ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

Economic Aspects of Reliability Assessment and Prediction of Open Source Softwares - A Review

Jitendra Singh[1], L.S. Maurya[2]

Research Scholar, IT- Department, SRMSCET, Bareilly, India[1]

HOD, IT- Department, SRMSCET, Bareilly, India[2]

ABSTRACT: **In this paper, Economic aspects of reliability assessment and prediction of OSSs reviewed. It needs a long time for a company to make a reputation for being "reliable", and only a short time to be branded as "unreliable" after shipping a flawed product. In Economic aspects, continual assessment of new product reliability and ongoing control of the reliability of everything shipped are critical necessities in today's competitive business arena. Open Source Communities have successfully developed a great deal of software. Government of India is also promoting usage of open source softwares due to its economic feasibility and security. In this paper various expert's views about OSS reliability have been analyzed. Different characteristics of OSS than proprietary s/w affect the reliability. Furthermore selection of appropriate reliability metrics has been discussed. Various SRGMs and Reliability tools were also discussed. These findings are also analyzed and verified in Economic Aspects. This paper will be helpful for research scholars doing their research in Software Reliability.**

Keywords: Software Reliability, Open Source Softwares, SRGM, Reliability Metrics

I. INTRODUCTION

In today's technological world nearly everyone depends upon the continued functioning of a wide array of complex machinery and equipment for their everyday health, safety, mobility and economic welfare. We expect our cars, computers, electrical appliances, lights, televisions, etc. to function whenever we need them - day after day, year after year. When they fail the results can be catastrophic: injury, loss of life and/or costly lawsuits can occur. More often, repeated failure leads to annoyance, inconvenience and a lasting customer dissatisfaction that can play havoc with the responsible company's marketplace position. In the same way Reliability of Softwares also cosiderable.

It takes a long time for a company to build up a reputation for reliability, and only a short time to be branded as "unreliable" after shipping a flawed product. Continual assessment of new product reliability and ongoing control of the reliability of everything shipped are critical necessities in today's competitive business arena. Software reliability is not as well defined as hardware reliability, but the Software Assurance Technology Center (SATC) at NASA is striving to identify and apply metrics to software products that promote and assess reliability.

The IEEE defines reliability as "The ability of a system or component to perform its required functions under stated conditions for a specified period of time." To most project and software development managers, reliability is equated to correctness, that is, they look to testing and the number of "bugs" found and fixed. While finding and fixing bugs discovered in testing is necessary to assure reliability, a better way is to develop a robust, high quality product through all of the stages of the software lifecycle. That is, the reliability of the delivered code is related to the quality of all of the processes and products of software development; the requirements documentation, the code, test plans, and testing.[10]

Open Source Software provides their users with both free access and the ability to modify the source code. It has influenced almost every dimension of the software development arena. The most successful examples include Linux operating system (Red Hat, Caldera, Suse, Ubantu etc.), Apache Web server, Open Office, Programming languages like Perl, PHP Python, Ruby, Mozilla Firefox Internet browser and Mysql database system. While many products under this model have been successfully developed, it became an interesting alternative for large software development companies. According to netcraft survey (2007) more than 58% web servers are on Apache, which is an open source web server. Software companies are releasing the source code of their commercial products and participating in OSS communities developing it further. Open source software is used all over the world and it has major contribution in software development and Operating system usage.[7]

Software reliability model has long been used as the most important and successful predictor of software quality when it hits the market.[1] Various views have been presented by researchers on the reliability of Open Source Softwares. These researchers have investigated the nature of the Open Source Software development process,

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

the various factors on which the reliability of a software depends and the consequences of continuously updating and modifying such a software.[3]

Companies that can economically design and market products that meet their customers' reliability expectations have a strong competitive advantage in today's marketplace.

II. OPEN SOURCE IN INDIA

Now a day open source software has become popular with technology users in India. The benefits of open source - affordability, availability of source code and freedom of choice - have made open source a preferred platform for many Indian organizations and individuals who want to use the power of high quality software which can be freely adapted to their own requirements. India has a diverse range of interest groups promoting the adoption of open source software. Major motivators for these groups include politics and special interests, government programs, outsourcing service requirements, and education as well as skills training. Global issues such as software patents vs. freedom of knowledge and adoption of open standards have also become pivots around which local organizations and activists promote free and open source software.

