

Building Quality into the SAS® System

Frank Lassiter, SAS Institute Inc., Cary, NC

ABSTRACT

Developing quality software systems requires building quality into each step of the software development process. This begins with customer driven design and development which is implemented by qualified personnel. Testing is initiated early in the development cycle and follows the software throughout the development life cycle. A formal validation process involving test planning, development, execution and documentation is required prior to release. After release, support needs to be provided in areas such as technical support, documentation and training. This talk describes the process of building quality into the SAS® System, with a focus on the testing and validation process.

THE QUALITY IMPERATIVE

The 1980's brought a call for quality to the U.S. business world which continues today. Industries and businesses from all sectors have heard the call to quality, and at this point, it is widely held that a commitment to quality is required to remain competitive in an evolving marketplace. Software development is no exception. At SAS Institute, development of quality products is a prerequisite to continued business survival. Users of the SAS System need to have confidence in the data maintained and analyses produced by their computer systems and they rely on the quality of the SAS System to meet this need.

BUILDING QUALITY INTO SOFTWARE SYSTEMS

In software development, it is recognized that quality is not simply added to the software product by testing at the end of development. Developing quality software systems requires building quality into each step of the software development process. The first step begins with customer-driven design. The software needs to be designed to address the various aspects of customer satisfaction such as:

- functionality
- reliability
- efficiency
- usability.

Design and implementation require qualified staff. Along with computer science skills, developers need product or subject area knowledge for effective

implementation of complex products. This knowledge may come from a combination of education, experience, and continued professional contact with the user community.

Rather than being a final step, testing should be initiated early in the development cycle and continued throughout the development life cycle to validate the successful implementation of those items required for customer satisfaction at each step. As with development, testing sophisticated software also requires highly skilled and knowledgeable staff with both product and testing skills. Release of a software product requires a formal validation process involving planning, development, execution and documentation of testing. Along with testing activities, appropriate source management controls need to be maintained throughout release testing and a reliable process for creation, packing and delivery of media is also required to ensure the quality built into the system is delivered to the customer. After delivery, support needs to be provided in areas such as technical support, documentation and training. Customer satisfaction needs to be the drive behind each step: design, coding, testing, and support of the products. In the remaining time I will describe this process of building quality into software systems as it is implemented for the SAS System, with a focus on the testing and validation process.

SOFTWARE DESIGN AND DEVELOPMENT

As noted earlier, the first component in the software development process is customer driven design. Product developers and development managers are responsible for the design and implementation of the software. For many product areas, developers are chosen for their computer science expertise. For the statistical and applications products, such as SAS/STAT, SAS/ETS and SAS/QC, developers are chosen for their statistical background and prior experience in their field. Developers hold advanced degrees, most of which are PhDs for statistical products.

Maintaining contact with the product related community helps determine the direction of new software. In order to keep abreast of developments in their respective areas, developers are involved in professional meetings and conferences, and are active in review, publishing, and serving on review boards of professional journals. Contacts are also

maintained with universities, where developers are involved in seminars and in some cases hold adjunct faculty positions. Customer comments are sought by staff members at the various meetings and conferences, along with site visits.

In addition to direct contact between developers and customers, customer input is obtained through the international and local user groups, Technical Support, Marketing, Education and the SASware Ballot. The Institute's Technical Support organization compiles and distributes the SASware Ballot annually in an effort to obtain customer input to the future directions of the SAS System.

Ballot items and other feedback are discussed and evaluated by the individual development managers and recommendations are made as to the timing of changes and enhancements. After direction has been established, initial design and programming is typically left to a small number of people, often a single developer and manager. Implementation of the design remains in the hands of those with the background in the various product areas.

Applications and core code are initially developed and tested on a network of Unix workstations and then ported to the various host platforms for integration with host dependent code and subsequent release testing. The Institute has developed internal coding standards to ensure consistency and efficiency in software development. Standards are defined for C programming, writing SAS Procedures, and implementation of host-level routines. In addition, many individual application groups have specialized coding standards to ensure consistency within their products.

