IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 12, Issue 1 (May. - Jun. 2013), PP 50-55
www.iosrjournals.org

A Survey of Software Reliability factor

[1]Nisha Sharma, [2] Ms. Parveen Bano
[1]*Computer Science and Engineering Deptt. , PDM College of Engineering, Bahadurgarh, Haryana*
[2]*Asst. Prof, PDM College of Engineering, Bahadurgarh, Haryana (India)*

Abstract: *Software Reliability is the probability of failure free software which work for a specified period of time in a specified environment. Software Reliability is also an important factor affecting system reliability. In the Existing, Software management approaches like CMM and SPICE gives the quantitative model of software process management. But these methods are based on the accuracy and the reliability of the input data to the system. This paper estimates the reliability factor issues of data for different metrics of software process model under different criteria. This paper is a survey which can be used to design a reliability system based on soft technique*

Key words: *- Software Reliability, Reliability Relevant Software Metrics, software approaches, software measurement, reliability factor.*

I. Introduction

Software reliability is the crucial factor to estimate the software quality as well as to estimate the software cost. There are number of software process model used to analyze the software system. The reliability of these models is based on the reliability of the collected data as well as system.IEEE defines software reliability as "the probability of a software system or component to perform its intended function under the specified operating conditions over the specified period of time". In other words it can also be defined as "the probability of failure-free software operation for a specified period of time in a specified environment". Software reliability is generally accepted software quality since it quantifies software failures that make the system inoperative or risky. For measuring the reliability of a product; the main issue is that of collecting accurate and complete failure and population data that is needed for determining reliability.

Software reliability problems are common because of the high software complexity. Software reliability is considered to be an important attribute of software quality. Other attributes of software quality are: software functionality, software usability, software performance, software serviceability, software capability, software install-ability, software maintainability, and software documentation. Software reliability is inversely related to software complexity. Whereas software complexity is directly related to other software quality factors such as software functionality, capability, etc. Software failures are basically due to software design errors. Software design errors are primarily because of erroneous specifications, immature program coding, insufficient testing, and erroneous usage. Therefore it is required to analyze the reliability of the metrics under different factors that can affect the reliability factor.

1.1 Software Reliability includes the following factors:-

Software reliability factor issues have to be applied in reliability estimation and Software measurement should deal with important development issues are following.

1.1.1 Software Defects/Faults

Software defects are one of the major factors that can decide the time of software delivery. Software defect is also affected the reliability of data as well as system. In the defect of Software and hardware also effected the system reliability .Defect (or Fault or bug) is a result of an entry of erroneous information into software. This could be due to an error in the requirements, design and architecture specifications. A defect is a variance from specification a defect is defined as: "Any significant, unplanned event that occurs during testing that requires subsequent investigation and/or correction. Defects are raised when expected and actual test results differ". The defect is the result of non-conformance to a standard, is related to the aesthetics of the system, or is a request for an enhancement. Defects at this level may be deferred or even ignored. We have to use some method to remove these faults. Debugging is considered in the sense that new faults can be introduced into the software during debugging and the detected faults may not be removed completely. Data collected from real applications show that the proposed model provides both, the traditional reliability measures, and also, some important in-process metrics including the fault removal efficiency and fault introduction rate.

1.1.2 Requirement analyze

Software requirement the product may also co-exist with many other types of software resident on the computer including games, video, audio and various programs downloaded from the Internet, etc. It is known that the failure rate of software often depends on the load on the Hardware or their capacity of the hardware. We consider the Software requirement analyze factor like question set, feasibility, reliable environment, interview and survey factor affected the reliability of system.

Feasibility

If the data is impure or some distorted or damaged data is there the quantitative results driven from these models are also not reliable. Therefore it is required to analyze the reliability of the metrics under different factors that can affect the reliability factor.

Survey

Questionnaires are a popular method of data collection. The advantages of using a questionnaire are well-known: data can be collected quickly on specific items; these data can be easily transferred into forms allowing proper analyses of the required project; and data collection tasks can be delegated to less expensive field staff. The questionnaire aims to check up the requirements engineering practice within your organization pointing out different viewpoints.