In this direction step of national government is appreciable because government has established a National Resource Centre for Free and Open Source Software (NRCFOSS) in April 2005 which is a joint venture between a university-based research lab (AUKBC Centre) and the Centre for Development of Advanced Computing (C-DAC) and Institute for Open Technologies and Applications (IOTA) as a joint venture between the state of West Bengal, Jadavpur University and open source industry players (Sun Microsystems and Red Hat). These steps recognize the fact that free and open source software is here to stay and it represents a paradigm shift in computing that is poised to change the Indian IT industry. IOTA's mandate is to promote open source software in government and academia. .[7]

Government organizations, at the national and state levels, have been using open source software to produce internal applications. Recently, e-government procurements have begun to allow open source software to be part of vendor provided solutions. Another important dynamic in driving open source adoption in India involves the efforts of free software groups, open source community members and special interest groups. These groups engage in advocacy as well as offer training and support at the grass root and others levels. Now a day Open Source Software is, developed to serve both high-tech professionals as well as users without deep technical knowledge that leads to higher quality expectations than earlier, when an end-user was seen as a co-developer. Thus *reliability of open source software* becomes an essential need for secure and efficient development of open source software.

III. QUALITY VERSUS RELIABILITY

The everyday usage term "quality of a product" is loosely taken to mean its inherent degree of excellence. In industry, this is made more precise by defining quality to be "conformance to requirements at the start of use". Assuming the product specifications adequately capture customer requirements, the quality level can now be precisely measured by the fraction of units shipped that meet specifications.

But how many of these units still meet specifications after a week of operation? Or after a month, or at the end of a one year warranty period? That is where "reliability" comes in. Quality is a snapshot at the start of life and reliability is a motion picture of the day-by-day operation. Time zero defects are manufacturing mistakes that escaped final test. The additional defects that appear over time are "reliability defects" or reliability fallout.

In order to achieve software reliability, different techniques for measurement have been developed. Their main purpose is to test the software and measure the reliability according to the predefined criteria of techniques. The final result then offers the software developers or users an understanding of the reliability of that software . This process is known as the Software Reliability Engineering can be summarized as the following diagram.

The following diagram of SRE was given by Lyu in 1996. In addition to measurement, there are models for software reliability which define how software should be developed while sustaining reliability.The models are based on some assumptions,state and failure based analysis and mathematical derivation indicating reliability.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

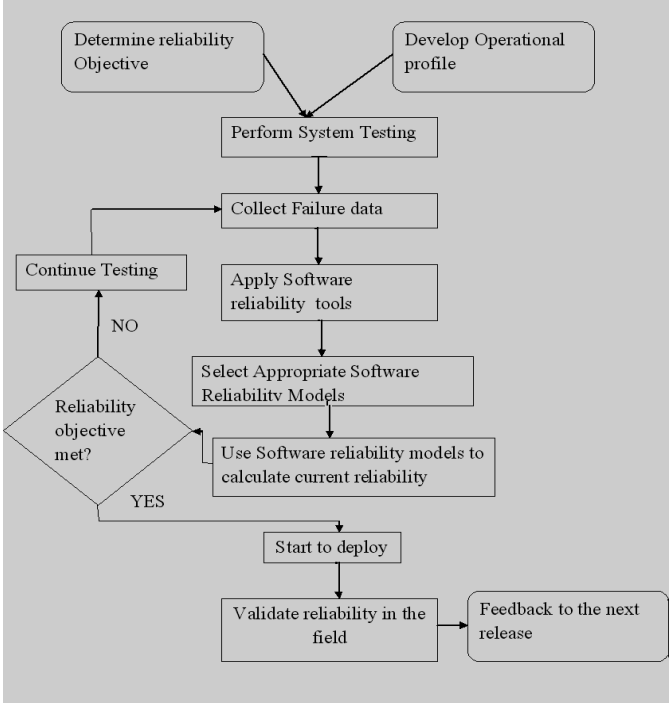

Fig. 1 Software Reliability Engineering process overview

IV. EXPERTS OPINION

Wasserman[2002]gives a modern definition of reliability including time,condition and customer matters: "Reliability is the probability of a product performing its intended function over its specified period of usage,and under specified operating conditions, in a manner that meets or exceeds customer expectations" What happens when a product cannot perform its intended function is called a failure. Following are possble reasons for product failures

A.Design deficiencies

a.Omitting an important customer requirement or design feature

b.Deficiencies in product design, which lead to early failure

c.The design of the process can also have deficiencies, resulting in a defective product.

B.Quality control

a.Due to quality-control problem,inefficient or unsuccessful products which lead to performance problems while being used.

b.Possible damage to products while handling or distribution.

C.Misuse : Possible misuse of the product by customer or during service.

Some researchers are of the view that since the development of an OSS is initiated by personal needs and the users are the developers thus they understand and analyse the requirements in an unambiguous way. Also as compared to the classical method of software development such a process is more flexible.

