JMP Statistical Discovery LLC

JMP® Research and Development: A Process for Quality

Note: This statement is part of a validation package for JMP (available at jmp.com/qualitystatement). JMP Statistical Discovery LLC is a wholly-owned subsidiary of SAS Institute Inc. This document is an addendum to the SAS quality paper, "The Quality Imperative: SAS Institute's Commitment to Quality" (available at www.sas.com/qualitypaper), where the information about SAS personnel, privacy issues and work environment pertains equally to JMP.

JMP Quality Statement

The first version of JMP software was released in October 1989. For more than 30 years, JMP has continued to refine a research-centric process that is built on development partnerships with customers in key industry segments and leading researchers in universities and academic circles. JMP software is developed at the SAS world headquarters in Cary, North Carolina.

JMP Research and Development (R&D) shares the central mission, philosophy, and software development lifecycle that has been established by SAS. All the information in "The Quality Imperative" about SAS personnel, privacy issues, and work environment pertains equally to JMP.

Our Software

JMP dynamically links statistics with graphics, enabling our customers to interactively explore, understand, and visualize data. Designed for anyone who wants to discover relationships and patterns in their data, JMP provides a comprehensive environment for exploratory data analysis, statistical modeling, and experimental design (DOE) in a single package. Moreover, JMP is designed to work with other SAS software in the SAS Business Analytics Framework, providing an interface to submit SAS code and interpret SAS results. JMP can also read and create SAS data sets, text files, Excel files, and access many relational databases using Open Database Connectivity (ODBC).

The JMP Development Process

The foundations of the JMP development process are: frequent customer contact; prototype releases as code is being developed; and open, informal communication between customers and senior development staff in the JMP development and marketing teams. The JMP development process is innovative, iterative, and highly collaborative. Sales success and customer longevity in key industry segments is strong evidence of the success of this philosophy of development.

Staff, Training, and Skill Sets

The JMP Research and Development (R&D) team consists of the following operational units:

- Statistical Development
- Host and Portable Development

- Testing and Validation
- Documentation
- Source Management/Test Automation/Release Engineering
- Localization/Internationalization

John Sall, co-founder and Executive Vice President of SAS, and primary architect and designer for JMP, leads the JMP business unit and its research and development efforts. Sall is also responsible for programming many crucial platforms and routines in JMP software. A biographical summary about John Sall can be found at: jmp.com/en_us/bios/sall-john.html.

A majority of the statistical development, testing, and documentation staff members have a master's degree or higher in statistics or operations research. Almost half of these staff members also hold a Ph.D. in statistics.

A majority of the staff members in the host and portable code development areas have a master's degree or higher in computer science or related fields.

Software Development Tools

JMP software is written primarily in C++, using various development environments that are based on Microsoft Windows and Apple macOS®. Perforce software tools provide configuration and source code management.

The testing and validation staff automate testing for much of the JMP software functionality using a proprietary framework that is based in the JMP scripting language (JSL). These tests are run on Windows and macOS in 64-bit environments. Testers also use other UI-based automation tools to automate graphical user interface-based tests across the same operating systems. Problems are reported and tracked using the Atlassian Jira system. (See the "Problem Reporting and Resolution" section in "Research and Development Tools" in "The Quality Imperative").

In JMP R&D, a team of technical writers create the JMP documentation and Help system. The documentation is produced using industry standard publishing tools that result in both printed documentation and Help files that are created from a single set of source files.

Software installers for JMP are built by using industry standard tools. The installers for JMP on Windows are built using Reverna InstallShield. Installers for JMP on macOS are built using Packages from WhiteBox Software.

JMP software is translated into French, German, Japanese, Simplified Chinese, Korean, Spanish, and Italian. Translation is done either during the development process or immediately after a major version of JMP is released to production. Localization experts and outside translation vendors are both engaged in translating help resources.

Selected JMP documentation resources are translated for the production release of a major version. Additional translated resources for selected languages become available in maintenance updates to the major version.

Software Development Lifecycle for JMP

The process that is used to develop JMP software is modeled generally after the process that is used by the Research and Development Division at SAS, as described in "The Software Development Life Cycle" section of "The Quality Imperative". Additional process details that are specific to JMP software development are explained below.

JMP software follows a multi-tiered release strategy to provide a balance between new features, stability, and security updates.

- Major releases: Scheduled for release on an 18-month cycle. Introduces significant new features and functionalities based on long-term product roadmaps and customer feedback.
- Minor maintenance releases: Scheduled every four to six months as necessary. Addresses reported customer issues and bugs to ensure software stability and reliability. Occasionally, these releases may include minor feature enhancements based on user requests.
- Security branches: Scheduled every two months as necessary. Provides critical security fixes for issues that are identified in JMP software.