Interview

Interviews are used to obtain information from an individual or from groups for an interview, the permission from the top management is necessary Information obtained through an interview can be very useful. It can give us the practical knowledge regarding the activities carried out in a company. An interview can provide information about our objectives, existing system and its drawbacks and requirements of the new system.

1.1.3 Cost

Cost important factor affected the reliability of software and there factor are programmer, software architecture and ratio between person to resources. Some other factor of the cost in the Manpower cost,faliure cost, prevention cost of the software.Programmer knowledge is affected the cost of software & good knowledge programmer is reducing the cost of software. As reliability is concerned with the frequency of different types of failures we need to have a clear and unambiguous classification of failures. This failure classification will have to be from the users' perspective, as we are trying to capture the reliability experience of the user. Ratio between the people to resources also affected the cost of software. Reliability factors like prototype reliability and prototype integration. Reasoning and modeling are necessary in order to achieve desired levels of reliability both during design and usage of software systems.

1.1.4 Size estimation

Size of software depends on the no. of lines and coding of the program. The availability would depend on a number of factors including the system size and configuration and on dates the release was launched and installed. Software metrics provide a quantitative basis for planning and predicting software development processes. Therefore the quality of software can be controlled and improved easily. The software industry does not have standard metric and measurement practices.

1.1.5 Measuring Reliability of Software Products

The main issue is that of collecting accurate and complete failure and population data that is needed for determining Reliability. Some of the key assumptions in measuring reliability in this manner are:

1) All failures have "equal" reliability impact, and that there is a single number that captures the reliability of the product under all usage scenarios.
2) All the F failures can be recorded, and the population size N is known.
3) By normalizing by T*N (and T is generally measured in days,) it is assumed that the system is in use for same amount of time each data (generally assumed to be 24 hours.)
4) The operational profile is consistent across the user base.

II. Software Reliability Approach

Software reliability is difficult to calculate, there are no of approaches available to improve the reliability of software. Each approach has strengths and weaknesses, and currently there is not one approach that is more accepted in industry over another.

2.1 FAULT REMOVAL EFFICIENCY IN SOFTWARE RELIABILITY GROWTH MODEL

Fault removal efficiency in the existing models is limited. This paper aims to incorporate the fault removal efficiency in software reliability growth modeling. In this paper imperfect debugging is considered in the sense that new faults can be introduced into the software during debugging and the detected faults may not be removed completely. This paper represent the reliability of the data depends on the source of the data collection. This paper incorporates fault removal efficiency into software reliability growth model. Data collected from real applications show that the proposed model provides both, the traditional reliability measures, and also, some important in-process metrics including the fault removal efficiency and fault introduction rate.

2.2 Analysis of Software Metrics

With the widespread acceptance of the Object-Oriented (OO) paradigm came the realization that, while existing software metrics would be useful, they would not be accurate to assess the use of abstractions such as encapsulation and inheritance in this new paradigm. As a result many new metrics for analyzing object-oriented systems have been defined. In this section, some metrics which have been proposed in the literature are presented in this section; some metrics which have been proposed in the literature are presented.

2.3 Reliability Analysis Using Weighted Combinational Models

In this paper, three weighted combinational models, namely, equal, linear, and nonlinear weight, are proposed for reliability estimation of web-based software. Weighted combinational model to a web-based ERP system are compared with traditional SRGMs and show that the proposed models can give better accuracy in software reliability prediction

2.4 Collaborative reliability prediction approach

The collaborative reliability prediction approach mainly focuses on service-oriented systems, since the Web services are usually publicly accessible and are invoked by a lot of service users. Therefore, it is possible to obtain the past failure data of Web services from different service users. The predicted failure probabilities of the elementary Web services are composed using different compositional structures to predict the reliability of the whole service-oriented system. We will address this Web service failure correlation problem in this approach.

III. Survey:-

This view of product reliability also provides a practical framework for improving the product reliability experience of users.

3.1 BRIEF DESCRIPTION OF COMPANIES:

In this section we present brief introduction of different software organization where we conducted the survey. We conducted the survey in six different software development organizations. These organizations are much matured in the field of IT. Because of privacy policies we are using Company a, Company B etc instead of their original names. We are presenting here a short description of the companies.

