Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 5, Issue. 2, February 2016, pg.132 – 138

The CMM Level for Reducing Defects and Increasing Quality and Productivity

[1]ASHWIN TOMAR, [2]V.M.THAKARE

[1]Computer Science, MCA Department, under Pune University, Pune, India
Siddhant Institute of Computer Application
[2]Computer Engineering & Science, Amravati University, Amravati, India
P.G. Dept. of Computer Science
[1] mcatomarashwin@gmail.com, [2] Vilthakare@yahoo.co.in

Abstract:

CMM - capability maturity model is a process improvement model, a framework which is used as powerful tool for understanding and improving performance. There are five levels of which Level fifth is highest. In this paper various variables and their behavior are studied. The CMM level and number of defects present at various STAGES are based on data file of defect. The relationship between the variables is established using regression technique.

Keywords: CMM – Capability Maturity Model, DRE – Defect removal efficiency, DP – defect potential.

Introduction:

CMM - capability maturity model is a process improvement model, a framework. A defect is nothing but a variance from the given specification, a hidden or coding error. Defects are undesirable, they cause increase in risk, revenue loss to the customer if they remain in the final product. Rectification may prove costly and results in risk and loss to the agency too. This is persistent from the innumerable surveys conducted by well recognized agencies[1].

Defects are brought to the notice of project team by a process known as bug reporting. Defect reports are used to alert software programmers about the defect and gives them sufficient information to find root cause of problem and fix it. It provides information to technical writers and add test cases in the regression suite for next release[2].

In this paper the relationship between CMM level and number of defect present in various STAGES like Requirement (REQ), Design (DSN), Coding (CDE), Document (DOC) is established based on data file collected by Dr Caper Jones. The paper is arranged in following sequence introduction, literature survey, CMM levels, data collection followed by result, conclusion and references.

Literature Survey:

Dr Caper Jones worked on measuring area of software using metrics[3]. Hazif A.Khan et.al[4] proposed a defect management process model & establish it in an organization to reduce the number of defects and produce a quality software product. The defects are detected, removed and prevented so that quality is increased. Models on quality assurance are proposed by different researchers. One such example is a quality assurance model for analysis phases by R. Ejaz, M. Nazmeen et.al[5]. Manju lata et.al[6] proposed a quality assurance model to optimize cost. B. H. Wu[7] proposed a defect prediction model. H.A.Khan[8] to establish a defect management process model in an organization to reduce the number of defects and produce a quality software product.

Capability Maturity Model Integration(CMMI), a process improvement model, a framework was first described in the 1989 book Managing the Software Process by Watts Humphrey. Predictability, effectiveness and control of an organization's software processes are believed to improve as the organization moves up from first to five levels. CMM is used to assess an organization against a scale of five process maturity levels. Each level ranks the organization according to its standardization of processes in the subject area being assessed. The subject areas can be diverse as software engineering, systems engineering, project management, risk management, system acquisition, information technology(IT) services and personnel management.

CMM is a way to develop and refine an organization's processes related to software development. The model identifies and defines five levels of process maturity for an organization. Within each of these maturity levels are KPAs (Key Process Areas) which characterize that level, and for each KPA there are five definitions identified Goals, Commitment, Ability, Measurement, Verification. The KPAs are the stages that organizations must go through on the way to become mature.

CMM Levels:

There are five CMM levels, of which Level 5 is the highest. They are defined and characterized as:

CMM Level 1 is called "Initial". The software process is at CMM Level 1, if it is an ad hoc process, not defined completely. At CMM Level 1, few processes are defined and success, in general, depends on individual effort and heroism.

CMM Level 2 is called "Repeatable". The software process is at CMM Level 2, if the subject company has some basic project management processes, in order to track cost, schedule, and functionality. Software processes are at CMM Level 2, if there is requirements management, project planning, project tracking, subcontract management, QA, and configuration management.

CMM Level 3 is called "Defined". The software process is at CMM Level 3, if the software process is documented, standardized, and integrated into a standard software process. Software processes are at CMM Level 3, if there is process definition, training programs, process focus, integrated software management, software product engineering, intergroup coordination, and peer reviews.

CMM Level 4 is called "Managed". The software process is at CMM Level 4, if the subject company collects detailed data on the software process and product quality, and if both the software process and the software products are quantitatively understood and controlled. Software processes are at CMM Level 4, if there are software quality management (SQM), and quantitative process management. Metrics are used to track productivity, processes, and products. Project performance is predictable, and quality is consistently high.

CMM Level 5 is called "Optimized". The software process is at CMM Level 5, if there is continuous process improvement, if there is quantitative feedback from the process, and from piloting innovative ideas and technologies. Software processes are at CMM Level 5, if there is process change management, and defect prevention technology change management.

There are different defect preventive methods like Inspection, Clean room software development, Checklist, Reviews. The below Table 1 shows application of defect removal technique ***Inspection*** and its percentage effectiveness in various organization rated on basis of CMM level. It shows that at CMM Level 5, the Inspection is 90-100% effective in removing defects.

