IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 10-14
www.iosrjournals.org

An overview and comparison of Software Reliability tools

[1]Dileep Sadhankar[1], Dr. Ashish Sasankar[2]
Department, of computer Science, G.H. Raisoni Institute of Information Technology, Nagpur, India

Abstract: *Software Reliability has been regarded as one of the important quality attribute and still there were very few standardized tool had been developed to estimate software reliability in quantitative terms. Software reliability is such a critical factor in deciding the overall quality of software, thus it should be estimated. Here we have surveyed many existing software reliability tools like SMERFS (Software Modeling & Estimation of Reliability Functions for Software), SRMP (Statistical Modeling & Reliability Program), SoftRel, SoRel, SRTPro, MEADeP and SREPT. In order to improve software reliability, the degree of current software reliability needs to be measured and managed by using an automated evaluation tool in a convenient way throughout the whole SDLC (Software Development Life Cycle). In this paper we highlighted few software reliability tools used in software development process or particular phase of life cycle and made comparison on the basis of few characteristics.*

Keywords –*Availability, Data analysis, Evaluation, Markov chain, Parameter estimation, Reliability, Software tool.*

I. INTRODUCTION

People think that software is always correct and once it runs correctly it will be correct forever this gave rise to the concept of Software Reliability. According to ANSI, "Software Reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment" [1]. Therefore we can say that reliability is a critical factor in deciding the overall quality of software. Therefore researchers move towards the estimation of Software Reliability which then gave rise to the various software reliability estimation tools.

Several software reliability tools are available for users to apply one or more of the software reliability model to a development effort and to determine the applicability of a particular model to a set of failure data.A major issue in modeling software reliability lies in the ease-of-use of currently available tools [12]. Nearly all tools have command-line interfaces, and do not take advantage of the high-resolution displays that would allow the construction of menu-driven or direct-manipulation user interfaces. The variety of results that may be graphically displayed is usually limited to interfailure times or failure intensities. In measuring software reliability, it is useful to see high-resolution displays of these quantities, as well as cumulative number of errors, the reliability growth curve, and the results of statistical methods used to determine whether the model being executed is appropriate for the current project[2,3].Following tasks are handled by the SRE tools:

- Collecting failure and test time information
- Calculating estimates of model parameters using information available.
- Testing to fit a model against the collected information.
- Selecting a model to make predictions of remaining faults, time to test, etc.
- Applying the model

II. BACKGROUND THEORY

Software reliability has become a buzzword for quitesome time, still there were very few standardized tool had beendeveloped to estimate software reliability in quantitative terms.Software reliability is such a critical factor in deciding the overallquality of software, thus it should be estimated.

Researchers developed various tools for estimating "Software Reliability". These tools give the detail analysis of reliability in such a way that it can be further utilized to improve the reliability criteria for real-time applications [4]. Any Software Reliability Estimation Tool involves 5 main steps:

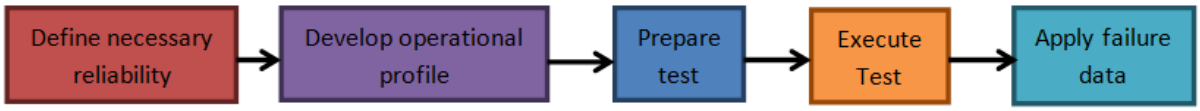

Figure 1: Steps Involved in Reliability Estimation Tool [4]

III. SOFTWARE TOOLS

1. CASRE (COMPUTER-AIDED SOFTWARE RELIABILITY ESTIMATION TOOL)

CASRE stands for Computer Aided Software Reliability Estimation. CASRE is implemented as a software reliability modeling tool that addresses the ease-of-use issue and other issues. CASRE is an extension of the public-domain tool SMERFS, and is intended to execute both in a DOS Windows environment and a UNIX -windows environment. Users are guided through the selecting of a set of failure data and executing a mode. Modeling results are presented in a graphical and tabular format. After one or more models have been executed, the predicted failure intensities or interfailure times are drawn in a graphical display window. User can manipulate window's controls to display the results in a variety of ways, including cumulative number of failures and the reliability growth curve[4,5].