3.1.1 COMPANY A

Company A is Software Company that combines deep technology acumen with industry knowledge to develop market-leading software products. A dedicated global organization, It offers innovative software solutions and the unmatched skills of experienced professionals to enable to meet business goals and achieve high performance. It also helps companies and governments achieve business results faster and at lower cost. It also builds their own software; also collaborate with market leaders such as Microsoft, Oracle and SAP to develop industry enhancements.

3.1.2 COMPANY B

Company B cited as a leader in global IT infrastructure outsourcing services by independent research firm. It is the second largest IT Company in India. It has developed a corporate R&D wing called Software Engineering and Technology Labs. Also experts in it discusses a multi-step approach that combines the strengths of IT Infrastructure Library (ITIL) and Control Objectives for Information and related Technology (COBIT), defect reduction capabilities of Six Sigma and waste elimination tools of Lean to develop effective ITSM processes

.3.1.3 COMPANY C

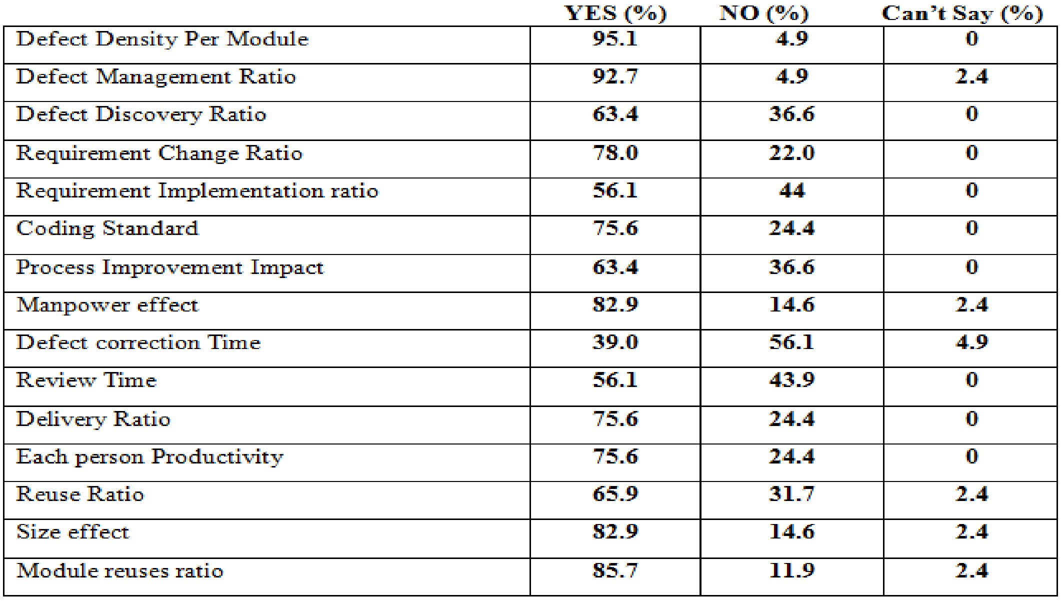
Company C is a global technology and innovation company that stands for progress. With operations in over 170 countries, around the world invent and integrate hardware, software and services to help forward-thinking enterprises, institutions and people everywhere succeed in building a smarter planet. It has been presented in India since 1992. The diversity and breadth of the entire portfolio of research, consulting, solutions, services, systems and software, uniquely distinguishes. It has been expanding its footprint in India - and has a presence in over 200 cities and towns across the country - either directly or through its strong business partner network.

3.1.4 COMPANY D

It is an Indian IT services, business solutions and outsourcing company headquartered in Mumbai, India. It is the largest provider of information technology in Asia and second largest provider of business process outsourcing services in India. It was the first Indian company to make forays into the US market with clients ranging from IBM, American Express, and Sega etc. It is presently the top software services firm in Asia.