Table: 1 CMM Level with Inspection effectiveness

CMM Level	Effectiveness
1	< 50 %
2	50 – 65%
3	65 – 75%
4	75 – 90 %
5	90 – 100 %

The Table 2 shows the percent of defect introduced in software development phases varies.

Table: 2 Defects associated with Phase

Software Development	% of Defect
Requirement	20 %
Design	25%
Coding	35%
User Manuals	12 %
Bad Fixes	8 %

Errors in software requirements and software design documents are more frequent as shown in Table 2 than errors in the source code itself, according to *Computer Finance Magazine*. Requirement defects are severe types that other types of defects. **DRE** i.e defect removal efficiency refers to the percentage of total defects found and removed before software applications are delivered to customers. **Defect potentials** refers to the total quantity of bugs or defects that will be found in five software artifacts: requirements, design, code, documents, and bad fixes or secondary defects[9]. If software has Defect potential less than 2.5 % and DRE above 96% then the Quality in the software is considered as Best class. If DRE is below 60%, in such a case the software is of POOR quality. Requirement should be stable and there should not be requirement change more that 2.5% [9].

Data Collection and Application of Regression Technique:

A good data file reported by Dr Caper Jones was taken as basis. Totally 61 cases (0 to 60) of Software Risk Master™ Quality with details on Stage (requirement, design, code, document, bad fixes). The data collected shows variables like Independent variable (CMM level), Dependent variable (Defects). The various short form are used for the Project Stages which are shown in the Table 3.

Table: 3 The short form used for Four Project Stages

Stage	Description	Remark
REQ	Requirement	It is the short form used
DSN	Design	It is the abbreviation used
CDE	Coding	It is the short form used
DOC	Document	It is the abbreviation used

Regression analysis is a form of predictive modelling technique which investigates the relationship between a dependent (target) and independent variable(s) (predictor). This technique is used for forecasting, time series modelling and finding the between the variables. For example, relationship between rash driving and number of road accidents by a driver is best studied through regression. Regression analysis is an important tool for modelling and analyzing data[2]. The below table shows the list of Independent variable with their abbreviation from the data file.

Table: 4 Table showing Independent variable with its description

Independent	Description
STAGE	Variable represents stage like requirement, design, coding, documents
C_LEVEL	Variable represents CMM level like 1,2,3,4,5

The Regression Technique is applied on the data file and after many operations on the data file, the relationship is established between defect, CMM level and STAGES. Table 5 shows number of defects related to CMM level and STAGES like Code, Design, Document, Requirement, Bad Fixes.

Table: 5 Average defects at various stages as per CMM levels

STAGES	0	1	3	5
Bad fixes	230.68	336.94	208	97.86
Code	941.51	1308.9	783.53	351.64
Design	1074.4	1403.46	1012.17	557.41
Documents	472.49	648.88	492.44	252.01
Requirements	790.64	1062.21	850.3	477.23

CMM level defects may decrease is not true specially in case of levels 0 and 1. Results of curve fitting exercises using the data are given below. The linear equation is written as $y = mx + c$ and of exponential it is written as $y = m.e^{-x}$

Table: 6 Finding relationship between Defect & CMM

CMM	Linear		Exponential	
	R Square	***Equation***	***R Square***	***Equation***
Bad fixes	$R^2 = 0.623$	$y = -34.92x + 296.9$	$R^2 = 0.713$	$y = 310.8e^{-0.19x}$
Code	$R^2 = 0.695$	$y = -149.1x + 1182$	$R^2 = 0.768$	$y = 1254.e^{-0.22x}$
Design	$R^2 = 0.658$	$y = -127.4x + 1298.$	$R^2 = 0.709$	$y = 1340.e^{-0.14x}$
Documents	$R^2 = 0.558$	$y = -55.04x + 590.3$	$R^2 = 0.620$	$y = 607.8e^{-0.14x}$
Requirements	$R^2 = 0.516$	$y = -78.41x + 971.5$	$R^2 = 0.573$	$y = 990.6e^{-0.11x}$

RESULT:

ALL best fits reported are Exponential. As per Caper Jones data all stages have exponential decay with increasing CMM Level of 0,1,3 & 5. The graph of Design is above all and next graph is of code. At third postion from top is graph for Requirment. It comes up towards right side and goes down as CMM level Increases. *In linear equation there is constant rate of increase or decrease while in quadratic there is constant addition. In case of exponential there is constant multiplication.*