In addition, CASRE is facilitated with a useful functionality. Namely, results from different models can be combined in various ways to yield reliability estimates whose predictive quality is better than the individual models themselves. CASRE isable toincrease predictionaccuracy bycombining the results of several models in a linear fashion and allows users to define their own combinations and record them as part of the tool's configuration. Weights for the components of the combination maybe static or dynamic, and may be based on statistical techniques used to determine the applicability of a model to a set of failure data. Once combination models have been defined, the steps required to execute them are no different than executing a simplemodel[6].

2. SMERFS (STATISTICAL MODELING AND ESTIMATION OF RELIABILITY FUNCTIONS FOR SOFTWARE)

The software reliability prediction tool is SMERFS(Statistical Modeling and Estimation of ReliabilityFunctions for Software) [Farr93], a well-known andwidely accepted software application for evaluation oftest data for failure rate and defect discovery rateprediction. The version of SMERFS used in this studyincluded a total of 15 different reliability growthmodels. The input to SMERFS is a set of valuesconsisting either of the time between discoveries ofdefects or the number of defects discovered per timeperiod. SMERFS then uses maximum likelihoodmethods or least squares methods to estimate theparameters used for one or more of these models(depending on the type of input and user selectedoptions). Its outputincludes the parameter estimates, predicted values and a measure of the goodness-of-fitusing the chi-squared distribution [4,7].

3. SOFTREL

The software reliability process simulator SoftRel captures the effects of interrelationships among activities, and characterizes all events as piecewise-Poisson Markov processes with explicitly defined event rate functions.

SoftRel simulates two types of failure events, namely, defects in specification documents and faults in code,all considered to be in the same seriousness category, as reflected by the single set of "model" parameters.The documentation' simulated by SoftRel consists only of requirements, design, interface specifications, and other entities whose absence or defective nature can root faults into subsequently produced code.Integration and test procedures, management plans, and other ancillary documentation, when deemed not to correlate directly with fault generation, are excluded. The presumption is that the likelihood of a fault at any given time increases proportionately to the amount of documentation missing or in error [4].

Requirements analysis and design activities are currently combined in the document construction and integration phases in SoftRel. All defects occur either in proportion to the amount of new and reused documentation, to the amount that was changed, deleted, and added, or to the number of defects that were reworked. Following are the characteristics of SoftRel[8]:

- Console-based application written in C (about 1300 lines of code)
- Source code is available
- One input project file (formatted text)
- Generates one output file (CSV)

4. SRMP (STATISTICAL MODELING AND RELIABILITY PROGRAM)

The SRMP was developed by the Reliability and Statistical Consultants, Limited of UK in 1988. SRMP is a command-line-oriented tool developed for an IBM PC/AT and also UNIX based workstations. SRMP contains nine models. SRMP uses the maximum likelihood estimation technique to compute the model parameters, and provides the following reliability indicators:

- Reliability function
- Failure rate
- Mean time to failure
- Median time to failure, and

- The model parameters for each model.

SRMP requires an ASCII data file as an input. The file contains the name (or other identification of the project, the number of failures involved in the reliability analysis, and the inter failure times of all the failures. The input file also specifies the initial sample size used by SRMP for the initial fitting of each reliability model to the data. The remaining failures are used by SRMP for accessing a reliability model's prediction accuracy [4].

5. MEADEP (MEASURE AND DEPENDABILITY)

MEADEP is a failure data based dependability analysis and modeling tool. Dependability measures generated by MEADEP are either directly obtained from data, such as failure rate and event distribution, or evaluated by combined use of failure data and dependability models, such as system level reliability and availability. Thus, two basic types of input to MEADEP are:

- Data — structured failure reports containing information on failure time, location, type, impact and other failurecharacteristics
- Models — graphical specifications of dependability models including serial and parallel reliability blocks (with the exponential or Weibull distribution), weighted blocks, k-out-of-n blocks, and Markov reward chains.

The output of MEADEP consists of results obtained from data and results evaluated from models. Results from data include various graphical representations (pie charts and curves) of event distribution, Mean Time Between Events (MTBE) and its confidence interval, histogramsfor Time Between Events (TBE) and for Time To Recovery (TTR) distributions, with super-plotting of typical analytical functions accompanied by the results of their goodness-of-fit tests, and the mean, lower and upper bounds for failure rate, recovery rate, and coverage. Results obtained from models include: Mean Time Between Failures (MTBF) for repairable and non-repairable systems, steady-state availability, and reliability for a given time point, and interval reliability for a time period (average reliability over the period) [4,9].