3.1.5 COMPANY E

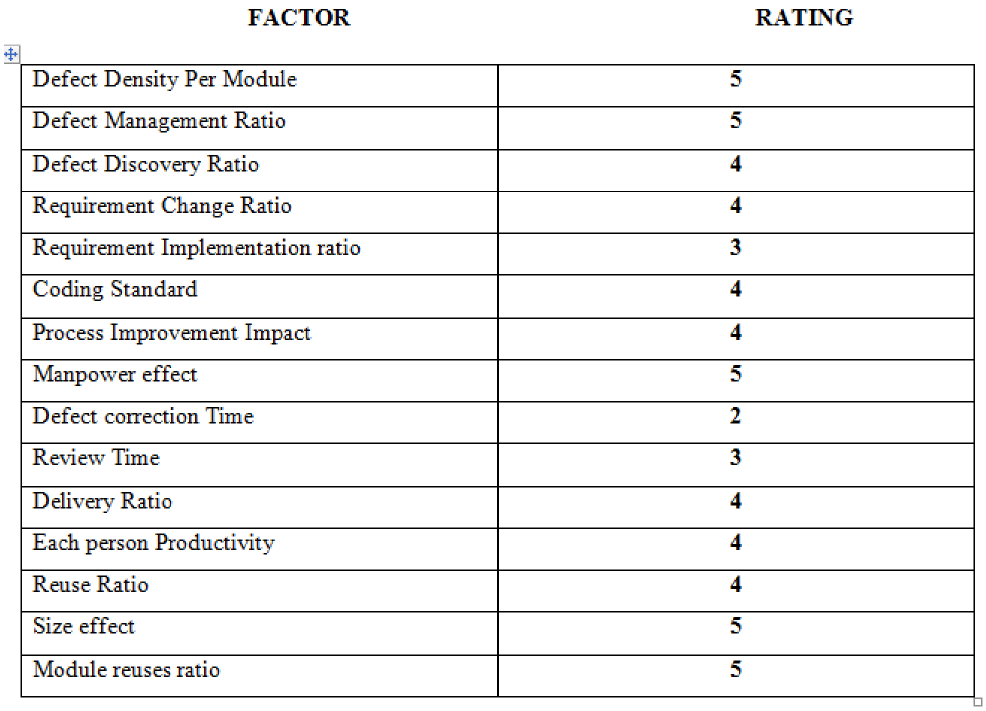
Company E is a UK based company with a strong base of clients and Strategic Technology partners in various parts of the world. It has been providing IT services to a variety of businesses since 2003. It has dedicated unit that deals with IT products and services for 'Small and Medium' size businesses. Apart from their dedicated units based in UK and India, they also work very closely with their UK based and Offshore Strategic Technology partners who specialize in different IT Products, Services and Technologies.



	YES (%)	NO (%)	Can't Say (%)
Defect Density Per Module	95.1	4.9	0
Defect Management Ratio	92.7	4.9	2.4
Defect Discovery Ratio	63.4	36.6	0
Requirement Change Ratio	78.0	22.0	0
Requirement Implementation ratio	56.1	44	0
Coding Standard	75.6	24.4	0
Process Improvement Impact	63.4	36.6	0
Manpower effect	82.9	14.6	2.4
Defect correction Time	39.0	56.1	4.9
Review Time	56.1	43.9	0
Delivery Ratio	75.6	24.4	0
Each person Productivity	75.6	24.4	0
Reuse Ratio	65.9	31.7	2.4
Size effect	82.9	14.6	2.4
Module reuses ratio	85.7	11.9	2.4

2.2 Survey Results

The results of survey questionnaire play key role to observe and know the views of practitioners and professionals about a specific problem(s).In this paper present the survey results obtained from different personnel (professionals, practitioners) of different software organizations.



FACTOR	RATING
Defect Density Per Module	5
Defect Management Ratio	5
Defect Discovery Ratio	4
Requirement Change Ratio	4
Requirement Implementation ratio	3
Coding Standard	4
Process Improvement Impact	4
Manpower effect	5
Defect correction Time	2
Review Time	3
Delivery Ratio	4
Each person Productivity	4
Reuse Ratio	4
Size effect	5
Module reuses ratio	5

3.2.1 ANALYSIS:

The analysis of collected data through the survey results is very important and difficult task. We have collected a lot of data during the survey, which has a direct or indirect relationship with our research area. We have analyzed the collected data from the survey and extract the most relevant data to our research questions.The results obtained from the survey questionnaire regarding use of Defect Reports for improving Software Processes are presented in above section. Furthermore, this section presents the analysis of results of survey questionnaire according to the flow of research questions. Data shown in table 1 is shown in the form of 1-5 Rating. Rating 1 is given to Values between 1-20; Rating 2 is given to values between 21 to 40 and so on.

