Quantifying the Software Quality Cost Drivers

Eng. Gheorghe NOSCA
Ministry of National Defense

The extension of software application in domains where errors recurrence can have the catastrophic consequences requires the increasing quality level. There is a strong connection between the quality level and the product cost. The relation between the economic growth and the quality cost proves that the latest must be one of the strategic elements for the management. It is intended to obtain a maximum quality level keeping the cost at the lowest level possible. The measure of the quality cost is a major element of a program for the quality improvement. Software has several properties that distinguish it from others products and, in comparison with these, the software development demands a high quality level from start. The quality cost improving leads to increasing software cost level that defines the quality cost. Therefore there is a need of finding the best methods for estimating, measuring, predicting and, finally, for optimizing the quality cost. To measure or to estimate software quality cost it is necessary, first of all, to identify the software quality cost drivers. The second step, very important, is to ascertain the level of influence in the quality cost for each cost drivers. Understanding, identifying, and reducing the cost of quality is the best way to improve the company's profitability.

The Software Quality

The software quality is a complex notion derived from an aggregation of some quality characteristics with different measure units. Another trouble is that same of the software quality characteristics are in conflict: *added efficiency is often purchased at the price of portability, and maintainability; added accuracy often conflicts with portability via dependence on word size; conciseness cans conflicts with legibility* [BOEH00].

The quality characteristics' importance level is, also, different according to the field where the software will be used. In the nuclear field, for example, it is imposed the maximum level for the reliability. Evidently, the increasing of quality level implies, automatically, the increasing of the costs. The practice has demonstrated that increasing the quality level above a certain limit means an exaggerated increasing of the cost.

The Software Quality Cost

The cost of quality, as part of the cost of elaborating a software product is the money expression of the consumptions of resources determined by the attainment of certain level of the quality characteristics or by their improvement. [BARO95]

The software products have some particularities as:

- The lack of the physical wearing
- The lack of the possibility to establish in design phase limits of tolerance. The expression *approximately correct* can not be used in the software field.

Studies in software field show that up to 60% of software developers are involved in fixing errors at any point in time, and 47 working days of a full calendar year are available for actually developing or enhancing software applications The study of software quality obviates the need for ascertaining the cost associated with attaining the quality as well.

Quality cost consists of cost incurred due to lack of quality and cost incurred for achieving a quality level. Costs due to a lack of quality are further divided into costs of internal failures and costs of external failures. Costs of achieving quality are further divided into appraisal cost and defect-prevention costs. The costs of achieving quality (costs of conformance) and the costs due to lack of quality (costs of nonconformance) have an inverse relationship to one another. Quantifying software quality is important because facilitates decision making.

Software quality cost drivers

A cost driver is any factor that affects cost. That is a change in the cost drivers will cause a change in the total cost of a related cost object. Cost reduction effort must focus on efficiently managing the use of the cost drivers in the value-added activities.

A quality cost drivers is a particular characteristic of the software development that has the effect of increasing or decreasing the amount of development effort, e.g. required product reliability, execution time constraints, project team application experience. There are many candidate factors to consider in developing a better model for estimating the cost of a software quality. Having in view the large number of quality cost drivers, it is almost not possible to build a model taking into consideration all cost drivers. Taking into account causal analysis or Pareto Analysis (80% of the quality costs are incurred by 20% of the quality failures), will be used to build a model only the quality cost drivers which have the relevant influence in quality cost. The software quality is influenced by a set of factors. For increasing the quality level with optimum resources consumption it is necessary to identify and to analyze these factors.

According to [BARO82] the software quality cost drivers can be grouped in the following categories: technical, economic, social, structural and psychological.

