Cost of Software Quality

Sheikh Waqas Jameel[1], Usman Waheed[2]
SZABIST
Karachi, Pakistan

Abstract: *It is often difficult to plan that how much cost would incur in quality management of the software project. Cost of Quality is always uncertain. The main problem is that we usually take quality as an overhead. Estimations are normally calculated for the development of project. In this research we focus on Cost of Software Quality, and how to handle appraisal efforts performed for achieving acceptable quality, and prevention efforts to prevent poor quality.*

Keywords: Prevention, Appraisal, Internal failures, and External failures, extended cost of quality, total cost of quality

1. INTRODUCTION

Quality Management is taken as the Cost center in many software houses. It is often asked that, 1) How much working on the quality processes cost? And,2) what is the return of investment?

The manager recognizes that to implement a quality improvement project, rich resources must be allocated toward processes not currently being undertaken. Prior management experience has proven that usually the resources available are barely adequate to meet aggressive project and schedule deliverables.

We need to define the quality of the product and identify the Cost items that are related to the development of the product. Distribute the cost according to the cost heads. Since conformance and non conformance costs have inverse relation, investing on the conformance of the quality will reduce the cost of the non conformance.

2. RESEARCH METHODOLOGY

The methodology selected for is research is based on the survey to identify the cost of quality heads implemented in industry.

3. QUALITY

Quality could be taken as the degree to which users perceive that a software product meets their composite needs and expectations.

The business meanings of quality have developed over time. Various interpretations are given below:

- ISO 9000 [1]: "Degree to which a set of inherent characteristics fulfills requirements." The standard defines requirement as need or expectation.
- Philip B. Crosby [2]: "Conformance to requirements." The requirements may not fully represent customer expectations.

3.1 Myths

There are many myths regarding Quality, but the most recognized five myths of Software Quality with their realities are mentioned below [9].

Myth: Quality is a discipline learned on the job, not in a classroom
Reality: bring the quality experts into the classrooms and teach quality in today's terms.

Myth: Cost, quality and schedule form an iron triangle
Reality: quality of a process is evaluated and improved; it will run faster, produce fewer defects and cost less.

Myth: No preparation is required to run an improvement program except willpower
Reality: Lack of good preparation often results in fixing mere symptoms or addressing the wrong issues.

Myth: Six Sigma and lean are great new tools
Reality: You can't train one person in a set of tools and expect to effectively implant business process improvements across an organization.

Myth: Choosing a quality approach is a task for senior leaders
Reality: the processes of upward communication are needed to keep information flowing,

4. QUALITY ACTIVITIES

The association between the quality of a product and the Quality policy of the organization responsible for the creation of that product is multidimensional. It also depends upon the combination of activities selected by the organization to achieve the desired product quality. These activities include assessment of that product, Test management, planning, development, automation and execution, process Improvement, risk management, release criteria, defect reporting, change management estimation and scheduling, peer reviews, inspections and verification & validation. It is discovered not all the quality activities are the cost of quality activities [8].

5. COST OF QUALITY (COQ)

Cost of Quality is a concept which provides the means to quantify the cost related to the efforts and deficiencies that are taken place during the development of any product. Basically the "cost of quality" isn't the price of creating a quality product or service. It's the cost of NOT creating a quality product or service [7]. Improving quality can do more than reduce costs.

The Cost of Quality can be widely distributed in two categories:
Costs of control (costs of conformance)
Costs of failure of control (costs of non-conformance)
Costs of control (costs of conformance) contain prevention costs and appraisal costs. A cost of failure of control (costs of non-conformance) contains internal failure costs and external failure costs [10]. Following are the Cost heads mentioned in table 1.

COSTS OF CONTROL (COSTS OF CONFORMANCE)		COSTS OF FAILURE OF CONTROL (COSTS OF NON-CONFORMANCE)	
Prevention Cost	Appraisal Cost	Internal Failure Costs	External Failure Costs
Quality planning	Test & inspection of purchased materials	Scrap	Complaints in warranty
Statistical process control	Acceptance testing	Rework	Complaints out of warranty
Investment in quality-related information systems	Inspection	Material procurement costs	Product service
Quality training and workforce development	Testing		Product liability
Product-design verification	Test and inspection equipment		Product recall
Systems development & management	Quality audits		Loss of reputation

