Proceedings of the International Conference on Intelligent Sustainable Systems (ICISS 2017)
IEEE Xplore Compliant - Part Number:CFP17M19-ART, ISBN:978-1-5386-1959-9

Software reliability models –A Comparative study

M Anila[1], N Sirisha[1]
[1] Department of Computer Science and Engineering
MLR Institute of Technology
Hyderabad, India
Email: anilarao.m@gmail.com

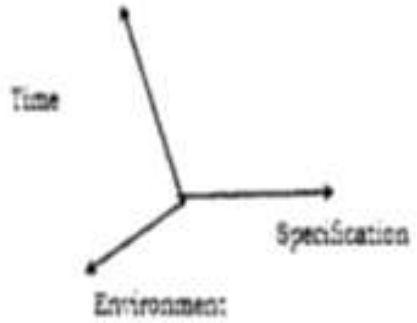
R Karthik[2]
[2] Department of Electronics and Communication Engineering,
MLR Institute of Technology
Hyderabad, India
Email: rayam16@gmail.com

***Abstract*— Software reliability defined as the probability of failure free operation for a specified period of time in a specified environment. Developing reliable software is one of the most difficult problems facing the reliability is known from customer feedback in the form of defects. A number of software reliability models have evolved since early 1970 to predict software reliability. This paper surveys the evolution of reliability models and purpose an approach in the selection of an appropriate model for different phases of software development lifecycle (SDLC).**

Keywords— Defects; SDLC; SRGM; Probability;

I. INTRODUCTION

The domain of reliability has three dimensions vis-à-vis quality, which is viewed as a one-dimensional aspect. In quality, the main consideration is on the product meeting the specification parameters. On the other hand, the reliability adds two more dimensions, one each for time and environment to an one dimensional aspect of quality, thus making it a three-dimensional domain as shown in Figure 1.



a.

Fig. 1. Three dimensional view of reliability

It is easier to measure the quality of software and a number of quality measures have been standardized over a period of time. The measurement of software reliability though has progressed since 1970; it is yet to reach the level of maturity.

II. SOFTWARE RELIABILITY VERSUS HARDWARE RELIABILITY

It is believed that "software never breaks"[4]. It is also considered that software does not age with time unlike

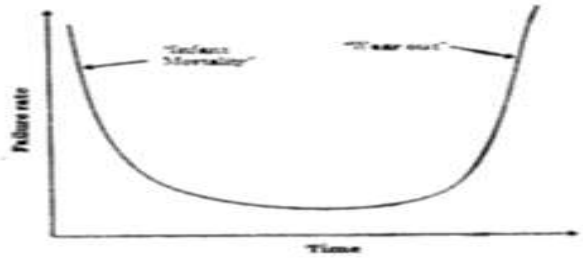


Fig. 2. Hardware aging curve

hardware. Which follows the bath-tub curve as shown in Fig 2. The reliability R(t), is related to failure probability F(t) by : R(t)=1-F(t).

Software reliability, however, does not show the same characteristics as the hardware reliability curve. Ideally, the failure rate curve for software should become flat once the initial high failures have been corrected. In practice, software will undergo changes during its lifetime. As changes are made, it is likely that some defects will be introduces, causing the failure rate curve to spike at the point of changes as shown in Figure 3.

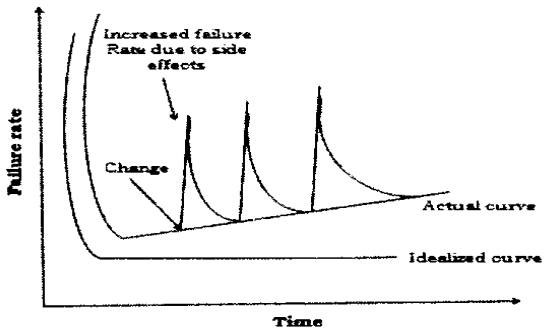


Fig. 3. Software failure curve

III. SOFTWARE RELIABILITY DURING SDLC

It is considered that once the software has run correctly at the time of testing, it will be correct forever. A number of catastrophic events, which have been witnessed over a period of time as a result of software failure of patriot missle to intercept the scud missile during the Gulf war was due to truncation error of 0.00000009 second in precision in every 10^{th} of a second. The failure resulted in a loss of 28 lives. There are a number of other examples like therac-25, Ariano-5, etc and many more scary stories to tell. This mistakes us wonder whether the software is reliable at all, whether we should use software in safety- critical application.

