



Software Reliability Chain Model

Pawan Kumar Chaurasia
Department of Information Technology,
Babasaheb Bhimrao Ambedkar University
Lucknow (U.P), India

Abstract: Today most of the people depend on electronic devices or handheld devices. With rapid changes in hardware and software, it affects cost, size, complexity and features of system. Due to increase complexity, faults and failures also increase/decrease in a same manner. Various disaster and the accidents are the examples of failure software. Author focus on the causes of failure and try to build chain models for various error, fault and failure. Software reliability are used for different purposes: system reliability estimation during development, decision making, maintenance recommendation and proposed for new version. In this paper, author tried to collect all the information related to failure of the software and establish the various chain models.

Keywords: Software Reliability, Error, Fault, Failure, Fault Prediction, Fault Tolerance, Fault Prevention, Operational Profile.

I. INTRODUCTION

In today era, most of the people depend on electronic devices, handheld devices or appliances like Mobile, Laptop, Internet, Microwave etc [1]. The function of these devices depends on the correct operation of the system. To use these electronic items, most of the organization provides a manual, how to use these devices. To avoid the failure of the system, customers must follow the rules. From last two decade, these electronic items are ascendant by Integrated Circuits (IC's). It becomes complex with the increase of function, which cause the failure of the system. With the use of the software, cost and size of the system is reduced. So, the dependency of the software is increased which cause the trouble in future and make a system failure. Most of the cases of software failures have various types of high profile programs and loss the data [2] due to high complexity. Reliability is one of the important pertain for industry and researchers [3]. Reliability behaves same as life and death. The system which is used by the people in different forms like: shopping, entertainment, internet, aircraft, telecommunication and home appliances are unreliable. To make system reliable, different professionals proposed so many standards, models and formulas to track the cause of the failure.

Software reliability is defined as the probability of failure free operations from a specified period of time in a specified environment. Failure rate depends on the number of failure occur in different interval of time. Cause of failure depends on the nature of the input. Nature of inputs depends on the execution of the program and the operational profile of the software. If operational profile is changed, reliability of software is also change. An operational profile of software is a key factor to analyze the failure rate of the software and it is the major consequences to estimate software reliability. Survey and literature shows several models for estimating reliability of software. This paper is organized in three sections, Section I, depicts about the introduction part of the software reliability and various definition of failure and fault. In section II, proposed the two chain model for failure chain and software reliability chain. In the last section author conclude the paper with future scope of the paper.

II. SOFTWARE RELIABILITY

Today every people want safety and to be secured when they used any software/hardware devices. Safety means, reliable system and easy to handle without any failure occur. Reliability is a measure of how the system is utilized by the user for the required function. Software reliability is also defined as the probability of the system, over a given period of time, that the system will give the performance correctly, as expected by the user in a specified time. In figure 1, there are two chain method, software reliability chain and failure chain. In reliability chain four components are used: Input type, Operational Profile, Reliability and Failure Chain (Error, Fault and Failure).

A. SOFTWARE RELIABILITY CHAIN:

Software reliability is an important attribute of software. Input data, Operational Profile and Reliability build the chain of software reliability. In software reliability chain, input data which is not matched with the specified data cause the failure of the system. This error comes from the failure chain model. Operational profile define

the user, input space and occurrence probability of the user input cause the failure of the system. Reliability of the system that can be calculated by the no of errors identified at the time of execution.

1. INPUT TYPE:

If there is any conflict between expected output by the user and actual output, which cause the error, and make a chain of failure system. The input data reliability depends on the usage behavior of different type of users in the system. The same system is either used by homogeneous or heterogeneous users. So, the reliability factors which affect the system are, no of inputs, inputs space, input partition space, occurrence probability of the usage of the data.

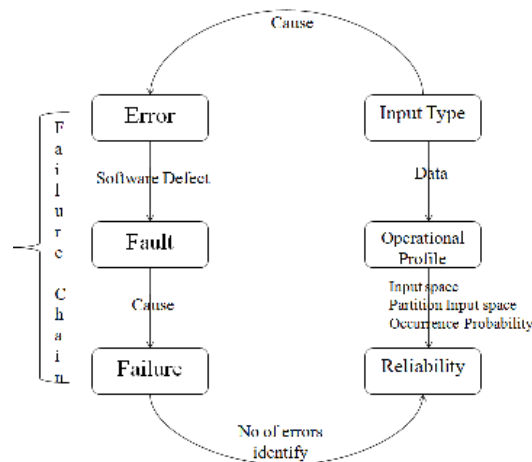


Figure 1: Software Reliability Chain

2. OPERATIONAL PROFILE:

Operational profile gives us information how to deploy the product we are establishing and can concentrate on development and test conditions [4]. An operational profile is a complete set of operations with their probabilities of occurrence. It is a quantitative characterization of how the software is used in different environment. We can also define as: “Operational profile is the set of input events that the software will receive during execution of the program, along with the probability that the events will occur at the time of execution”.

