

Comprehensive View on Defect Analysis and Change Control Process

D.Kavitha

Abstract - In this research paper, to improve software quality, defects are identified from different projects. The defects are classified and then analyzed. Those defects are then eliminated by finding the root causes, for which defect preventive mechanisms are established for reducing reoccurrences of similar defects in the subsequent projects, thus improving quality. Effective change management methodology can keep quality under control.

Keywords— Defect management, Software quality, Defect prevention,

I. INTRODUCTION

This research, describes the approaches for analyzing defect data to understand the quality control process and identify process improvement opportunities. The identified defects are generally recorded in a database, which is used for tracking and prioritizing defects. The scope of this research is to provide a comprehensive view on the principle of defect prevention, defect analysis and change control process. Managing the software changes is an essential activity of software project life cycle.

II. LITERATURE REVIEW

Defect management activities are also supported by various quality standards. CMM at Level 5 considers defect management as a key process area with the following goals: defect prevention activities are planned, common causes of defects are sought and identified, and common causes of defects are prioritized and systematically eliminated [1].

According to the Quality Assurance Institute (QAI), the defect management process consists of six elements: defect prevention, deliverable base lining, defect discovery, defect resolution, process improvement, and management reporting [2].

According to Sakthi, “Defect Prevention also increases the quality of a software product while reducing overall costs, schedule and resources”. Defect Prevention activity enhances the total productivity of the software projects [3].

Software change management is the process of requesting, analyzing, approving, developing, implementing, and reviewing change within the software industry. Manoj state’s that, Change management and dealing with changes is a key project management function for any software industry [4].

III. METHODOLOGY ADOPTED

In this research the following methodology has been adopted.

Steps

1. Identified the error causes and the root causes of the defects raised.
2. Analyzed the Change request data.

IV. DEFECT DATA ANALYSIS

Defect Analysis is using defects as data for continuous quality improvement. In this research, the defect data during the software maintenance for five different projects has been collected. Defects are raised by the customers through the defect tracking system which contains all the information’s regarding the defects. This information is called as Defect Report.

A) Statistical Process Control Chart

The analytical tool from Total Quality Management TQM is the Statistical Process Control. This quality tool is useful for detecting the root causes of a defect and for classifying and prioritizing issues in a well established and ordered manner. Statistical Process Control is used to identify the process in control, and identify problem to make improvement. Control Charts is a graph that monitors process quality. Control Charts process is within Statistical control limits. Control limits upper and lower bands of a Control Charts.

Defect report has been consolidated in a excel sheet. Defect Report from five different projects has been taken for the period of 17 months (Sep 2010 to Jan 2012). Statistical process control chart has been drawn for the average defects for the five projects. Due to some reasons the defects get incorporated in the project after it is delivered to the customer.

Control chart shows variation in a measurement during the period of month that the process is observed. The points are plotted on an x/y axis with the x-axis representing months and y-axis representing the number of defects. Some additional horizontal lines representing the average measurement and control limits are drawn across the chart.

Manuscript received May 25, 2015.

D.Kavitha, Assistant Professor, Department of Computer Science and Applications, Nazareth College of Arts and Science, Chennai, India.
d.kavithamoorthy@gmail.com

Comprehensive View on Defect Analysis and Change Control Process

Defect Report contains the following information's during the Software Maintenance:

Table 1: DEFECT REPORT

Field	Explanation
Call Ref	Specifies the call reference No
Call Log Date	Specifies the call Log Date
Delivery Date	Specifies the Delivery Date
Incident Description	Details description of the defect
Subject	Shot description of the defect
Support Status	Specifies the Support Status (By the developer)
Customer Status	Specifies the Customer Status (By the customer)
Call Type	Bug/Clarification/Implementation/General
Customer Priority	Show Stopper/ High / Medium /Low
Support Priority	Show Stopper/ High / Medium /Low
Project	Project Name
Product	Product Name
Module	Specifies the Name of the Module
Reported By	Specifies the Name of the Company
Closed Date	Specifies the Date closed
Status Updated Date	Specifies the Status Updated Date
Allocated To	Name of the person allocated
URL/Form Name	Status URL/Form Name
Severity Level	Show Stopper/ High / Medium /Low
Reopen Count	Gives the reopen count
Shipped Date	Specified the postponed data

Control chart has helped determine whether special-cause variation is present implying that action needs to be taken to either eliminate that cause if it has a negative effect on the process or to make it standard operating procedure if that cause has a beneficial effect on the process. The main purpose of this study was to identify the problems that organization have in software maintenance. The challenge in any software product development lies in minimizing the number of defects.