Design and applicable prototypes are subject to review by developers supporting similar applications and development managers. Language reviews are also held to maintain similar syntax across many procedures and avoid poor choices in implementation. In addition to internal review, we often send experimental copies of software and preliminary documentation to outside individuals versed in the relevant subject areas. Certain product areas have also made use of usability sessions to provide a "hands-on" environment that allows the development staff to evaluate user interaction with new products and make any needed changes early in the development process. The process of developer research, customer input and interaction, internal review, and outside review brings a more mature and useful product to our customers.

DEVELOPMENT TESTING PROCESS

Development testing is an important part of the software development process. The staff involved in

this step are highly experienced and motivated. Generally, testers have as much applied experience as developers do. In statistical applications areas, testers hold advanced degrees in their fields of study. The development testers are chosen for their abilities to understand the methods being implemented, their ability to test, and their ability to provide feedback to the developer.

Testing is first performed by the developer writing the code, as the required subroutines are completed. Once past the initial tests by the developer, a development tester is assigned to the project and begins to help identify problems in the software. An initial period of informal testing takes place in an effort to find as many problems as quickly as possible.

As the product or procedure becomes stable, testing attention turns to the creation of test suites. A suite is a collection of reusable test cases that are written to test the new procedure. A test case is composed of a test program, usually a SAS program, with an associated set of benchmark output files. The benchmark output files contain correct test results which have been reviewed and approved by the product tester. These test suites serve as evidence that the procedure performs as documented, on both the development and production host machines. They will continue to be executed throughout the development and validation phases of release testing. In addition, they will be used to test future versions of the procedures in new releases of the SAS System. If individual test results from a new release match the established benchmarks, then the test passes. If not, the test fails, and the developer and tester work together to determine the reason for the failure.

Each test suite is developed to include a group of tests which systematically cover the different areas for each product or procedure. Typical categories of systematic testing include parsing, input data, error-handling, output data sets, missing data, printing, bug verification, and memory-testing. Examples of typical missing values conditions covered in test cases are shown in Table 1.

In the test case design, testers are also encouraged to write test programs that simulate typical user applications. This includes:

- the use of realistic data for test programs
- writing realistic applications as test programs
- concentrating on mixing global constructs of the SAS System into the test program.

In establishing the benchmarks for each test, validation of results is the most important testing task. Individual test cases are validated in one of several ways:

- comparison of results to known, published

- results
- hand calculations
- cross-validation by comparing results to those produced by other procedures that can be manipulated to produce the same results
- independent coding of the algorithm with our matrix language software.

The resulting tests and benchmarks make up a permanent library of testware, maintained in an environment which is protected by change control. Testers are challenged to continually review and update the test suites over time. This allows for incorporation of new ideas and feedback from past effort in test programs for the future. Currently, development test suites contain over 14,000 development tests which are run routinely on both the development host and production hosts.

Table 1. Typical Missing Value Test Conditions

- all missing values in dataset
- missing values at beginning of dataset
- missing values at end of dataset
- missing values at beginning and throughout dataset
- missing values scattered throughout dataset
- one or two variables completely missing
- missing values for BY and ID variables
- missing values for independent variables
- missing values for dependent variables
- overlapping missing values among analysis variables
- NOMISS and related missing data options
- missing CLASS or grouping variable values

SYSTEM VALIDATION AND RELEASE

As a part of its broader role of promoting the quality process at SAS Institute, the Quality Assurance (QA) department is ultimately responsible for the formal software validation process.

Quality Assurance is an autonomous organization, with final authority on declaring a product or release of production quality. The QA organization faces the challenges of validating both complex procedures and products as well as the integrated SAS System on multiple hosts simultaneously. QA is staffed to maintain product expertise across a growing list of products, each of which carries its own set of complexities. In addition to portable products, host-level routines must be exercised, interactive testing must be performed under various graphical user interfaces, and full-scale integration tests must be carried out on the production hosts to ensure correct behavior on each platform. This integration

testing involves testing of the many interactions and interdependencies of the various products in the SAS System, the host-level routines and the various operating systems.