In most of the cases, system engineers will take care off the product for working with different users to execute the operational profiles [5]. All testers need to know how to develop and execute operational profile. Operational profile is used to increase test efficiency, performance analysis, so it is very cost effective and helping in managing software development and performance. There are basically five stages to develop operational profile:

- Identify operations
- Create operation list
- Update operation list
- Occurrence rates of the operations
- Identify the probability of the occurrence rates

3. RELIABILITY:

The main objective of software reliability engineering is to facilitate to engineer, manager or user of software learner to do more accurate decisions. Reliability depends on the user rather than the developer because it relates to the operation of the program and cause of usage it is known as dynamic rather than static. The standard definition of “reliability” for software is the probability that a system or a capability of a system will continue to function without failure for a specified period in a specified environment [6]. If the system is failure means that the system is not met the requirement of the users in some manner.

B. FAILURE CHAIN:

Failure depends on the behavior of program. If the program is not responding as given input and expected output is not agreed. Then the behavior of the system is verified and identifies the cause of the failure. Faults are the defect in the program, when executed under particular conditions causes the failure. There are many consequences for the failure of a system, some of them are harmful and other is disastrous. Failure chain is defined in figure 1 with software reliability chain method.

- **Error:** It refers to a wrong or missing action in certain fault being injected into software. It also includes error sources like an engineer may make an error in defining the requirements resulting a fault in the code, which cause the results, failure of the system when executed under certain conditions.
- **Fault:** A fault is a structural change in a software system that may lead to the system's eventually leading. It also refers as an underlying condition within software that causes the failure.
- **Failure:** Every failure have various impacts on the software. In project every failure typically assigns class to distinguish them with each other. Failure intensity is an alternative way of expressing reliability. It is defined as failure per unit time.

There are many causes of failure of a system.

- Software failure
- Hardware failure
- Human error
- Security intrusion
- Usage failure

a) ERROR CHAIN PROCESS:

If there is a variance between an observed data, computed data, specified data or defined data, correct value or condition. It is used to identify the arbitrary stage between faults and failure [7]. Software faults, failure is caused of improper input by user or some other devices of the system or by failing equipment, power failure and hardware failure of the system.

Error has two different significances. First, a discrepancy between observed, measured, computed value or condition and the specified, theoretically correct or true conditions. Second, human input action, cause fault. Figure 2 shows that there are basically three stages of error chain to process the errors and identify the faults with the fault chain method.

(i) Error Process: It is the first step to recover the error. Error process is to remove the error from the software. To remove the errors

(ii) Error Recovery: It is the next step to recover the error. Error recovery can be achieved by either forward or backward error recovery.

(iii) Forward Error Recovery: It involves correcting the error without repairing to revoking previous operations. It is done through identify a secure position of the system. Flight control is the best example for forward error recovery. It involves achieving a safe state and terminating processing and can be implemented in a passive or an active manner.

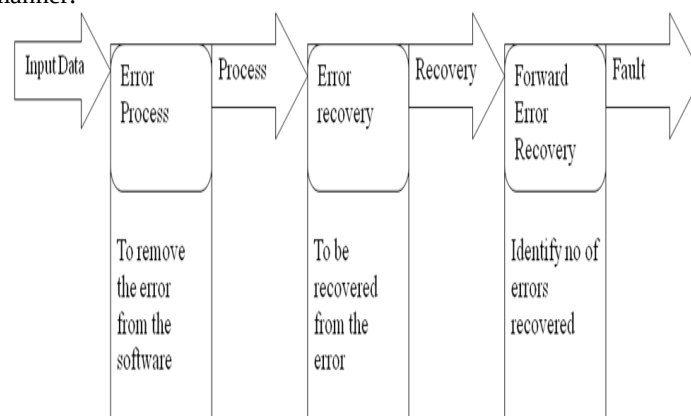


Figure 2: Error Chain Process

b) FAULT CHAIN PROCESS:

It refers to the series of component failures that extended up to the user-visible failures. Gray [8] found that 60% of failure has fault chain of length two. Oppenheimer et al. [9] found new failures with a failure chain of three or more in their study of failure in internet services. Amir and Wool discuss [10] discuss that hardware failures occur in large scale internet systems. A fault in software is an incorrect process, inability of a system or component to perform its required functions within specified performance requirements or data which cause a failure of a system.

The origin of failure comes from fault. Fault is stored in a static form in a program which occurs due to failure of certain conditions. Fault in the software is occurred due to error in program while an error occurs due to human action or nature of input. In figure 3 there are five stages of the fault chain method to predict, identify, prevent, removal and tolerate the faults.

(i) Fault Prediction: It is the first step to evaluate no of faults and the probability of occurrence of failure. Fault prediction is to formulate the relationship between failure data and the operational profile [5]. At the initial level prediction of software errors and failure are decreased.

(ii) Fault Identification: It is the next phase after prediction. Error and failure is used to identify the number of faults in the software. There are several methods to identify the fault from the software.

(iii) Fault Prevention: It is implemented at the design level, to make software demonstrated fault-free. In figure 2, if faults cannot be identify, then it can be prevent from faults in the software by using good software, good writing code, understand the function and user friendly of the software [11][12].

