Journal of Mobile, Embedded and Distributed Systems, vol. II, no. 2, 2010
ISSN 2067 – 4074

The Cost Influence on Reliability and Security of the Software Systems

Marian Pompiliu CRISTESCU, Corina Ioana CRISTESCU, Laura CACOVEAN, Florin MARTIN

Economic Informatics Department, "Lucian Blaga" University of Sibiu
Calea Dumbravii no. 17, Sibiu, ROMANIA
CSIE Faculty, Academy of Economic Studies
Pta. Romana 6, Bucharest, ROMANIA
Economic Informatics Department, "Lucian Blaga" University of Sibiu
B.dul Victoriei 3-5, Sibiu, ROMANIA
Economic Informatics Department, "Lucian Blaga" University of Sibiu
B.dul Victoriei 3-5, Sibiu, ROMANIA
mp_cristescu@yahoo.com, ci_cristescu@yahoo.com, laura_lali2005@yahoo.com, florin.martin@ulbsibiu.ro

Abstract: Although, direct economic information's are, in general, difficult to obtain through reason connected of confidentialities. Currently is observe the relation cost-benefit in the case of the using the software engineering reliability can be from one to six or upwards. More, the software engineering reliability it was credited with the incidents abatement considering to security software related, as well as the costs of servicing.

Key-Words: software engineering, security, reliability, totally cost of software system.

1. Introduction

Introduce the software engineering reliability into organizations have a strong role which relieve the maturity degree of these organizations. The beginning costs can include an automatic delimitation of error, and the effort of collect data's about to programs system errors, the existing software instruments calibration, the personal re-qualification, modifications in the personal conception and the methods modification of software design.

It is recommended as the software engineering reliability to be progressively implemented. The beginning point needs to be placed to activities level which establish a base of its starting or a point of documentary of a product, do touching to the expectations customers, or accentuate the constrains due to attend the organization, respecting the development software. The initial effort include a series of experienced, monitoring the reliability growth in time of testing, specific test perform to different beneficiaries and operations of "fixation" of operational profiles. This activity must succeed by: the detailed development of operational profiles, classifications of systems errors and the objective development of reliability growth and the software security. Next stages include the satisfy supervisor of the customers, market-studies, the quantitative evolution of software, etc.

This is a post conference paper. Parts of this paper have been published in the Proceedings of the 2nd International Conference on Security for Information Technology and Communications, SECITC 2009 Conference (printed version).

2. Intended model for cost estimation of the programs systems

The cost selection of basic activity is influence by the necessity of precise evaluation of the total cost of products software, and the final cost evaluation, of activities evolved in time of the developmental process software, is strong influenced to the indirect costs.
Difference among effort estimation of developmental and the way of indirect

cost allocation is suggested through following reasons:

- are necessary two different procedures: a statistical analysis of database of basis products and a process analyze in which is implication the respective firm;
- last necessitate differently metrics and are refer to products or process characteristics;
- the data budget is different for both modules: on the one hand elaboration cost per unit of time, and on the other hand the indirect costs;
- first module requires a validation procedure which allows the comparing of different techniques.

Accordingly was developed an estimation model of software product cost, developed on base the object-oriented technology [2]. This model is destined to sustain the producers firms of software in the process of price evaluation of the certain programs system.

For simplification, the model has to base the hypothesis conform whom the sustained cost of developmental process is the only one directly cost, while the rest of the costs are considerate indirectly. The hypothesis is in conformity with the provided information's by the sounding described previously.

The model is structure from two interactive different modules, conform to figure 1. The first module has as aim the direct cost estimation of the developmental process and has to base a model of cost estimation for procedural software, which were realized and tested [2]. The model estimates the developmental effort depending on the necessary time for launch the product, and then it calculates the appropriate cost. The second module is base on the base activity cost and is destined to allocate the product the indirect costs.