C –Chart:

Calculated the specific values of UCL /LCL and mean value of a metric:

1. C= No of defects per sample

$$C = 256/17 = 15.06$$

2. Standard Deviation $\sigma_c = \sqrt{\bar{c}}$
 $\sigma_c = 3.88$

3. Upper control limit $UCL = \bar{c} + z\sigma_c$
 $UCL = 15.06 + 3*3.88 = 26.70$

4. Lower control limit $LCL = \bar{c} - z\sigma_c$
 $LCL = 15.06 - 3*3.88 = 3.42$

No of defects should fall between the lower control limit (LCL) **3.42** and an Upper control limit (UCL) **26.70**, with a mean of **15.06** no of defects.

Distribution of a metric lying outside UCL and LCL:

From the figure1, it is clear that most of the values fall in the range UCL and LCL. But there are 3 points got fixed above UCL and 1 point got fixed below LCL. Even the values are within the acceptable ranges, it is important to see where the value falls outside the UCL/LCL limits and understand the root cause of why such a deviation took place. After analyzing the defect report thru statistical process control chart, we

understand that software defects during the maintenance are out of control due to some reasons. Defect prevention is also an essential part of software process quality improvement. Defect prevention is a process of identifying defects, their root causes, and also preventive measures have been taken to prevent them from recurring in the future, will lead to the better quality software product.

C- Chart:

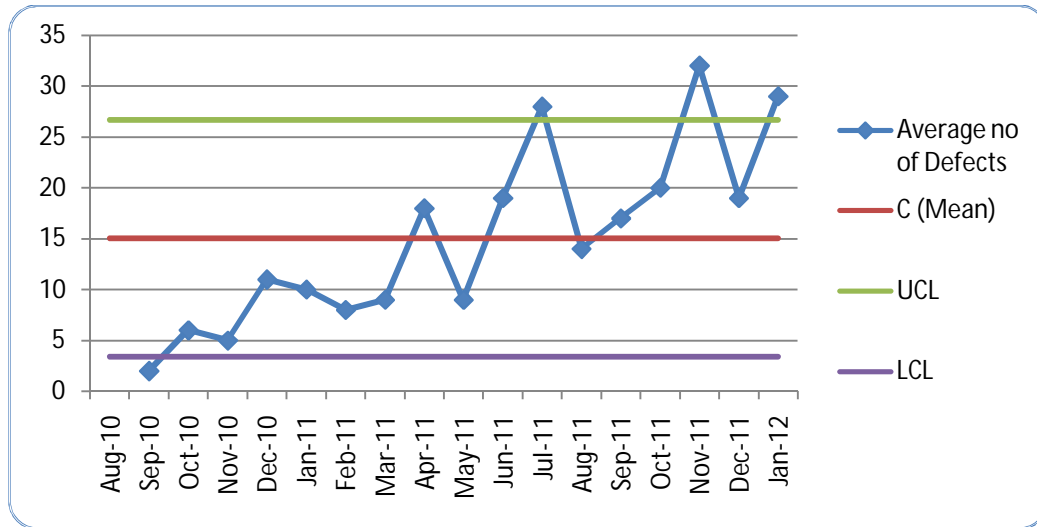


Fig 1: Control chart

B) Error Cause and Root Cause of defects

The purpose of this research is to underline the importance of Defect Prevention activity in software maintenance. Defect prevention activity will collect the defect data during the software development but it is not been taken place during the software maintenance. Defect prevention methodology is applied to the defect report (during the software maintenance) to identify the error cause and the root cause of the defects. Root cause analysis is the process of finding and eliminating the causes of defect, which prevents the problem from recurring. Root cause analysis is a group reasoning process applied to defect information to develop organizational understanding of the cause of a particular class of defects. The analysis should lead to implementing changes in processes that help prevent defects from re-occurring.

During the Causal Analysis, Root cause analysis review will be done. Long term action will be taken, because of process improvement needed. After defects are logged and documented the next step is to analyze

them. Defect sheet should contain correct and complete description of the defect, and then the root cause analysis and preventive measures will be taken care. All proposed preventive measures are analyzed by the Defect Prevention board. The Defect Prevention board then passes on the preventive actions to all other projects. If these actions involve any changes to the Organizations Standard Software Process, they are communicated to the Process Change Management Board through formal "Process Improvement Proposals".

Defect prevention activity does not take place during the software maintenance. Defect data has been collected for five different projects. Defect data has been consolidated and after analyzing the defect originated phase, and the error cause of the defects, the preventive action for the defect has been provided. This will reduce the number of defects occurring henceforth.

It is been identified that defects are getting raised during the software maintenance, due to various reasons. Defects are originated during the different phases like Requirement, Design, Coding, Testing and Maintenance.

During the software maintenance, the error cause are Insufficient / Incomplete information in CR Log, Insufficient / Incomplete information in CR/PR, New Requirement, Requirement document not Available / Updated and Requirement Changes.

