are "Compare and Show" and "Remove from the List".

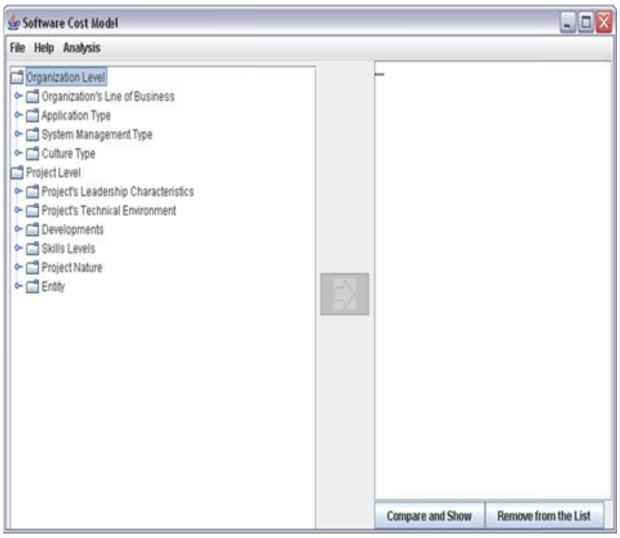


Figure 12. SEEOS user interface

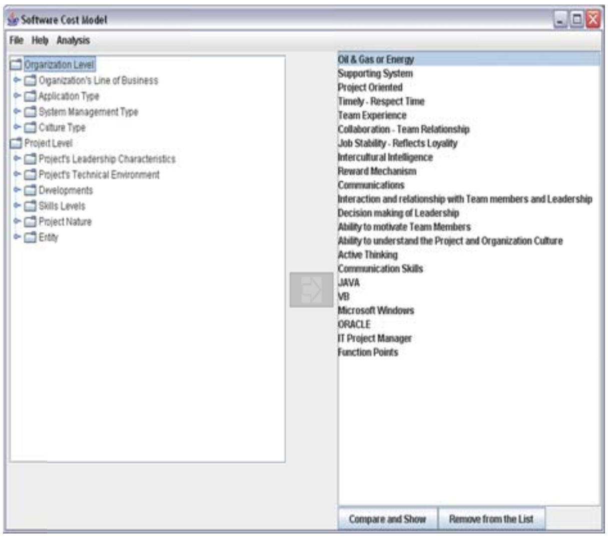


Figure 13. SEEOS keys selected

To find the most similar projects, the project manager can select the project's attributes from a particular leader in the tree structure; he then clicks the "Right Arrow" button to transfer the project attributes to the right panel. The project manager has the option to remove attributes from the list by clicking the "Remove from the List" button in the right panel. Figure 13 shows an example of keys selected from a list.

There are uncertainties in the way various project terms, variables and factors are interpreted. Two projects that may seem similar may indeed be different in a critical way. Moreover, the uncertainty in assessing similarities and differences means that two different analysts could develop significantly different views and effort estimates.

The project manager is able to select similar existing software projects based on well-understood project similarity features. The SEEOS system establishes a set of common project parameters between different projects and provides a common understanding (ontology) of project parameters and their semantics. It accomplishes this by allowing the project manager to give more semantic content to the new project attributes by selecting the attribute's value and measurement. Figure 14 shows an example of keys surveyed to be completed.

After populating the list with different keys related to a project, the project manager can compare these keys with the ones in the database for different projects by clicking "Compare and Show". A dialog box will appear showing the comparison of results between the new project and projects in the database. Figure 15 shows an example of results.