Generally it is difficult to estimate the quality cost drivers. It can be possible to know the exact measure of some qualitative activities. That does not mean the transformation of the quality cost drivers into quantitative value. Because quality is really a composite of many characteristics, the notion of quality is usually captured in a model that depicts the composite characteristics and their relationships. The studies in software development processes show that it is possible to identify almost all the quality cost drivers, but only some of them are relevant. The relevant quality cost drivers are accordingly the software nature. *There are quality cost drivers influencing product, process, project and personnel.*

The product quality cost is generally influenced by the quality characteristics, the most important being reliability, usability, efficiency, maintainability, portability, reusability, complexity.

The process quality cost is influenced by the level of customers' participation and management. The requirement consistency, the architecture, the data processing, the constraints regarding schedule and budget are the project quality cost drivers.

Among the personnel quality cost drivers the most important are the analysts' capacity, the experience, and the cohesion of the team.

There are two principles to reduce the large number of cost drivers to a relatively manageable number of factors for practical cost estimation:

- general significance: this tends to eliminate factors which are significant only in a relatively small fraction of specialized situations.
- independence: this tends to eliminate factors which are strongly correlated with product size, and to compress a number of factors which tend to be highly correlated on projects into a single factor.

For example the COCOMO model to estimate software cost uses the following cost drivers: software reliability, size of application database, complexity, analyst capability, software engineering capability, applications experience, virtual machine experience, programming language expertise, performance requirements, memory constraints, volatility of virtual machine, environment, turnaround time, use of software tools, application of software engineering methods, required development schedule. These cost drivers can be considered the most important for the software quality cost level.

These cost drivers are grouped into four categories: software product attributes, computer attributes, personnel attributes, and project attributes. All these cost drivers are rated from very low to extra high

Product Attributes

- Reliability: the extent to which the software product must perform its intended functions satisfactory over a period of time.
- Data Base Size: the degree of the total amount of data to be assembled for the data base.

- Complexity: the level of complexity of the product to be developed.

Computer Attributes

- Execution Time Constraint: the degree of the execution constraint imposed upon a software product.
- Main Storage Constraint: the degree of main storage constraint imposed upon a software product.
- Virtual Machine Volatility: the level of the virtual machine underlying the product to be developed.
- Computer Turnaround Time: the level of computer response time experienced by the project team developing the product.

Personnel Attributes

- Analyst Capability: the level of capability of the analysts working on a software product.
- Applications Experience: the level of applications experience of the project team developing the software product.
- Programmer Capability: the level of capability of the programmers working on the software product.
- Virtual Machine Experience: the level of virtual machine experience of the project team developing the product.
- Programming Language Experience: the level of programming language experience of the project team developing the product.

Project Attributes

- Use of Modern Programming Practices: the degree to which modern programming practices (MPPs) are used in developing software product.
- Use of Software Tools: the degree to which software tools are used in developing the software product.
- Schedule Constraint: the level of schedule constraint imposed upon the project team developing the software product phase as well as development phase.

When the software products consist of a combination of newly developed software and previously developed software, it is necessary to adapt the mentioned cost drivers set for the use in the new product.

To estimate software quality cost it is necessary to identify both the quality cost drivers and the relationships among them. There are two kinds of relationships among the software quality cost drivers: direct relationships and interaction relationships.

The direct relationships means the quality cost driver has a direct contribution to quality cost variation, without any connection with other quality cost drivers evolution. The interaction relationship supposes that each quality cost driver has its influence in quality cost variation and the total contribution is the sum of individual influences.

Methods to quantify the quality cost drivers

The quality cost drivers are cost drivers for which it is difficult to assign a precise value. It can be given only a value through heuristic methods. Although it is possible to assign exact values to some activities, these numbers come from a qualitative point of view. To assign an exact value does not mean that qualitative cost driver converts into a quantitative one. As well as defining the time spent on indirect tasks the drivers take into account resource and environmental factors.

An essential condition to understand and to develop a comprehensive analysis of economical phenomena is the measurement of its essential parameters. Such kind of activity is very difficult because imply the appreciation both quantitative and qualitative factors. To solve this problem it is necessary to establish the measurement method. For measuring data referring to the economical processes can be used the following scales: non-numerical scale; numerical scale; proportional.