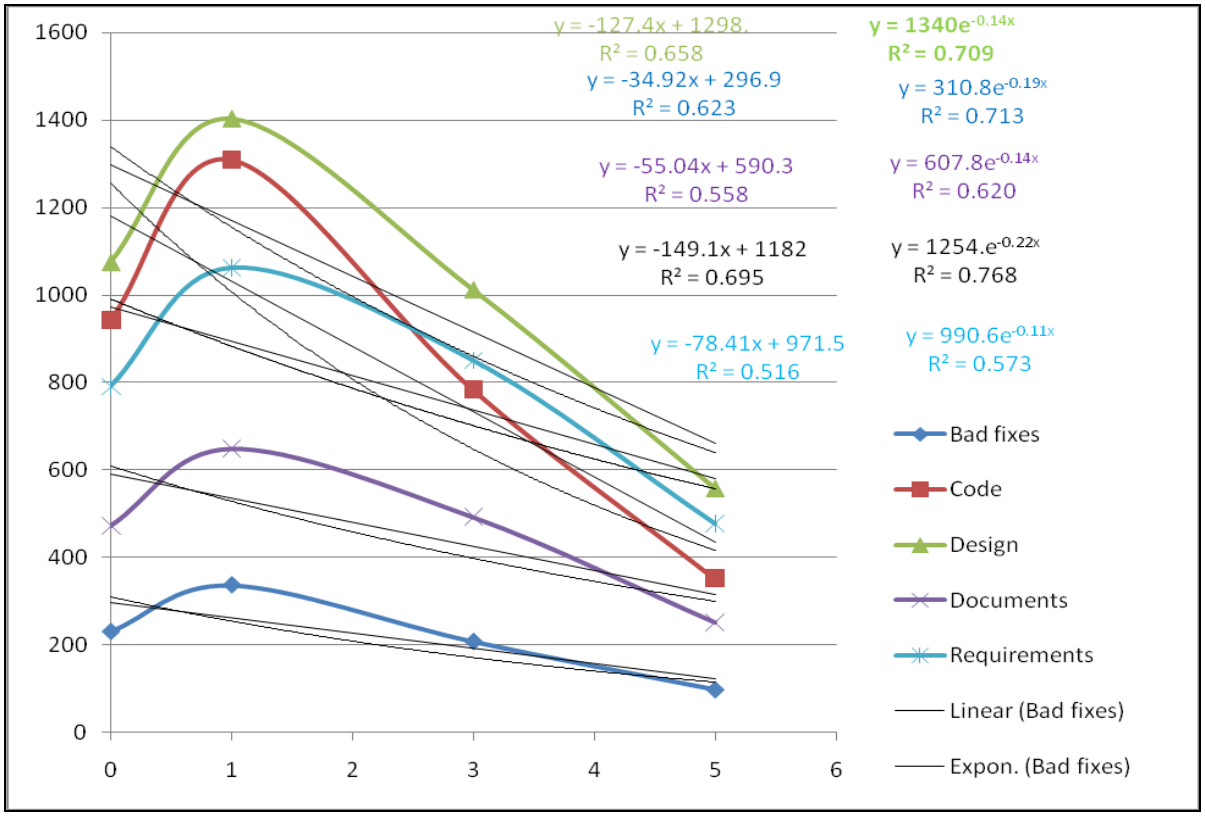

Fig 1 Curve Fitting for Defects Stage Vs CMM level

The graphs Fig 1 indicate that as CMM level increase the number of defects goes on decreasing i.e removal efficiency is more in latter stages of CMM. The number of defects is associated with cost, time and quality. The lesser the number of defect indicates best quality.

Conclusion:

The relationship between CMM level and number of defects present in various STAGES like Requirement (REQ), Design (DSN), Coding (CDE), Document (DOC) are established in terms of prediction. Number of defects predicted are based on CMM level i.e 0,1,3,5. The number of defects present are associated with cost, time and quality. The lesser the number of defect indicate better the quality. Thus CMM level is associated with cost, time, improvement, quality and reduce rework. Based on it good or poor quality software can be rated.

References:

[1] Caper Jones, " Software Quality in 2010 : A survey of state of the Art Survey for 2010", www.spr.com.

[2] www.wikipedia.org/wiki/Bug_tracking_system; www.wikipedia.org / wiki/ Defect_tracking; www.wikipedia.org/wiki/Software_bug; www.wikepedia.com

[3] Capers Jones, "Software Defect Origins and Removal Methods", Dec 28, 2012, www.Namcook.com.

[4] H. A. Khan, "Establishing a Defect Management Process Model for Software Quality Improvement", *International Journal of Future Computer and Communication,* vol. 2, no. 6, 2013.

[5] R. Ejaz, M. Nazmeen, M. Zafar, "A quality assurance model for analysis phase," Proceedings of the [1]*National Software Engineering Conference on NSEC* ' pp.1-4, 2010.

[6] M. Lata, R. Kumar "An Approach to Optimize the Cost of Software Quality Assurance Analysis," IJCA, Vol. 5, no. 8, pp. 1–4, 2010.

[7] B. H. Wu, "Modeling defects in software systems," *2011 IEEE International Conference on* Granular Computing, pp. 739–744, Nov. 2011.

[8] H. A. Khan, "Establishing a Defect Management Process Model for Software Quality Improvement," vol. 2, no. 6, 2013.

[9] Caper Jones, "Measuring Defect Potentials and Defect Removal Efficiency", Software Productive Research, June, 2008.

[10] Ashwin Tomar, V.M.Thakare, " The Study of models for Software Quality Assurance, reuse and predicting a customized model" 2015, Thesis.