Table 1. Types of Cost of Quality

5.1 Prevention Costs
Prevention costs include investments in establishing a software quality infrastructure, updating and improving that infrastructure, as well as performing the regular activities required for its operation. A significant share of the activities performed by the SQA team is preventive in character, as reflected in the SQA budget. Typical preventive costs include:
1. Investments in development of new or improved SQA infrastructure components
2. Regular implementation of SQA preventive activities:
3. Control of the SQA system through performance of internal quality reviews, External quality audits by customers and SQA system certification organizations

5.2 Appraisal Costs
Appraisal costs are devoted to detection of software errors in specific projects or software systems. Typical appraisal costs cover:
1. Reviews:
2. Costs of software testing:
3. Costs of project progress reporting.
4. Costs of assuring quality of external participants

5.3 Internal Failure Costs

Internal failure costs are those incurred when correcting errors that have been detected by design reviews, Software tests and acceptance tests performed before the software has been installed at customer sites.

In other words, internal failure costs represent the costs of error correction subsequent to formal examination of the software during its development, prior to the system's installation at the customer's site.

5.4 External Failure Costs
External failure costs entail costs of correcting failures detected by customers or maintenance teams after the software system has been installed at customer sites. These costs may be further classified into overt and hidden external failure costs. In most cases, the extent of hidden costs is much greater than that of explicit costs.

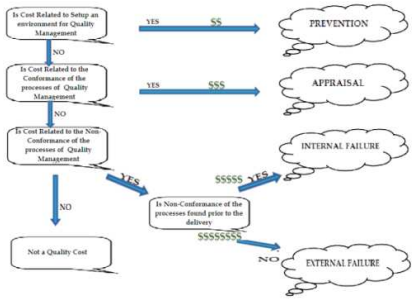

Figure 1. Cost related to the conformance and non Nonconformance

According to figure 1 if the cost is related to the setup and infrastructure of the Quality management then this could be termed as the prevention cost. If the cost is related to the conformance of the processes of Quality management then the cost could be stated as Appraisal.
If the cost is related to non conformance and is found prior to the delivery then it is the cost of the internal failure. And if after delivery of the product, the non conformance found is external cost of quality.
Figure 1 concludes that early investment in quality at prevention level will lead to less investment, whereas at appraisal level the investment increases. If not justified investment is assured then it will lead to higher cost at non conformance level.

6. COST OF SOFTWARE QUALITY AND CMM

By the time when processes starts to get matured and things get streamlined the Total cost of quality start to decrease and with that, other Cost of non conformance (internal cost of error and external cost of error) starts to come down.
Due to the SEI CMM process maturity movement, we have an aggregation of the payoff data that has been collected as a result. The following model makes predictions about Cost of Software Quality across the levels of the Software Engineering Institute's Capability Maturity Model [3].

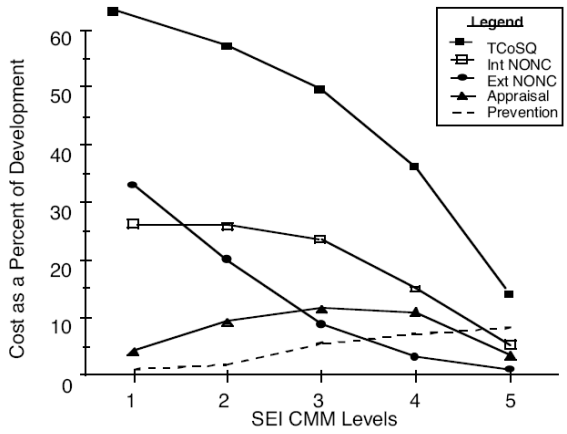

Figure 2. Relation between cost of quality and CMM level

The figure 2 concludes that as we move to higher maturity level all types of cost of quality parameters reduces except appraisal cost initially reduces and remains higher for 3rd level then reduces gradually till last level. It also concludes that gradually investing at conformance cost (appraisal, prevention) will lower the cost of non conformance (internal failure, external failure).

7. COST OF SOFTWARE QUALITY & SOFTWARE ESTIMATION

Unfortunately Cost of Quality is never added or taken as a part in the software estimation.
Here we take a look in to the five software estimation tools that are widely used in the estimation.

7.1 Work breakdown structure

WBS is a way of organizing project elements in to a hierarchy. A software WBS consists of two hierarchies one of which represents software product itself and the other representing activities needed to build that product [6].