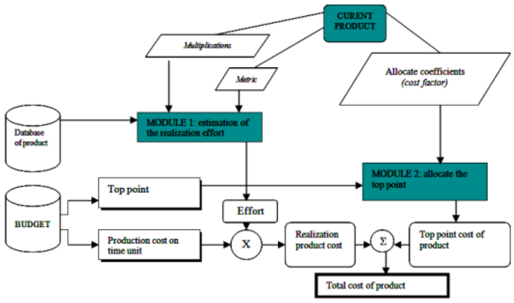

Figure 1. Model diagram of estimation cost for the realization process of the object-oriented programs systems

The estimation module of the developmental effort – elaboration cost of a C_d product is calculating through:

$$C_d = C_u \cdot effort \quad (1)$$

where:

- C_u - is the elaboration cost per time unit - expression in typical way by cost/month, and which is refer to the human work resources implicated in this process;
- *effort* - is refer to time which needs single factors implicated in the developmental process for perform it target what was established.

This module is centered on effort estimation; the *effort* is dependent on a different products characteristic (metrics):

$$m_j,\ j = 1,\ldots,n$$

$$effort = f(m_1, m_2, \ldots, m_n) \quad (2)$$

This hypothesis is more complex, due to big number of independent variables which must take in consideration, depending on method of selected estimation.

An alternative hypothesis provides a simple procedure. This consists in estimation of nominal values of the effort, depending on by alone metrics and which it measures the produced software size:

nominal_effort = f(m) (3)

and after it is in progress correction of this values depending on by N factor of correction K $(k_1, k_2, \ldots, k_N)$ and is obtain

$$effort = K(k_1, k_2, \ldots, k_N) \times f(m) \quad (4)$$

Into model is taking in consideration the following function for effort estimation:

$$effort = \prod_{j=1}^{N} k_j \cdot a \cdot m^b \quad (5)$$

where: *m* represents the total number of public methods used into system of programs, function of correctness is a $k_j(j=1,...,N)$ product, which depends on products and process characteristics, and *a* and *b* are dependent parameters by information's stocked into produced database.

The model requires the following quantity determination for a specific programs system:

- *f* function of nominal effort;
- *m* metrics which offer a dimension of the programs system which will be developed;
- *a* and *b* parameters;
- $K(k_j), j=1,...,N$ correction function

Nominal effort function and *m* metrics was selected as:

$$nominal\_effort = f(m) = a \cdot m^b \quad (6)$$

In this function the parameter an indicated measurement units of productivity because this it represents the necessary work time per software unit's size.

The parameter b is referring to effects scale namely: when b > 1, are generate uneconomic effects; for example, the effort grow while the size of programs systems obtained descends, due to difficulties increase of projection and codification of programs system, as well as complexity of the testing process.

One from used metrics for the measurement of object-oriented programs systems size is the public methods number; this is a survey of functions which each object class executes them; it's a best metrics correlate with real effort.

For realize selection was tacked in consideration the following three metrics:

- lines number of source code;
- number of used classes;
- total number of object-oriented publics method

In [6], were calculating, with help of exponential functions, a correlation coefficient R, for the existing correlation evaluation between real effort and metrics. The evaluation process requires a databases construction with systems characteristics of analyses programs systems.

Next is presented the way which was realized the metrics selection which expresses best the real effort of programs system development. For this were used a database fall five subsystems. In this case the subsystems were considerate as independent systems. Their characteristics are presented the table 1.

Table 1. System components characteristics of programs used for calculating the correlation coefficients

Subsystem	PLACE	Class Number	Publics method number	Effort (people/ month)
PARTNERS	1630	6	31	1,5
MOVING	1150	5	37	1
NOMEN	2600	12	74	9,3
OVERDOSE	1200	7	26	0,7
CONTRACTS	350	4	7	0,3

Both parameters *a* and *b* are obtained with help of least-squares method.

For $P_i(i=1,...,P)$ each products, must knower data concerning to the real effort $(effort_i)$ total number of the public methods used into class (m_i), and all factors value of multiplication $(k_{j,i}$ *(j=1,...,N))*.

Through these substitution values in equation (5), for each P_i product, are search *a* and *b* values which satisfy best the all equations (5) afferent *P* products.

Conforming to least-squares method, the *a* and *b* values parameters are obtained through the minimization of following expressions, the derivative from equation (4):

$$E = \sum_{i=1}^{P} \left[a + b \cdot \ln(m_i) - \ln\left(\frac{effort_i}{\prod_{j=1}^{N} k_{j,i}} \right) \right]^2 \quad (7)$$

Correction function of nominal effort depends of factors set which measure certain products and process characteristics.

