Software Reliability Measurement: A SAS® Application

James A. McKenzie, SAS Institute Inc., Cary, NC

ABSTRACT

Information services is a growing industry in the expanding world economy. Computer technology and software continue to play a critical role in the information services industry. The reliance on software demands that its validation, verification, and reliability. This paper will demonstrate the uses of the SAS® System in the engineering and management of software development projects. Its focus is the use of the SAS System to create tracking databases, and generate software reliability measures in graphical and summary format.

INTRODUCTION

The beginning of the system test period is a critical phase of the software development process. It marks the completion of the system and the beginning of the verification and validation activities of the Quality Assurance (QA) Department. Although the goal of testing software is to reduce the number of software failures, testing can also generate new failures. New failures are created when the modified code contains new faults or when the new corrected code reveals faults hidden by the original fault. If there is a failure in the processes that create the software, the rapid repair of faults and replacement of the test image also generates new failures. The ability of the software to perform fundamental tasks without exposing new product or process failures is very important to the completion of the system test period.

This paper’s focus is on the use of the SAS System to track, analyze, and archive software failure information. This system will allow development managers, developers, quality assurance personnel, and marketing representatives easy access to reliability measures of the software products and development processes. The archival of failure information provides data for the exploration of other metrics and reliability models.

TESTING ENVIRONMENT

The various host groups at SAS Institute are responsible for providing host-dependent services to the portable supervisor and creating the test images. In addition to the development activities, the groups execute and review the output of the SAS System’s Development Test Library (DEVTESTS). The development and testing tasks must be completed before the two-week code migration cycle (porting cycle) is repeated. Although porting is typically bi-weekly, the cycle may be shortened to twenty-four hours as the scheduled release date approaches.

The DEVTESTS contains test cases to verify the portable products, core supervisor services, and host level services have been implemented correctly and provide basic functionality. During the early stages of the development cycle, the DEVTESTS also provide verification of the porting and building processes. The library is not limited to validation of basic functionality; there are test cases to validate more complex aspects of the software. Execution of the DEVTESTS usually begins after the software is “up” on the host. The software is considered “up” when the system can be brought to a line mode prompt and can successfully execute a series of host level test cases. Execution of the DEVTESTS begins early in the development cycle to allow the repair of major product and development process faults as soon as possible.

Once the software has been delivered to QA, the DEVTESTS are used to verify the absence of regressions. Regressions are software failures that are exposed after other faults have been repaired. The host group typically attempts to execute and review the DEVTESTS before a new image is delivered to QA. However, limited resources prevent both the execution and review of the complete test library. In an effort to gain a preliminary evaluation of the software, the Data General and Prime® host group investigated software reliability measurement (SRM). This type of software metric was chosen because it evaluates the software and the processes that create the software, and minimizes both testing man hours and computer resources.

The Reliability Test Library (RELTETS) consists of a subset of the DEVTESTS. Test cases from the DEVTESTS which use date, time functions, and system dependent information to create data are not included in the RELTESTS. A randomly selected sample of test cases that produce consistent output are used to measure the software’s reliability. This application defines a software product failure as the inability of the software to produce output that matches an established benchmark and defines a process failure as an incomplete image. Process failures occur when a part of the product is built incorrectly or is missing from the image.

The Mean Time Between Failure (MTBF) model is chosen for this application. This model is selected mainly because it can be integrated into the development processes without disrupting the existing process. Other benefits of SRM models include: focusing on software verification, providing a measure to compare the reliability products across hardware platforms, and providing primitive data for future reliability models. Also, the MTBF measure is easily understood; a larger value should reflect greater reliability.

SRM APPLICATION REQUIREMENTS

The application has three major requirements. First, the application must be easy to use, allowing frequent users quick access to testing information and providing new users a simple method to gain testing information. The application must automate the collection and reporting of testing information for both the DEVTESTS and RELTESTS. Finally, it must be flexible enough
to allow the exploration of other metrics and reliability models.

In addition to the application, a test driver and a file comparison utility are required. Test case information is read from a table file created by the SRM application. As each test executes, the driver will record test case information, current date and time, an image identification number, domain identifier and the amount of CPU time. After the test case has been executed, the output is compared to an established benchmark, and the number of differences will be recorded.

SAMPLE SESSION

The SRM application provides an interface into the test information data set using the SAS/AF® and SAS/FSP® products. The main menu allows you to select tests for execution, read audits from previous samples, enter failure classifications, generate reports of testing status, or display graphs of the reliability measures (see Figure 1.)

![Figure 1. Display of Main Menu of SRM application.](image1)

The first option from the main menu allows you to randomly select a sample from the RELTESTS. The sample is selected from the entire library or for a specific product. The selected test case names and associated system options are written to a table file in the current directory.

The second option displays a window allowing you to enter the pathname of the audit file. A new data set is created using the specified file. The new test results are sorted by table name, test name, domain, and date, and are then added to the master reliability data set.

The third option from the main menu allows you to enter failure classification information (see Figure 2.) All test cases not matching the established benchmark are required to have a failure class value. Failures are classified as either a product or process failure.

![Figure 2. Display of SAS/FSPEDIT® screen of the SRM application.](image2)

The fourth option allows you to generate summary reports of the testing status or display graphs of the MTBF model. When this option is selected, you must choose between a summary report or a graph. Summary reports are generated using the domain and image you specify. When the graphics report option is selected, you must choose from an existing SAS/GRAF® catalog containing charts of the MTBF model for each image (see Figure 3). This graphics catalog is updated nightly to reflect new information entered into the data set. An example of the MTBF graph is shown in Figure 4.

![Figure 3. Display of Catalog window for reliability graphs.](image3)

CONCLUSION

The SRM application allows the host group to efficiently evaluate the quality of the products and the porting and building procedures. The application tracks test case execution and failure information. The data collected can be applied to other reliability models and software metrics.
MTBF Results for Test Image 6.07.01.14NOV90P

Figure 4. Display of MTBF reliability graph.

REFERENCES


SAS, SAS/AF, SAS/FS, and SAS/GRAPH are registered trademarks of SAS Institute Inc., Cary, NC, USA

Prime is a registered trademark of Prime Computer, Inc.