6. SOREL (SOFTWARE RELIABILITY ANALYSIS AND PREDICTION)

SoRel is a tool for Software Reliability analysis and prediction. It is composed of two parts allowing reliability growth tests and application of reliability growth models. It allows two kinds of failure data processing (inter-failure data and number of failures per unit of time, i.e. failure intensity data). The main measures that can be evaluated are: the mean time to next failure (or MTTF), the intensity function, the cumulative number of failures and the residual failure rate of the software.

Four reliability growth tests are available: the arithmetical mean (for inter-failure time's data), the Laplace test, the Kendall test and the Spearmann test (for both inter-failure and failure intensity data). These tests allow periods of time during which reliability is increasing or decreasing to be identified in order to apply reliability growth models on data exhibiting trends in accordance with their assumptions (which generally leads to better predictions)[12].

Four reliability growth models are implemented allowing different kinds of behavior to be modeled:

- A decreasing failure rate tending to zero when time tends to infinity: the exponential model (Goel-Okumoto) which is an NHPP and the Littlewood-Verrall model which is a failure rate model,
- A decreasing failure rate tending to a non-zero value: the hyper exponential model (Kanoun-Laprie) which is an NHPP model
- An increasing failure rate followed by a decreasing failure rate: S-Shaped model (Yamada et al.).

The results are available into two forms:

Immediately on the screen (numerical results and curves), in the form of files which can be read and used by other Macintosh applications (Excel, Word…), numerical results and curves can thus be used directly for publications and reports.

The program is modular and new reliability growth tests and models can easily be added. It is written in Pascal and has the advantages of multiple window management since it runs in a Macintosh environment.The human/machine interface has been denoted special attention [4, 10].

7. *SOFTWARE RELIABILITY ESTIMATION AND PREDICTION TOOL* (SREPT)

There is an increasing need for a tool that can be used to track the quality of a software product during the software development process, right from the architectural phase all the way up to the operational phase of the software. SREPT offers several techniques that can be used at various stages in the software life-cycle. [11] Thus it makes it possible to monitor the quality of the entire software development process under a unified

framework for software reliability estimation and prediction. SREPT combines the capabilities of the existing tools in a unified framework [12]. In addition, it offers the following features-

1. Provides a means of incorporating test coverage into finite failure NHPP (Non-Homogeneous Poisson Process) reliability growth models, thus improving the quality of estimation.
2. It offers a prediction system to take into account finite fault removal times as opposed to the conventional software reliability growth models which assume instantaneous and perfect fault removal. The user can specify the fault removal rate which will reflect the scheduling and resource allocation decisions.
3. It incorporates techniques for architecture-based reliability and performance prediction.

8. SRTPRO

SRTpro (Software Reliability Tool professional) was developed as a software reliability measurement tool during software development life cycle for software reliability engineers. SRTpro provides software prediction models and estimation models in order to deal with software reliability at early phases such as requirement, design, and code and subsequent phases such as software testing phases. SRTpro incorporates the mathematical modeling capabilities of the public domain tool SMERFS (Statistical Modeling and Estimation of Reliability Functions for Software), and runs in a Microsoft Windows environment [13].

IV. COMPARISON OF SOFTWARE RELIABILITY TOOLS:

Factors			CASRE	SMERFS	SOFTREL	MEADEP	SRMP	SOREL	SREPT	SRTPRO
Language			FORTRAN	FORTRAN	C	VC++	FORTRAN	PASCAL	JAVA	C#
Performance & usability	Reliability or failure rate		Y	Y	Y	Y	Y	Y	Y	Y
	Total failure		Y	Y	Y	N	Y	Y	Y	Y
	Remaining failure		Y	Y	Y	N	Y	Y	Y	Y
Number of models supported			16	12	2	1	9	4	1	14
Available models for	Estimation		Y	Y	Y	Y	Y	Y	Y	Y
		prediction	N	Y	Y	Y	Y	Y	Y	Y
Graphics			N	N	N	Y	N	N	Y	Y
User assistance			Y	Y	Y	Y	Y	Y	Y	Y
Used in phase			TESTING	TESTING	EARLY DEVELOPMENT, TESTING & OPERATIONAL MANAGEMENT	SDLC	TESTING	TESTING	SDLC	SDLC