In response to this complex testing task, QA is staffed with nearly 100 individuals, equipped with product knowledge, educational background, software quality training, and tools to complete the needed testing. As with development testing, QA staff are selected for their education and experience in the product areas in which they test. For statistical and applications products, many staff hold advanced degrees in their field of study. The QA staff holds primary responsibility for the System Validation process, with the support of product and host development testers, who initiate testing earlier in the software life cycle. The system validation process includes effective test planning, design, execution, and documentation along with problem reporting and resolution.

Test Planning

Test planning is carried out at various levels at the Institute. The Quality Assurance Master Test Plan provides the overall framework for QA testing for individual releases. In addition, group test plans are developed by the individual QA testing groups, and a Domain Test Plan is developed for each host environment detailing specifics of the platform on which the software will be tested. Finally, detailed Test Design Specifications (TDS) are developed for each product or procedure as appropriate. The TDS contains features to be tested, approaches to testing, and a systematic inventory of test cases to be tested, along with a prioritization of the test cases. TDS are prepared based on user documentation and input from developers, along with feedback from Technical Support and customer input. TDS review by peers and developers is employed to ensure quality planning of the test cases.

Test Design and Development

Development of test cases and associated benchmarks is carried out in a similar fashion as described earlier for development testing, including the systematic testing and validation of reported results. Coordination with development testers is maintained to ensure proper coverage of the procedures. Tests may consist of SAS programs that generate logs and lists for comparison against benchmark results, or they may consist of activity scripts that guide interactive, hands-on testing. Input data for tests may be generated by the test program itself, based on user applications, or research from relevant literature. Testers review user documentation as testware is developed, verifying

that intended functionality and operation has been both implemented and documented as planned.

In developing testware, we have to keep in mind the reality that testing every possible combination of input data, requested functions of each procedure, and the interactions with other portions of the SAS System is not practically possible. Likewise, exhaustive testing of each possible terminal emulator, printer or other output device is not feasible. Given this understanding, testers at the Institute must make choices about what test cases to cover. While attempts are made to anticipate many different situations, the goal is to focus the majority of our testing efforts on the most typical situations. What are the most frequent types of data sets used? What are the most common analyses performed by each procedure? What are the most common computing environments and devices? Testers rely on prior experience and knowledge, and continually seek information from sources such as Technical Support and Marketing staff to help answer these questions and direct the focus of testing.

Test Case Execution and Documentation

New release testing is required on each production host. The QA validation testing is performed in the customer environment on each host and is accomplished through the use of creative and sophisticated testing tools for batch testing along with hands-on, interactive testing. Testing tools available include a portable batch submission tool to run batch jobs on each production host, a batch driver to automate testing for interactive products, a differencing utility to compare new test results with established benchmarks, graphical user interface (GUI) tools to aid in review of test results and differences from benchmarks, and a database for storing test results.

Automated tools are used to run more than 52,000 batch tests. Test logs and output are compared against benchmarks and test results are reported in 'difference' reports. The test results are stored and tracked in QATRACK, a SAS data base application. As in development testing, QA analysts review the test results from failed tests and work with developers to determine the reason for the failure. The results of the resolution are stored in the QATRACK database.

Many tests of the SAS graphical user interface are automated through the use of an internal capture-playback tool. This tool enables testers to capture a sequence of interactive events and then replay those events at a later time, producing output which can be compared against established benchmarks. Even with available automation, many tasks require hands-on, interactive testing. Interactive testing tasks are outlined in the individual product

Test Design Specifications and tracked in a SAS data base application called QAPLAN. These tasks provide direction for QA analysts in testing areas for which batch testing is not applicable. As in development of batch tests, interactive testing is focused on user-oriented scenarios.

Problem Reporting and Resolution

Problems found in the software during the QA validation phase are reported and tracked in another SAS data base application, DEFECTS. This system serves as a central repository of all reported problems, providing capabilities of entry, prioritization, notification, tracking, and documentation of problem resolution. High priority problems include those that result in the loss or corruption of data; produce incorrect, unreliable, or misleading results; cause system failure; or depart significantly from product function specified in user documentation. Lower priority bugs may include non-functional cosmetic differences or problems for which there are convenient workarounds. Problems entered into DEFECTS are automatically forwarded to developers for resolution.

Throughout the development and validation process, strict control is maintained over all source code. The Institute's Development Source Management System provides a full complement of tools for management of changes to source code. In the validation phase, source changes are only allowed with approved change ID numbers assigned by QA staff.