(iv) Fault Removal: The aim of this phase is, to remove the faults after the development of the software by verification & validation. Exhaustive testing is applied and removed all the faults from the software.

(v)

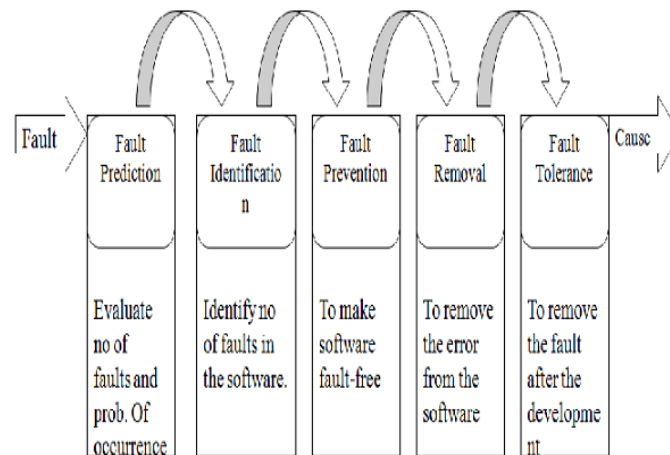


Figure 3: Fault Chain Process

(v) Fault Tolerance: It deals with all the faults which are stay after the system is developed and established. It is the last step to identify the reliable software. Fault tolerant techniques have two different groups, single version and multi-version techniques [13][14]. Fault-tolerance that has been dealt with critical applications like aerospace system, railway systems and nuclear power plant.

III. CONCLUSION

Today, people depend on computer which increases the problems of software reliability. Several happenings have been occurred from last three four decades regarding the failure of the system. System is failure when faults occurred. Every fault has a failure which activate at the time of execution. Failure rate increases if the fault function occurrence rate is increased. So, the reliability of the software depends on the fault, failure and operational profile. An operational profile if the set of operations that can execute in a specified period of time in a specific environment, probability of occurrence.

There are various causes of a failure of a system. Failure cause due to some discrepancy in the program. If the behavior of the software is changed, which can estimate as the software discrepancy in between obtained value to actual output. So, the error, fault and failure make a chain, if there is any wrong input given by the user. The above chain model is to facilitate to trace the error, fault and failure occurs in the model. Proposed chain is to identify the stage of the error or fault.

IV. FUTURE SCOPE

From the above it is found that fault detection, fault prevention, fault identification, fault removal and fault tolerance is one of the procedures to derogate faults in the reliable software development. Find the stages of failure or fault in development automata or Markov chain model are used to estimate the fault. It also help in identify the probability of failure occurred in future.

V. REFERENCES

- [1] J. Pan, "Software Reliability", Carnegal Mellon University, Dependable Embedded Systems, 1999, pages 1-15.
- [2] K. S. Trivedi, "SHARPE 202: Symbolic Hierarchical Automated Reliability and Performance Evaluator", in Proceedings International Conference on Dependable Systems and Networks, 2002.
- [3] S. R. Dalal, M.R. Lyu, C. L. Mallows, "Software Reliability", Bellcore Luscent Technologies, AT&T Research.
- [4] R. A Khan, K. Mustafa and S.I Ahson, "Operational Profile-Factor for Reliability Estimation", University Press, Gautam Das & V P Gulati, CIT, 2004, pp 347-354.
- [5] P. K. Chaurasia and Jahid Ali, "Operational Profile: A Critical Review", Journal of Global Research in Computer Science, JGRCS, Vol. 4 No 8, August 2013, pp 57-61.

- [6] J. D. Musa, A. Iannino and K. Okumoto, (1987), "Software Reliability: measurement, prediction, application", New York: McGraw Hill.
- [7] C. William, C. McCarthy, C J. Cook, "Predicting Reliability Improvements", IEEE Power & Energy Magazine, 2008, pp. 53-60.
- [8] Giridharan Vijayaraghavan, "A Taxonomy of E-Commerce risks and Failures", Masters thesis, Florida Institute of Technology, July 2003.
- [9] Oppenheimer et al, "Failure of penicillin in a newborn with congenital syphilis", JAMA 1970;212 135-9.
- [10] Y. Amir and A. Wool, "Evaluating quorum systems over the Internet", on Fault Tolerant computing, June 1996.
- [11] M. R Lyu, "Handbook on Software Reliability: Introduction", AT & T Bell Laboratories, 1996, pp. 1-42.
- [12] J. D. Musa, A. Iannio, and Okumoto, Software Reliability: Measurement, Prediction, Application, Professional Edition, Mc Graw Hill Publishers, New York 1990.
- [13] J.D. Musa, The Operational Profile in Software Reliability Engineering: An Overview in Proceeding proceedings of the 3rd International Conference on Symposium on Software Reliability Engineering, Research Triangle Park, NC, pp. 140-154, 1992.
- [14] J.D. Musa, Operational Profile in Software Reliability Engineering, IEEE software, vol. 10, no2, pp 14-32, 1993.