The Root causes of the defects, which are originated during the software maintenance, are due to requirement change and insufficient information available in the current change request form and change request log. Hence it is understood that, there is a relation between Defect Report and the Change Request, by identifying the error cause and root cause of the software defects.

V. CHANGE REQUEST ANALYSIS

Change Management Process starts with the creation of a Change Request within the software companies, and results in implementation of the change. The purpose of the change control activity is to identify and track all actions and changes resulting from a change request, from initiation to release. Software configuration control board SCCB approves the changes entered in the new software baseline.

In this research, the Change Request data (during the software maintenance) for five different projects has been collected, and calculated the number of defects raised due to each Change Request by finding the root cause of the defects.

There are five Projects taken for analysis. Maximum of 36 Change Request are there in each Projects. Maximum number of defects raised due to each Change Request is 6. On an average of 1.83 defects are raised due to change request.

Change request form (Table 2) plays a vital role in the software maintenance activities. Change request are raised by the customers through the change request form which contains all the information regarding the change. From the defect data analysis, it is been identified that defects are getting raised during the software maintenance, due to Insufficient / Incomplete information in Change Request Log, Insufficient or Incomplete information in Change Request form, New Requirement and Requirement Changes. Root cause has been identified for the defect report, and it is been understood that the current change request form is insufficient for change request process.

VI. CONCLUSION

This research, describes the approaches for analyzing defect data to understand the quality control process and identify process improvement opportunities. The main goal of the Software Change Management is to carry out the Software Change Request in a very efficient manner, by minimizing the impact, costs, and the risks. Requirement changes are recorded by the change management using the software change request form. Change request are raised by the customers through the change request form which contains all the information regarding the change. From the defect data analysis, it is been identified that defects are getting raised during the software maintenance, due to Insufficient / Incomplete information in Change Request Log, Insufficient or Incomplete information in Change Request form, New Requirement and Requirement Changes. From the defect report, and it is been understood that the current change request form is insufficient for the change request process. Analyzed the missing attributes in the Change Request Form and designed a New Change Request Form for Process improvement.

REFERENCES

- [1] Pankaj Jalote (2000), "CMM in Practise, Processes for Executing Software Projects at Infosys", Addison-Wesley,2000.
- [2] Marke, Janth, Tanja Torai & Anno terrla, "Difficulties in establishing a defect management process. A Case study" University of Kuopio, Department of Computer Science, PL 1627, 70211 Kuopio, Finland.
- [3] Sakthi Kumaresh, R.Baskaran,(2010), "Defect Analysis and Prevention for Software Process Quality Improvement", International Journal of Computer Applications (0975-8887) volume 8-No.7 October 2010.
- [4] Manoj Deshmukh, "Change management", White Paper.



D.Kavitha is a Assistant Pprofessor in the Department of Computer Science and Applications, Nazareth College of Arts and Science, Chennai, India. She is M.C.A., M.Phil., (Ph.D). She is also a Research Scholar ,(BHARATHIYAR UNIVERSITY), Area of interest is Software Engineering. Published papers in National and International conferences and journals. Eight years of experience in teaching undergraduate as well as postgraduate students.

The Change Request Form contains the following information's:

Table 2: CHANGE REQUEST FORM

Field	Explanation
Product & Project ID/Name:	Product Name followed by project id/Name
Request No. : (By Project Team)	The Change Request/Problem Report No. is a key to identify the particular Change or the Problem report. It is a running sequence No. Number (it can be taken from the CR log.)
Request By :	The person making the request
Request Date	The date when the CR/PR is received.
Customer	Name of the customer in case of customization projects
Problem Reported Date	In case of Bug reported, the date of the reporting. In case of other teams like PMG, the date of reporting of the problem
Reported By :	The name of the person reporting the problem
RTT ID or issue id	
Brief Description/Nature of the change or problem	Brief Description/Nature of the change or problem
Priority /Severity Level	Severity of the CR/PR 1: Show stopper: The issue is stopping work and needs immediate resolution 2: High – The CR/PR is affecting the major processing of the system. 3: Medium – The affected functionality of the CR/PR is not major. 4: Low – Only the minor changes (cosmetic) are requested.
Change Request Decision	Indicate if this CR is accepted/rejected/postponed.
Module(s)	The modules which are affected by this CR
Affected Items	Specific CI's which are impacted
Job Assigned to :	Name of the person working on the CR. . There may be different persons handling different CI's and they can be mentioned here against each CI
1	
2	
Effort Estimate (Person Days)	Estimated efforts required to carry out the change.
CR/Solution Approved By (PL) :	Name of the person approving the CR
Status	Status of the Change Request Open/Closed/ Pending for approval from CCB
Closed on	Date of closure of CR