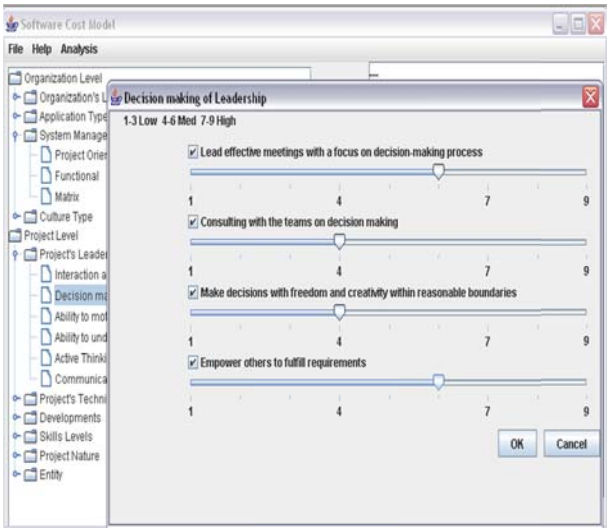
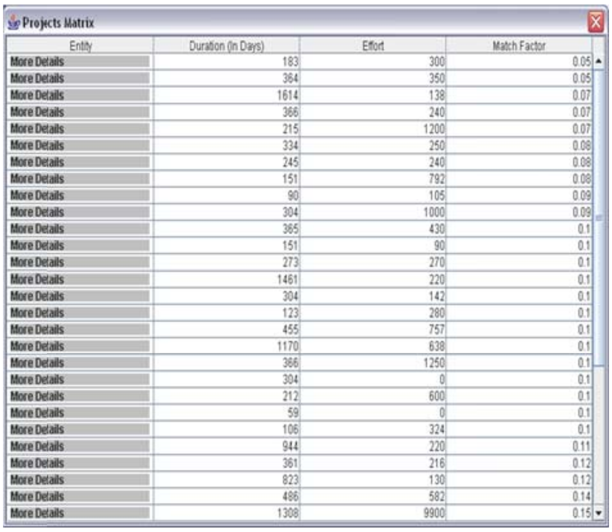


Figure 14. SEEOS Questionnaires



Entity	Duration (In Days)	Effort	Match Factor
More Details	183	300	0.05
More Details	364	350	0.05
More Details	1614	138	0.07
More Details	366	240	0.07
More Details	215	1200	0.07
More Details	334	250	0.08
More Details	245	240	0.08
More Details	151	792	0.08
More Details	90	105	0.09
More Details	304	1000	0.09
More Details	365	430	0.1
More Details	151	90	0.1
More Details	273	270	0.1
More Details	1461	220	0.1
More Details	304	142	0.1
More Details	123	280	0.1
More Details	455	757	0.1
More Details	1170	638	0.1
More Details	366	1250	0.1
More Details	304	0	0.1
More Details	212	600	0.1
More Details	59	0	0.1
More Details	106	324	0.1
More Details	944	220	0.11
More Details	361	216	0.12
More Details	823	130	0.12
More Details	486	582	0.14
More Details	1308	9900	0.15

Figure 15. SEEOS Results

The "Duration" column shows the number of days taken to complete a particular project. The "Effort" column shows the number of man-days involved in completing a project; and the last column, the "Match Factor", shows the calculated similarity value of a project with that created by the user.

B. SEEOS Business Logic Layer

The SEEOS business logic layer consists of seven Java classes. The "Ontology Tree" is the main class for drawing the main window with a tree of different elements and a list box with different command buttons. "DBBean" is a class used for the database management. "InformationEditor" is a class used to add buttons in particular cells. "ProjComparator" is a class used to sort the values of vectors in ascending order. "Projects" is a class used to store all the values against each project in the Database. "Questions" is a class used to store question and answer values. "TitleRenderer" is a class which keeps track of cells of tables. "UtilityFunctions" is also a class consisting of miscellaneous methods providing different functionalities.

C. SEEOS Database System

The SEEOS database system consists of 23 tables (see Figure 16). These tables contain information about different project attributes. The SEEOS online subsystem contains project information from different organizations and inserts them into the database system.

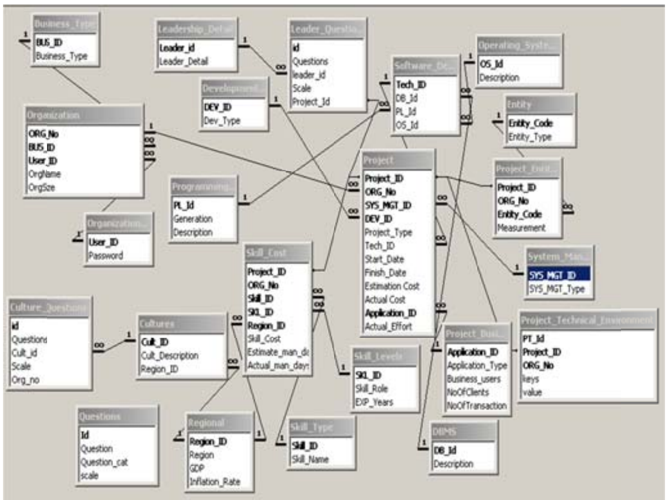


Figure 16. Database entities

Each organization has a user name and password (Figure 17). Each organization has to input project data such as organization name, region, organization type such as public (non-profit), private (for-profit), and semi-government. Also, the organization's line of business such as medical, governmental services, telecommunications, tourism services, public services, education and oil and gas has to be added.