On the other hand, OSS, with its reliance on selfinterested developers, may be less well suited for developing applications that address problems that developers tend not to face. Also since there is no legal licence that comes along with an OSS there is no reliable technical support for an OSS and there can be reliability issues with the modifications and bug correction solutions provided by other members.[3]

The OSS development mainly relies on the practise of welcoming every enthusiastic individual who would like to contribute in the project.On top of this, the freedom of using ,modifying and distributing OSS leads to more robust software and more diverse business model[Wu and Lin,2001]This freedom of participation constructs the OSS development cycle for both individuals and groups and a typical OSS project follow the cycle summarized as a flow in following figure.

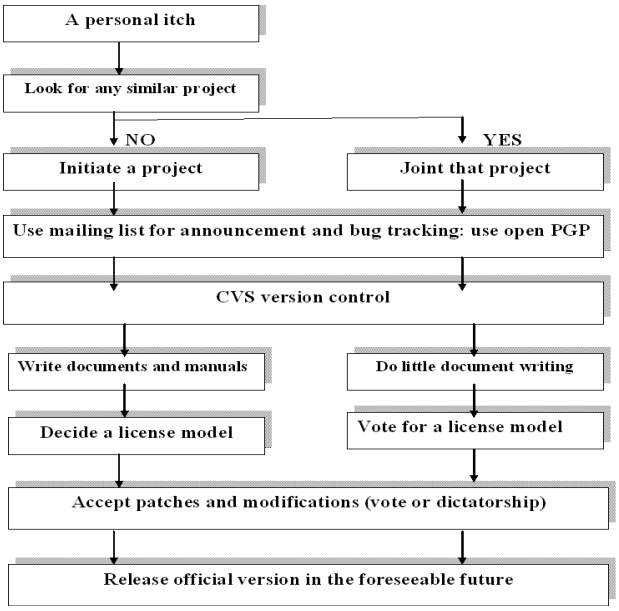

Fig.2 OSS Development Cycle [Wu and Lin,2001]

This typical OSS project development is originated by a personal need which results in a project initiation if any similar one doesn't exist or contribution to the existing one.The Open Source principles of using mailing lists,version control systems, writing documentationand manuals, deciding on a license model,the project matures into its final stege of releasing.[11]

V. FACTORS AFFECTING RELIABILITY

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

Software Reliability which is also important factor affecting system reliability is the probability of failure-free software operation for a specified period of time in a specified environment. Some of the factors affecting reliability are[9]:

1. The user group involved in its usage.
2.Type and rate of failure, where failure can be due to some planned or unplanned events; or human
3.Number of operational units of the software
4. Time the user spends using the software
5. Load on the underlying hardware
6. OSS testing
7.Bug identification and testing
8.Personal expectations and analysis.
9. Community or individual members working on the distributed development of the OSS activities

Above factors make an impact on reliability of OSS and due to that profitability of product also affected.

VI. SELECTION OF SOFTWARE RELIABILITY METRICS

An appropriate and relevant selection of software Reliability metrics causes time and cost effectiveness and lead the project towards economic advantage. Metrics can be defined as "the specific characteristics of the product we want to measure". Software reliability metrics are used for software reliability evaluation and assurance , and they are very important for software reliability because they can be used to :

1) provide quantitative indicator for reliability management;
2) be a trade-off among cost, schedule, and reliability;
3) monitor testing process and forecast testing schedule;
4) evaluate and validate reliability;
5) evaluate software engineering technology from the viewpoint of reliability;
6) interpret reliability behavior (recur to software reliability models).

Selecting criteria are several attributes by means of which the metrics can be compared. We adopt the following five criteria according to[6]:

1. Relevance (to reliability): this criterion reflects the relationship between metrics and software reliability.
2. Experience: this criterion reflects the degree to which this metric has been used and recognized.
3. Correctness: this criterion includes 1) Objectivity. The input and results of this metric can't be easily influenced; 2) Justness. The metric is not partial to any specific result; 3) Precision.
4. Practicality: this metric should be concerned and required in development.
5. Feasibility: this criterion means that 1) the formula of this metric should be understood easily, and supported by tools; 2) data collection should be easily;
3) the results of this metric can be evaluated and confirmed conveniently.