If we think that software is not reliable and as such, we should not use it, we are going back to history when computers were not there. There is no alternative but to use software in almost every application because all functions to be performed by hardware would make the system un-yielding. Today, computer software is the single most important technology on the world stage [5].but, it is paramount that the software shipped to the customer should be as reliable as possible.as such, we should attempt to apply techniques to achieve high reliable software. Reliability cannot be measured directly. However, it can be predicted based on indirect measures using reliability forecasting models. In this paper, we focus on fault fore-casting technique and in that too the emphasis will be on reliability models.

IV. SOFTWARE RELIABILITY MODELS.

Once the first aim in achieving the high reliable software has been met, the next important goal is to apply software reliability models to predict or estimate reliability so that the software product is delivered to the customer with high-degree of confidence. Over 200 models of software reliability have emerged with pioneering works by [Mora 72, Mora 75a, shoo 72. Shoo 73, shoo 76, shoo 77s, and shoo 77b, cout 73].the basic approach is to model past failure data to predict future behaviour.

V. SELECTION OF RELIABILITY MODEL

There is a real challenge in selecting a reliability model foe a specific application.. A comparative study of a number of reliability models has been carried out.as an outcome of this study, this of paper suggests the approach for selection of a reliability model for different phases of software development life cycle (SDLC) in the following section.

There are a number of early prediction models which can be applied during the requirement, design and coding phases of SDLC.The most important model is Musa prediction model. The model is discussed below.

A. Musa prediction model:

Amongst the models listed above, Musa prediction model has the widest distribution and was developed by john musa of AT&T Bell laboratories [Musa 75, Musa 78, Musa79s, Musa 79b, Musa80, Musa 87].Musa has been a leading contributor in this field and has been a major proponent of model to aid in determining the reliability of software.

Musa model follows a Poisson's distribution with mean value function where, β_0 gives the overall number of faults. The failure intensity function for this module is

$$\lambda(t) = \mu^{1}(t) = \beta_0\beta_1 \exp(-\beta_1 t)$$

$$\mu(t) = \beta_0 [1 - \exp(-\beta_1 t)] \text{ where } \beta_0 \beta 1 > 0$$

B. Practical Application of SRGMs

Tondomcomputer [7] have applied the SRGMs in predicting the number of software defects remaining in the software. The result of their research [7] shows the suitability and predictive ability of software growth models as shown in Table 1.

TABLE 1: SUITABILITY AND PREDICTIVE ABILITY OF SOFTWARE GROWTH MODELS

Release	Predicted residual detects	Defects in first year (field defects)
1	33	34
2/3	33	29
4	10	9

It can be seen from the above result that the predicted values are close to the actual defects observed during operation.

VI. CONCLUSION

The selection of software reliability model is a difficult task for the practitioner of software reliability. Over 200 models have evolved since early 1970s with more models being developed every year. A comparative study of the important software reliability of an appropriate model for a particular phase of software development life cycle has been suggested. The paper also recommends the use of SRGMs during the testing phase of SDLC. A model which can fit into and cater for all the phases of software development encompassing requirement, design, implementation, testing and validation is yet to be evolved.

Software reliability is a fertile field for research and there is enough scope for development of new reliability models.

REFERENCES

[1] Ch.AliAsad and etcÄn Approach for software reliability model selection",proceedings of the 28 Annual international computer software and Application conference.

[2] C. Smidt etc., "An approach to early reliability prediction", university of Maryland at college,park IEEE 1996.

[3] A handbook of software reliability engineering", IEEE computer society press.

[4] Jianto pan "Software reliability", Carnegie mellon university.

[5] Swapna S. Gokhale, etc. "An Analytical Approach to Architecture-based software reliability prediction", proceedings of the IEEE international computer performance and dependability symposium (IPDS'98).