After resolution of the reported problems, QA staff ensure appropriate resolution by performing fix verification, which is also recorded in the DEFECTS data base. Where possible, new test cases are added to existing test suites to cover high priority problems which have been fixed. These tests are then available to run against current and future releases to avoid recurrence of the problems.

Regression Testing, Final Inspection, and Delivery

After a first pass through the batch and interactive tests is complete and identified problems have been resolved, a final testing pass is made on each system to verify changes and ensure there are no regressions from earlier images of the system.

Customer package preparation is authorized by QA when the following conditions have been satisfied:

- regression testing has been completed
- the rate of newly reported problems has declined below an acceptable threshold
- newly reported problems are minimal and not of high priority

- all requested fixes have been implemented and tested
- items deferred to maintenance have been documented in the SAS Notes database which is shipped with every release of the SAS System.

As part of the package preparation, the Software Production Services (SPS) department forwards the final installation package to Quality Assurance for final acceptance testing and verification that the images on the customer media match the tested image. Software Production Services, in cooperation with the Software Distribution department, prepares shipments of production certified software to customer sites. Media creation, packing, and final quality checks on the integrity of the shipment itself are all performed by SAS Institute staff rather than outside suppliers to ensure that the Institute's quality imperative is maintained and controlled.

In addition to formal testing, informal testing of a new release is provided by all SAS software users at the Institute. SAS programmers in Management Information Systems(MIS), Technical Support, Marketing, Development testing and QA use the new releases in daily programming, applications and production work. In fact, 95% of all production systems at SAS Institute are built using the SAS System. Applications in Human Resources, accounts payable, payroll, financial management, and shipping and receiving all rely on the continued quality of new releases. Production versions of the SAS System are maintained by MIS on MVS, CMS, OpenVMS and HP-UX. New releases of SAS are installed using the production installation process on each of these hosts on or before the shipment of each new release. This provides the opportunity to check out installation procedures and the reliable and consistent performance of production applications from release to release of the SAS System.

SUPPORT SERVICES

The Institute provides documentation, education, technical support, Ambassador and consulting services to support the software product. Education, documentation and Ambassador services help train and equip users to effectively use the software products. Technical Support responds to user problems by phone and a variety of electronic methods. Once the production software has shipped, the Institute's Technical Support department serves as a clearing house for reporting problems involving software products in production. Problems found after the software is in production status are entered in the DEFECTS data base where they become items for development action and for QA in the never-ending process of refining and improving test coverage. You

have an opportunity to contribute to this process by sharing your test cases which reproduce reported problems. When available, test cases are stored in directories associated with reported problems and may be used by testers to augment existing test suites. You can provide test cases to Technical Support electronically via the Online Customer Support Facility(OCSF), Electronic Mail Interface to Technical Support(EMITS) or anonymous FTP. Please see Technical Support document TS-301 for further details. The SAS Notes data base, documenting known issues, is also updated on a continuous basis and made available to customers on all product installation tapes. In addition, important problems are addressed in periodic maintenance releases.

SUMMARY

As you can see, our goal is to build quality into the SAS System at each step of the development process. We start with user driven design, developed and implemented by qualified staff and guided by input from the user community. Testing is in place throughout the development process, carried out by various groups including developers, development testing, and QA, and supplemented by production use in MIS. Test plans and results are documented for each release and user documentation is prepared and reviewed for agreement with intended functionality. Configuration management, change control and problem reporting tools are also a part of the validation process. After testing, the commitment to delivery of a quality product is continued in the production, delivery, and support services provided by the Institute.

In today's changing environment, SAS Institute continually seeks opportunities for quality improvement. Current products, processes, and customer services are reviewed for improvement opportunities. Today, I've provided an overview of the Institute's efforts in building quality into the SAS System. Each of the quality initiatives are undertaken to ensure that customer needs are understood and built into the SAS System.

ACKNOWLEDGMENTS

The author thanks Robert N. Rodriguez, Maura Stokes and Sue McGrath for technical review.

SAS, SAS/STAT, SAS/ETS, and SAS/QC are registered trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.