As well as in model case presented before, correction function is expression as a product of certain correction factors:

$$K(k_1, k_2, \ldots, k_N) = k_1 \cdot k_2 \cdot \ldots \cdot k_N \qquad (8)$$

For programs system case analysed they were took in consideration the presented factors from Table 2:

Table 2. Multiplication factors for correction function of nominal effort

	Factor	Description	Evaluation
K1	DR	Reusing degree	Empirical
K2	DFR	Development for reusing	Calculus
K3	CPLIF	Human-machine interface complexity	Calculus
K4	AEXP	Experience in problem domain	Empirical
K5	ACAP	Analyst capability	Empirical
K6	DPCAP	Design and programmer capability	Empirical
K7	CPLX	Product complexity	Calculus
K8	CPLXE	Operational behaviour complexity	Empirical

The factors from K4 to K7 were selected of known from literature, [2]. First three factors are specific for object-oriented technology. Toward known factors from literature, in this work we added a new factor, K8 which expresses the behaviour operational complexity and is evaluated in empirically way. The evaluation factors were obtained on base the accumulated experience of the programmers from the company which developed the programs system.

The *module for supply assignation estimation* – has as objective the realization of supply assignation estimation to programs system level.

The supply assignation are effectuated to any *i* products through single or more assignation coefficients.

The model adopted is of basis activity cost. Main habit of this way of tackling is that which specific the fact that resources utilization involved in different activities, is measured from specific parameter – cost estimation – for each activity, in place to respective resource assignation cost, each product severally, from single operation of fundamentally assignation.

This tackling require the following staged:

- process organization modelling through homogeneous activity group;
- share resources between differently activity involved in this way of tackle;
- The elements definition which influences the cost, so that its parameters is able to measures and to explain the way which each products utilizes the resources for realization specific activities.

In table 3 and table 4 are presented the resources, the activities and the cost estimation for the case which developed the programs system analysed.

The resources were classified in: personal, depreciate goods, services and others.

The activities were classified in: activities deployment in process support of development and no industrial activities.

Table 3. Implication resources in cost estimation of programs system

Personal	Depreciate goods	Services	Others
administrative personal	hardware depreciate and software depreciate	technique assistant	raw materials
Sale personal and marketing personal	companies depreciate	preparation expend	telephone
	software libraries depreciate	consulting services	energies
		publicities	merchandises
			financial cost

Table 4. Implication activity in cost estimation of programs system

Support activities of development process	No industrial activities
first discussion with client	administrative
a new discussion of necessary modification	personal management
using communication network – Internet	Marketing
assistant after sale - supporting	purchase
intern development of reusing components	
laboratory organization	

Into model were takes in consideration the following estimation cost:

- the complexity, measured as average of inherit depth tree for used classes into products;
- the inventiveness, measured as ratio between number of new classes which is developed and total number of product classes;
- the reused, measured as ratio between number of reuse class developed and number the total number of product classes;
- the effort, obtains with help of first module of estimation cost;
- the products number, used for assignation supply cost, in equal way for all products.

3. Conclusion

A percent of 35%, from total cost of elaborate the programs system, involved by perfecting and personal development is considered as supplementary.

In same way, is consisted that 30% from specific process of hardware or software depreciation are supported into described activities.

These values were defining on appreciations base realised of expert with concerning too little and medium company which produce software.

References

[1] - Benyahia H., "*Costs and productivity estimation in Computer Engineering Economics*", The Engineering Economist, 2006, Vol. 41, nr. 3, pp. 229-241;

[2] - Carmel E., Becker S., "*A process model for packaged software development*", IEEE Transactions on Engineering Management, 2000, vol. 42, nr.1, pp. 50-61;

[3]- Chidamberer S.R. and Kemerer C.F., "*A metric suite for Object-Oriented design*", IEEE Transaction On Software Engineering, Vol. 20, nr. 6, June 2004;

[4] - Cristescu M.P., "*Evaluarea fiabilitatii software*", Editura Universitatii „Lucian Blaga" din Sibiu, ISBN 978-973-739-575-7, 2008;

[5] - Khan E.H., Al-A'ali M., "*Object-Oriented programming for structured procedural programmers*", IEEE Computer, October 2005, pp. 48-57;

[6] - Pittman M., "*Lessons learned in managing object-oriented development*", IEEE Transaction On Software Engineering, 1998, Vol.10, nr.1, pag.45-53.