IV. Conclusion And Future Scope

The reliability of software is important to all the lives software can impact. In the Existing, Software management approaches like CMM and SPICE gives the quantitative model of software process management. But these methods are based on the accuracy and the reliability of the input data to the system. This paper estimates the reliability factor issues of data for different metrics of software process model under different criteria. This paper is a survey which can be used to design a reliability system based on software technique**.** In the future work of this survey result is helpful to estimate the reliability of software by using any approach of software reliability.

References

[1] ANSI/IEEE Standard Glossary of Software Engineering Terminology, IEEE STD-729, 1991.

[2] M. Agrawal, K. Chari, Software Effort, Quality and Cycle Time: A Study of CMM Level 5 Projects, *IEEE Transaction on Software Engineering*, vol. 33, no. 2, pp. 145-156, 2007.

[3] Pankaj Jalote," Measuring Reliability of Software Products", (pp 1-8)

[4] Franz Brosch," Reliability Prediction for Fault-Tolerant Software Architectures",QoSA+ISARCS'11, June 20–24, 2011, Boulder, Colorado, USA. ACM 978-1-4503-0724-6/11/06 (pp 75-84)

[5] I Mandal," Software Reliability Assessment using Artificial Neural Network", International Conference and Workshop on Emerging Trends in Technology (ICWET 2010) – TCET, Mumbai, India ICWET'10, February 26–27, 2010, Mumbai, Maharashtra, India. ACM 978-1-60558-812-4 (pp 698-699)

[6] Aleksandar Dimov," How do we collect data for software reliability estimation?", International Conference on Computer Systems and Technologies-CompSysTech'10 CompSysTech'10, June 17–18, 2010, Sofia, Bulgaria. ACM 978-1-4503-0243-2/10/06 (pp 155-160).

[7] hao-Jung Hsu,"Reliability Analysis Using Weighted Combinational Models for Web-based Software",WWW 2009, April 20–24, 2009, Madrid, Spain. ACM 978-1-60558-487-4/09/04. (pp 1131-1132)

[8] PURNAIAH B.1*, RAMA KRISHNA V.2 AND BALA VENKATA KISHORE G.3, "FAULT REMOVAL EFFICIENCY IN SOFTWARE RELIABILITY GROWTH MODEL" ISSN: 0975-3273 & E-ISSN: 0975-9085, Volume 4, Issue 1, 2012, pp.-74-77.

[9] Goutam Kumar Saha, "Understanding Software Fault Tolerance Using a Concept Map," IEEE Reliability Society Newsletter, Vol. 54, No. 2, June 2008, IEEE Press, USA.

[10] Neelamegan, Dr. M. Punithavalli, “A Survey- Object Oriented quality metrics”, Global journal of Computer Sc.And Technology, Vol 9, no 4, 2009.

[11] Amjan Shaik, Dr C.R.K. Reddy, Dr A Damodaran, “Statistical Analysis for Object Oriented Design Software security metrics”, International Journal of Engineering Science and Technology, Vol 2 (5), pg 1136-1142, 2010.

[12] Dr Kadhim M. Breesam, “Metrics for Object Oriented designfocusing on class Inheritance metrics”, 2nd International Conference on dependability of computer system IEEE, 2007.

[13] E.J. Weyuker, T.J. Ostrand, and R.M. Bell. Comparing the Effectiveness of Several Modeling Methods for Fault Prediction. Empirical Software Eng. Vol 15, No. 3, June 2010.

[14] Huihua Lu “An Iterative Semi-supervised Approach to Software Fault Prediction” ’11, September 20–21, 2011, Banff, Canadat 2011 ACM 978-1-4503-0709-3/11/09 ...$10.00.

[15] Zibin Zheng," Collaborative Reliability Prediction of Service-Oriented Systems", ICSE ’10, May 2-8 2010, Cape Town, South Africa ACM 978-1-60558-719-6/10/05 (pp 35-44)

[16] Sang-Pok Ko," Secure Reliability of Measurement Data through Application of Mechanism Design Theory".