Figure 17. Online system login screen

Figure 18. Online interface

Other essential information to be added includes organization types such as 'project-oriented' where the project manager has the highest power in making decisions, 'matrix' where the project manager has moderate power in making decisions, and 'functional' where the project manager has the lowest level of power in making decisions. Of equal importance is the SEEOS user interface, as well as the information about the developed project such as the project name, application types (core or support), project duration, the estimated and actual project cost, the estimated and actual effort (man-days), entities such as number of transactions, application specific information (source line of code, process, objects or class diagrams, tables and entities, technology used, application architecture, skill sets and accumulated years of experience), the project's leadership, and project team cultural characteristics defined with scale point values (Figure 18).

The SEEOS system has been tested on a cluster sample dataset of 41 cases (projects) collected from more than 20 organizations. The test demonstrated the utility of the system, and its capability for providing a more comprehensive model of all characteristics, including for the first time, leadership and culture.

VI. Conclusion

In a global and diverse environment, culture and leadership play an important role in that impacts work performance, and consequently, software cost estimation. To determine the extent of this role, we analyzed statistically responses to questionnaires and interviews. The results showed a strong influence of these two factors on cost estimations. We also integrated culture and leadership in the CBR model. The inclusion of leadership and culture in the cost estimation model constitutes an enhancement and refinement. We demonstrated that the inclusion of culture and leadership improves accuracy prediction. The Bootstrap method was used to validate our model. The implementation of the model offers an effective tool to help managers maintain historical data on past projects and estimate costs for new projects. This tool is being tested by the local industry.

References

[1] A. Aamodt, and E. Plaza, "Case-based reasoning: Foundational issues, methodological variations, and system approaches", AI Comm. 7(1), pp. 39–59, 1994.
[2] A. Abdulla, and M.A. Al-Homoud, "Exploring the implicit leadership theory in the Arabian Gulf states", Applied Psychology: International Review, 50(4), pp. 506-31, 2001.
[3] W. Aha, "Case-Based Learning Algorithms", Proc. DARPA Case-Based Reasoning Workshop, (1991).
[4] A. J. Albrecht, "Measuring Application Development Productivity", PrIBM Applications Development Symp. Monterey, California, pp. 14-17, (1979).
[5] L. Angelis, and I. Stamelos, "A simulation tool for efficient analogy-based cost estimation", Empirical Software Engineering, 5(1), pp. 35-68, 2000.
[6] L. Angelis, I. Stamelos, , and Morisio, "Building a Software Cost Estimation Model Based on Categorical Data", Proceedings of the 7th IEEE International Software Metrics Symposium, pp. 4-15, 2001.
[7] J. Bellout, H. Harbi, and K. Hamdan, "Bootstrap and other Methods to Measure Learning in UAE Universities and Higher Colleges", METSMac: UAE, pp. 221-230, 2006.
[8] W. B. Boehm, *Software Engineering Economics*. Prentice-Hall, New Jersey, 1981.
[9] S. Conte, H. Dunsmore, and Y. Shen, Software Engineering Metrics and Models. Menlo Park, California, Benjamin Cummings, 1986.
[10] B. Efron, "Jackknife-after-Bootstrap Standard Errors and Influence Functions", J.R. Statistical Society, 54(1), pp. 83-127, 1992.
[11] B. Efron, and R. Tibshirani, *An Introduction to the Bootstrap*, New York, Chapman and Hall, 1993.
[12] W. Fairholm, Values of Leadership: Toward a new philosophy of leadership, Greenwood, NY, pp.4-150, 1991.