The method for selecting appropriate software reliability metrics based on AHP[6] is a decision-making process which evaluates the applicability of each metric in each development phase subjectively based on AHP and expert judgment. The main process is shown in Figure given below

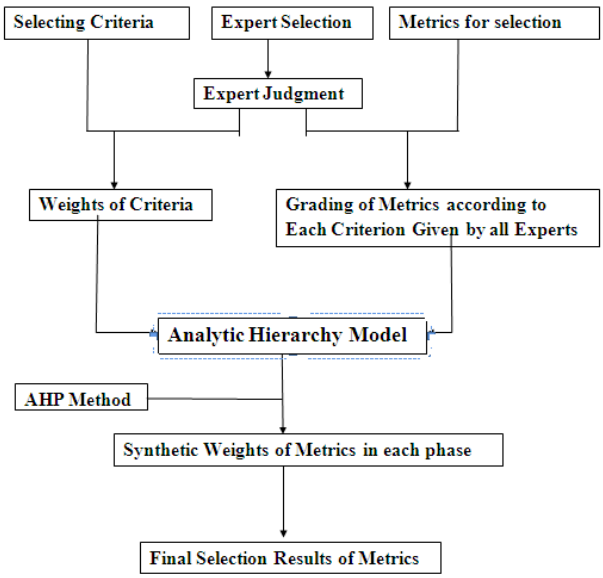

Fig.3 AHP based Reliability Metrics selection method

VII. OSS RELIABILITY GROWTH MODELS

In OSS reliability growth models we are assuming some effort of fault removal.This leads to a variable failure intensity λ(t).Every reliability growth model is based on specific assumptions concerning the change of failure intensity λ(t) through the process of fault removal. It is important that SRGMs should be able to;

a. Predict probability of failure of acomponent or system
b. Estimate the mean time to the next failure
c. Predict number of (remaining) failures during development

SRGMs are classified in two parts based on failure data

1. Time between failure models
a.Jelinski-Moranda
b.Musa-Basic
c.NHPP
d.Geometric
e.Musa-Okumoto
f.Littlewood-Verall

2. Failure count models
a.Generalized Poisson
b.Shick-Wolverton

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

c.Yamada S-Shaped
d.Schneidewind

SRGMs Assumptions vs Practice

Assumptions	In Practice
Software does not change and defects are fixed immediately	Software change rapidly and certain defectsare scheduled to be fixed in later date
Testing Operational profile (OP)	It is difficult to define OP
Constant test effort and independent failure interval	Varying test effort and failure interval
All failures are observable	Testing in a controlled environment may be different from running software in live environment
Collect failure reports	Collect defect reports.not all reports address a failure
Remaining failures are either constant or decreasing	Remaining failures may actually increase due to improper bug fixes

There are various SRGMs to predict and estimate Reliabilty of OSS in Industry but the important thing is "Relevant model for Right software". So by economic point of view Growth models should be selected efficiently by taking sequence steps as:- collect failure data(failure specifiation), examine data(Density distribution vs. Cumulative distribution), Select a model , estimate model parameters , customize model using estimated parameters , Goodness- of- fit test and then make reliability prediction if model will suits.

VIII. OSS RELIABILITY TOOLS

The tool has a user-friendly interface which includes features like autodetection of datatype and a model selection wizard.Following tools generally used for Reliability prediction and estimation.

A).Computer Aided Software Reliability Estimation (CASRE): CASRE is a Software Reliability Engineering program that estimates failure intensity from failure data. It has a convenient user interface, which makes it easy to track software reliability growth as you test your software. Execute this program periodically and plot the ratio of Software Reliability to Software Reliability objective (in terms of failure intensity to failure intensity objective ratio). By tracking software reliability growth, you can uncover possible problems and take timely corrective action. CASRE also guides you in deciding when the Software Reliability has reached the Software Reliability objective, such that the software can be released.

B).COQUALMO:Constructive QUALity Model (COQUALMO) , formerly called CODEFMO, is an estimation model that can be used for predicting the number of residual defects/KSLOC (Thousands of Source Lines of Code) or defects/FP (Function Point) in a software product.

C). Frestimate: SoftRel developed this software reliability tool provides basic software prediction capabilities. FRestimate is also available as components. It includes the WhenToStop software estimation for determining When To Stop testing. The components include: SoftRel's Short Cut Model, SoftRel's Full scale Model, Rome Lab Model, SSQA model, CMM model, and the WhenToStop module. See the website for a description of each component.

D). SoftRel - Software Reliability Prediction: SoftRel develops predictive models. Namely numerical and classification models via data mining, knowledge discovery, and knowledge extraction.

IX. ECONOMIC ASPECTS OF RELIABILITY ASSESSMENT

Reliability is a major economic factor in determining a product's success. Companies that can economically design and market products that meet their customers' reliability expectations have a strong competitive advantage in today's marketplace. "Some failures have serious social consequences and this should be taken into account when planning reliability studies" Accurate prediction and control of reliability plays an important role in the profitability of a product. Service costs for products within the warranty period or under a service contract are a major expense and a significant pricing factor. Proper spare part stocking and support personnel hiring and training also depend upon good reliability fallout predictions. On the other hand, missing reliability targets may invoke contractual penalties and cost future business.