V. CONCLUSION

We highlighted several software reliability tools currently available in the market. For each tool we briefly describe its functionality, capability and user interface.**CASRE tool uses** set of failure data and executing a model. The predicted failure intensities or interfailure times are drawn in a graphical display window. User can manipulate window's controls to display the results in a variety of ways. **SMERFS** a well-known and widely accepted software application for evaluation of test data for failure rate and defect discovery rate prediction. It uses total of 16 different reliability growth models. The input to SMERFS is a set of values consisting either of the time between discoveries of defects or the number of defects discovered per time period. **SoftRel** captures the effects of interrelationships among activities. It simulates two types of failure events, namely, defects in specification documents and faults in code. All defects occur either in proportion to the amount of new and reused documentation, to the amount that was changed, deleted, and added, or to the number of defects that were reworked. **SRMP** uses estimation technique to compute the model parameters, and provides reliability function, failure rate, mean time to failure and median time to failure.**MEADEP** is a user-friendly dependability evaluation tool for measurement-based analysis of critical systems. Use of the tool on failure data from measurements provides quantitative evaluations of dependability for the target system, while greatly reducing requirements for specialized skills in data analysis and system modeling from the user. **Sorel** (Software Reliability analysis and evaluation tool) implements a global method for reliability follow up and evaluation in presence of reliability growth due to design fault removal. **SREPT** is the high-level design of a tool offering a unified framework for software reliability estimation that can be used to assist in evaluating the quality of the overall software development process right from the architectural phase all the way up to the operational phase and the incorporation of several techniques in a systematic, user-friendly form in a GUI-based environment.**SRTpro** (Software Reliability Tool professional) which can predict and estimate software reliability throughout the whole SDLC in a convenient way. "SRTpro", used to iteratively manage software

reliability and deal withlimitations. SRTpro canprovide user convenience and project management forassessing and improving software reliability.

REFERENCES

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

[2]. Lyu, M. and A. Nikora (1992a, July). Casre - a computer-aided software reliability estimation tool. In Proc. of the Fifth Internation al Workshop on Computer-Aided Software Engineering, Montreal, Que., pp. 264–275. IEEE Computer Society

[3]. M.R. Lyu, Handbook of Software Reliability Engineering, IEEE Computer Society, p.71-117, 1996

[4]. B.H.Far, Software Reliability &Quality[Online]. Available:http://people.ucalgary.ca/~far/Lectures/SENG521/PDF/ SENG521-12_handout.pdf

[5]. Allen Nikora, CASRE–A Computer-Aided Software Reliability Estimation Tool, http://www.openchannelfoundation.org/projects/CASRE_3.0.

[6]. CASRE, International Institution for Technological Research and Development Volume 1, Issue 1, 2015 A Survey on Refinement of Reliability by Improving Reliability Estimation Tool, MohitSurati ,BhaveshTanawala , P.B.Swadas

[7]. William Farr, Oliver Smith, SMERFS – Statistical Modeling and Estimation of Reliability Functions for Systems, http://www.slingcode.com/smerfs/drfarr.php, 1996.

[8]. **SoftRel**, REVIEW OF QUANTITATIVE SOFTWARE RELIABILITY METHODS (Tsong-Lun Chu, MengYue, Gerardo Martinez-Guridi, and John Lehner), Brookhaven National Laboratory Letter Report, Digital System Software PRA,JCN N-6725,September 2010

[9]. IEEE Transactions on Reliability, MEADEP — A Dependability Evaluation Tool for Engineers (Dong Tang, Myron Hecht, Jeffrey Miller and JadyHandal)

[10]. **SoRel:** A tool for reliability growth analysis and prediction from statistical failure data (KaramaKanoun, Mohamed Kaˆaniche, Jean-Claude Laprie, Sylvain Metge) LAAS – CNRS 7, Avenue du Colonel Roche - 31077 Toulouse Cedex (France)

[11]. **SREPT**: Software Reliability Estimation and Prediction Tool? SrinivasanRamani, Swapna S. Gokhale, and Kishor S. Trivedi

[12]. Software reliability toolkit (a practical guide for understanding and implementing a program for system reliability)http://spiral.kaist.ac.kr/wp/research/reliability-tools/