[13] D. Futrell, Shafer, and L. Shafer, *Quality Software Project Management*, Software Quality Institute, 2002.
[14] W. Gardner, "On Leadership", The Free Press, New York, Viewed: 1/7/07, http://develop-net.com, 1990.
[15] K. Hamdan, F. Abu Sitta, J. Moses, and P. Smith, "An Investigation into the Gulf States Government Approaches to Software Development and Effort Estimation", BCS-SQM, pp. 111–126, UK, 2005.
[16] R. T. Hughes, "Expert Judgment as an estimating method", Information and Software Technology, 38(2), pp. 67-75, 1996.
[17] M. Jorgensen, and M. Shepperd, A Systematic Review of Software Development Cost Estimation Studies. IEEE 33(1), 33–53, 2007.
[18] B. Kitchenham, S. Pfleeger, and N. Fenton, Towards a Framework for Software Validation. IEEE Trans. on Software Engineering 21(12), pp.929–944, 1995.
[19] B. Kitchenham, T. Hughes, and S. Linkman, "Modeling Software Measurement Data", IEEE Transaction Software Engineering, 27(9), pp. 788-804, 2001.
[20] B. Kitchenham, S. Pfleeger, B. McColl, and S. Eagan, "An Empirical Study of Maintenance and Development Estimation Accuracy", Journal of Systems and Software, 64, pp. 57-77, 2002.
[21] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection", Proceedings of the Fourteenth International on AI 2 (12): San Mateo, pp. 1137–1143, 1995.
[22] M. Krishnan, "The role of team factors in software cost and quality": An empirical analysis, Information Technology & People, MCB University Press, 11(1), pp. 20-35, 1998.
[23] A. Lederer, and J. Prasad, "A Causal Model for Software Cost Estimating Error", IEEE Trans. Software Engineering, 24(2), pp. 137-148, 1998.
[24] T. Mukhopadhyay, S. Vicinanza, and J. Prietula, "Examining the feasibility of a case-based reasoning model for software effort estimation", MIS Quarterly, 6(2), pp.155-171, 1992.
[25] B. Poole, "Values and Culture". Viewed: 1/12/07 <http://www.muw.edu/nursing /tupelo/Culture.html>, 2006
[26] L. Putnam, "A general empirical solution to the macro software sizing and estimating problem", IEEE Trans. on Software Engineering, 4(4), pp. 345-61, 1978.
[27] C. Sarros, K. Cooper, and M. Hartican, "Leadership and Character", Leadership and Organization, Development Journal, Emerald Group Publishing Limited, 27, pp. 682-699, 2006.
[28] C. Schneider, and J. Barsoux, Managing across cultures, Pearson Education, 2nd Edition, England, 2003.
[29] E. Schein, *Organizational Culture and Leadership*, 3rd (ed.), Joddry-Bass, John Wiley & Sons, pp. 129-225, 2004.
[30] M. Shepperd, "Case-Based Reasoning and Software Engineering", Managing Software Engineering Knowledge, in Aurum et al. (ed), Springer-Verlag, 2003.
[31] M. Shepperd, C. Schofield, and B. Kitchenham, "Effort Estimation Using Analogy", 18th International Conference on Software Engineering, IEEE, Berlin, pp. 170-178, 1996.
[32] M. Shepperd, and C. Schofield, "Estimating Software Project Effort Using Analogies", IEEE Transactions on Software Engineering, 23(12), pp. 736-743, 1997.
[33] K. Swigger, F. Alpaslan, R. Brazile, and M. Monticino, "Effects of culture on computer-supported international collaborations", Elsevier Ltd International Journal of Human Computer Studies, 60, pp. 365–380, 2003.

Khaled M. Hamdan, PhD is a lecturer in the IT Unit of the General Requirements Unit at the United Arab Emirates University. Dr. Hamdan has won many awards for his work as a leader and classroom teacher. His research interest includes Software engineering, empirical research, cost modelling and prediction, machine learning (case-based reasoning).

Boumediene Belkhouche is a Professor of Software Engineering in the College of Information Technology at the United Arab Emirates University. His research interests include formal specifications, game-based learning, and modeling of hybrid systems.

Peter Smith is a Professor of Computing at University of Sunderland, UK. His research interests include Intelligent Systems, Applied Artificial Intelligence, Software Engineering and modelling and simulation. He is Fellow of the British Computing Society; Fellow of the Royal Society of Arts, and Chartered Engineer.