Elaborating a scale it is necessary to have in view:

- the scale must to be understood by all the personnel involved
- the scale must to establish intensity levels for the characteristics analyzed

Before to present the method of association a numbers to qualitative factors, it is necessary to define and classify non-numerical factors.

An economical process influence factor is:

- essentially non-numeric factor if it is not known yet a method to measure one of its characteristics;
- conjuncture non-numeric factor if it is known a method to measure at least one of its charac-

teristics, but due to some circumstances it is not possible to do it.

Quantification methods:

- *the establishment of a ordinal relationship among the results of some experiments*

In this case it is necessary to realize observation concerning some objects O_1, O_2, ..., O_n.. The object can be activities, elements etc. After that, having in view some criteria, an ordinal relationship, is established.

$$O_i \succ O_j \qquad i, j = 1,2,...........,n.$$

Among the elements belonging to the same group there are indifferent relationship.

- *giving rate, marks, score etc.*

According to this method the specialist k gives for each object i a mark N_{ik}. If the specialists give the rates, it is easy to transform them in marks. Example:

Rates	Marks
Very low	1
Low	2
Medium	3
High	4
Very high	5

- *Neuman-Morgenstern utilities*

The method is used in case it is necessary to quantify both numeric and non - numeric factors. Nonnumeric factors are firstly evaluated by marks $N_{ij.}$ the utility is given by the follow equation:

$$U_{ij} = \left(N_{ij} - \underset{i}{Min} N_{ij}\right) \Big/ \left(\underset{i}{Max} N_{ij} - \underset{i}{Min} N_{ij}\right)$$

This method advantage is that all the values belong to (0, 1). In this way it is possible to compare them. The utilities can be also used to estimate a set of heterogeneous characteristics

- *using fuzzy set theory*

This method associates, according to an given criterion, to each characteristic O_i, a degree of appartenence: $I(O_{ij}) = m_{ij}$. For using the measurement theory basic criteria in software field it is necessary to identify and to define the following:

- software products and processes attributes
- models including attributes
- the relation between model and the characteristic which is to be modeled.

It is very difficult, even for the companies which have obtained the third level according to CMM, to take into account all the software quality drivers cost.

In order to establish the level of contribution in quality cost for each quality cost drivers can be used experts judgement. It is necessary, also, to collect data from many software projects.

Conclusions

Understanding, identifying, and reducing cost of quality is the best way to improve the company's profitability. Managing through cost of quality show how to start or improve a cost of quality system, and how to combine cost of quality with corrective action tracking for effective management. The major benefit of cost of quality analysis is to ascertain the return on investment on the software process improvement measures. Quality costs analysis offer, also, the possibility to identify the most effective cost drivers and to quantify their influence in order to manage them, and to maintain the optimum between the quality and the cost levels.

References

[ALBU00] Albuquerque, J., Coelho Jr., C.: *An Approach to Multi-Team Design and Cost Estimation,* www.dcc.ufmg.br/pos/htm/spg2000/anais/joa/cicc00.html

[APRE95] Apreutesei P., Nosca Gh.: *Optimizing the Software Quality Cost,* International Conference Software Quality, University of Maribor, Slovenia, November 1995.

[BARO88] Baron, T., Ivan, I., Goga, A.:*Modelling the Cost of the Quality of Program Systems* ;Economic Computation and Economic Cybernetics Studies and Research, r.2/1988, Academy of Economics Studies, Bucharest-Romania.

[BOEH00] Boehm,B.,W. *Spiral Development: Experience Principles and Refinements,* http://www.sei.cmu.edu/spiral2000/Boehm

[IVAN95] Ivan, I., Nosca, Gh., Apriotesei, P.: *Optimizarea costului calitatii stemelor de programe,* A XXVI-a Sesiune de comunicari stiintifice, Academia Tehnica Militara, Bucuresti 1995