7.2 Algorithmic Cost Modeling

An algorithmic cost estimates in its most general form can be expressed as equation 1
Effort = A * SB * M eq - 1

Where S is the code size, A and B are functions of other cost factors, and M is the multiplier made by combining process, product and other development attributes. Surprisingly none of the Cost of Quality heads are mentioned in cost factor in standard equation as well as equations derived from it.

7.3 Function Point

A function point is a unit of measurement to express the amount of business functionality an information system provides to a user [4]. The basic equation of function point is stated as equation 2
FP = count total x [0.65 + 0.01 x $\sum$(fi)] eq - 2
But unfortunately none of the 14 points mentioned in $\sum$(fi) is reflecting cost head defined by Cost of quality.

7.4 COCOMO

The constructive cost model was first introduced in 1981 by Barry Boehm. The model assumes that size of a software project is calculated in KLOC. COCOMO exists in three different stages i.e. Basic, Intermediate and Advanced. The formula that COCOMO uses is expressed as equation (3) and (4)
Large Projects Effort = a * Size (KLOC) b eq - 3
Small projects Effort = a * Size (KLOC) * b eq - 4
The parameters a & b are defined on the basis of characteristics of the project. COCOMO executes on three different classes of projects which are Organic, Embedded and semi detached.
But none of the parameter or cost drivers are reflecting cost head defined by Cost of quality.

7.5 COCOMO II

COCOMO II [5] is detailed and major revision of COCOMO 81 which is evolving to deal with the shortcomings of COCOMO 81. COCOMO II focuses on issues such as non-sequential and rapid-development process models; reuse driven approaches involving commercial-off-the-shelf (COTS) packages, reengineering, applications composition, and software process maturity effects and process driven quality estimation.
The COCOMO II effort estimation model is introduced in equation 5. This model is used to estimate both early design and Cost architecture models.
Effort (Person-Month) = A*(Size) B Πn EM I eq - 5 i=1

Where A is a constant derived from historical project data (A = 2.94 for COCOMO II). Size is in KLOC or converted from function point or object points, B is an exponent for the diseconomy of scale dependent on five additive scale

drivers and EM i is the Effort Multiplier for the *i*th cost driver). But unfortunately none of the scale drivers and cost drivers are reflecting any cost head defined by Cost of quality.
But unfortunately none of the scale drivers and cost drivers are reflecting any cost head defined by Cost of quality.

8. SUGGESTION

It is suggested that by adding two more parameter in the effort estimation equation that are prevention "p" and Appraisal "a" we can include the cost of Quality in the estimations of the development of the product as mentioned in equation 6.

Effort = A * SB * M (p+ a) eq - 6

9. CONCLUSION

In the end it is concluded that, gradual investing at conformance cost (appraisal, prevention) will lower the cost of non conformance (internal failure, external failure). As we move to higher maturity level of CMM all types of cost of quality parameters reduces except prevention costs.
It is also observed the cost heads are not calculated at the time of the development estimation. None of the major estimation techniques include any cost head. We suggest that the Prevention and appraisal cost should be added in the basic estimation equation. The new equation should be like equation (6).

REFERENCES

[1] TC 176/SC, ISO 9000:2005, "Quality management systems -- Fundamentals and vocabulary", International Organization for Standardization, 2005

[2] Crosby, Philip, "Quality is Free", New York: McGraw-Hill. ISBN 0070145121, 1979

[3] Knox, S.T, "Modeling the Cost of Software Quality", Digital Technical Journal, Vol.5, No:4, pp 9-16, 1993

[4] Wikipedia - Function points, http://en.wikipedia.org/wiki/Function_points (Last visited 13 Nov 2009)

[5] COCOMO II Model Definition Manual V2.1 Center for software Engineering USC

[6] B Boehm, "Software Engineering Economics", Prentice Hall, 1981.Sd

[7] G. Gordon Schulmeyer, "Handbook of Software Quality Assurance", Fourth Edition, 2006

[8] Pressman, R., "Software Engineering – A Practitioners Approach", 5th edition, McGraw-Hill, 2005

[9] Douglas C. Wood, "Blurred Vision: Do Our Business Leaders See What Quality is All About?", DC Wood Consulting, July 2008

[10] SA Slaughter, DE Harter, MS Krishnan, "Evaluating the cost of software quality ", August 1998/Vol. 41, No. 8, COMMUNICATIONS OF THE ACM