It is up to the reliability engineer (and the relevant customer) to define what constitutes a failure in any reliability study. More resource (test time and test units) should be planned for when an incorrect reliability assessment could negatively impact safety.

Open source is becoming more mainstream in the economy. A variety of interests on the part of industry and academia are encouraging adoption of open source softwares. Demand for open source has followed the increase in demand for information technologies in all sectors. While most open source software is still used

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 1, Issue 10, December 2012

internally in organizations, the potential for significant contributions back to global open source projects is increasing. Growth in contributions may accelerate when the need to scratch an itch (or many itches) to support local needs grows. As demand grows reliability becomes an essential tool for the growth of OSS as well as Indian economy from software. There are varieties of models available for evaluating software reliability that could be used in different situations.

All the above-mentioned approaches are used to predict the reliability of the software in the later stages of the life cycle thus if any modification is needed it becomes cost effective. Thus future prospects and research may be in the field of "reliability prediction for open source softwares in early phases" which will reduce the cost effectiveness of the overall project.

X. CONCLUSION

In this paper , Economic aspects of reliability assessment and prediction of OSSs reviewed. In today's technological world we depend on demand and expect reliable products. Shipping unreliable softwares can destroy a company's reputation. Reliability is "quality changing over times". We can assume quality as a snapshot at the start of life and reliability as a motion picture of day –by – day operation.

Motivation by Indian Government toward OSS also analyzed in Economic view. Expert views over OSS and their Reliability also taken in account. Previously mentioned factors which make an impact on reliability of OSS and due to that profitability of product also affected. We analyzed that an appropriate and relevant selection of software reliability metrics causes time and cost effectiveness and lead the project towards economic advantage. We discussed that among various SRGMs, it is economically considerable fact to "select relevant model for right right software". Finally we have taken economic aspects of Industrial Arena in the field of Reliability of OSS and conclude the paper with this remark that continual assessment of new softwares reliability and ongoing control of the reliability of everything shipped are critical necessities in today's competitive business arena.

XI. ACKNOWLEDGEMENT

We would like to express our sincere gratitude towards our chairman Shri Dev Murti ji and Prof. Prabhakar Gupta (Dean).Without these members this manuscript is not possible. Under their cooperation we are able to make this manuscript successful. Last but not the least we are heartily thankful to your institution which bestowed this opportunity to us.

REFERENCES

[1]Zouh, Y.; Davis, J.;, "Open source software reliability model: an empirical approach," in ICSE:5-WOSSE, Open source software engineering, pp 1-6, 2005

[2]Jalote, P.; Murphy, B.; Garzia, M.; Errez, B.;, "Measuring Reliability of Software Products" IEEE,MSR—TR-2004-145,14 pages,2004

[3]Yadav,Dr. Kalpna; Jain, Eshna;Bhatia, jaspreet; "Comparative Study of Open Source Software Reliability Assessment Models", IEEE-ICCRD, vol 1, pages- 384-388, 11-13 march 2011

[4]Tamura, Y.; Yamada, S.; , " Comparison of Software Reliability Assessment Methods for Open Source Software and Reliability Assessment Tool," Journal of Computer Science, vol. 2, pp. 489-495 , 2006.

[5]Goel, A.L.; , "Software Reliability Models: Assumptions, Limitations, and Applicability," Software Engineering, IEEE Transactions on , vol.SE-11, no.12, pp. 1411- 1423, Dec. 1985.

[6]Haifeng Li.; Minyan Lu.;Qiuying Li;" Software Reliability Metrics Selecting Method Based on Analytic Hierarchy Process", IEEE-QSIC, pages 337-346,vol-6 **,**27-28 oct 2006

[7]Jha Sanjeev; Dwivedi Dr. A.K.D; Tiwari Dr. Pramod; "Reliability Models and Open Source software-an empirical study", IEEE-ICCIC,pages-1-5, 2010

[8] Software Reliability: Measurement, Prediction, Application; John D. Musa (McGraw-Hill series in Software Engineering and Technology)

[9]Software Reliability Engineering;John D. Musa(McGraw-Hill series on Software Development)

[10]Software Reliability Methods By Doron A. Peled; ISBN 10: 0387951067

[**11**]Kemal Kagan Ishtan :"An approach to establish a software reliability model for different free and open source software development paradigms" May 2011

[12] Bugzilla, http://www.bugzilla.org/.