Requirements and Planning

Early requirements are gathered through several channels:

- Customer and development partner requests
- Marketing analysis and requests
- Technical support requests
- Development and testing staff discussion and recommendations

Development and marketing priorities are tracked informally on Atlassian Confluence collaboration software. Technical Support and other customer-facing staff also enter requirements into the Atlassian Jira system as suggestions from customers. These requirements are prioritized with other development work as development progresses on a new version.

Requirements and design evolve iteratively through a process of prototyping and pre-production beta releases to select customers and academic advisors across multiple industries and disciplines. Input is gathered through e-mail, and communication between early adopter customers and development team members is archived in technical support tracks. Problems with early adopter versions of the software are tracked through the Atlassian Jira system.

Major Release Development and Testing

Milestones and goal dates are reviewed throughout the release in weekly management meetings, and adjusted within the cycle based on code stability, quality, testing coverage, and additional product needs.

R&D management meets weekly throughout the development cycle to track internal development phases:

1. **Early Development:** During this phase, new features are implemented. New feature development can involve peer code reviews, interaction with customer development partners, or experts in a particular discipline. During peer code reviews, a developer selects one or two other developers to explain their code modification and get feedback on coding and implementation. Feature-related code changes are often followed by an e-mail notification to the development and management team with details about the implementation. Automated regression testing begins on existing features at this phase, and development testing begins on new features as they become functional.

2. **Feature Freeze:** When this phase is reached, new features have been implemented but have not been fully tested. Fixing bugs becomes the primary focus.

3. **Code Review:** After continued testing and bug fixing, the software is fairly stable and the code review phase is reached. At this phase, all new code changes are reviewed to ensure that the changes address the intended issue and do not introduce additional errors. The review is conducted by one or two peer developers. The peer reviewers are noted in code push reports as code is checked into the source management system.

4. **Code Approval:** In this phase the software release is imminent and code changes are made only with permission from an approval board. Approval by at least four of the eight board members is required before a code push. The approvers are named in the push report that is created when the code is checked into the source management system with comments that explain the code change. An e-mail is sent to the development and testing team explaining the change and push.

5. **Handoff to release engineering groups:** Final release engineering and package validation is completed before shipment to customers.

Pre-Production Feature and Usability Testing

The JMP research and development process relies heavily on customer data and on customer feedback as a mechanism for providing input regarding usability and feature validation within specific and unique customer environments.

Throughout the development cycle, as the software becomes stable and as new features are ready for user testing, pre-production releases are made available to internal users and external customers under non-disclosure agreement. These early adopters provide input to JMP research and development regarding usability, installation, feature implementation, and any other challenges that are presented by their environment.

Using the Atlassian Jira system, issues are evaluated and prioritized, and then either fixed within the remaining cycles of the release or deferred to a future version.

Testing and Validation

Throughout the development process, JMP testing and validation specialists continually test updated versions of the software. A specialized statistical team verifies statistical results from JMP. Results from JMP are compared to the SAS System and other sources to ensure the highest level of accuracy that is possible. Methodology for testing the JMP software's numerical accuracy is effectively the same as the SAS approach. See "Validating a Statistical Procedure" in Appendix 1 of "The Quality Imperative" for more information.

An automated process compiles and tests JMP software every week night. After the build process is completed, automated tests are run against an installed image. When this process is complete, user-ready installs are built and delivered to the JMP testing department. Testing results are published to a development website for investigation daily. Bugs are entered and if larger scale issues are discovered, they can prompt a rebuild of the system or another action.

To reduce the risk that daily code pushes could cause large scale issues with negative impact to nightly builds and unit tests, JMP employs an automated continuous integration, build, and test process. The continuous integration system polls JMP's source management system regularly for new code changes. When updates are detected, new JMP builds are generated and a small subset of JMP's automated tests are run to test critical application and platform functionality. If build or test problems are detected by this continuous system, they are reported immediately in the Atlassian Jira system so that they may be fixed in advance of the next nightly run.

Testing of new features is done manually and ad hoc in early development and tests are added to the automated frameworks as the code becomes more stable.

Automated regression testing and manual testing of new features begins when the development track for a major version is established and new feature development begins. JMP testing and development groups maintain a set of approximately 2 million programmatic tests that run in a proprietary framework on all currently-supported operating systems for a given release. These tests and the automated test framework are written by quality assurance engineers and developers using the JMP scripting language (JSL). The tests are run regularly across all releases under development and exercise code at the unit, system, integration, and regression levels. Tests are also written at the unit level in C++.

To ensure that important combinations of options and system testing considerations are developed efficiently, the experimental design features of JMP software itself are used for generating testing coverage matrices. There is an additional suite of performance tests for basic JMP functionality. This full suite of automated tests runs nightly on all operating systems that JMP supports. Additional testing procedures are run on a periodic basis to detect memory related issues and issues reported in the Atlassian Jira system.

As new features are developed, testers, writers, and programmers meet to review the functionality. As they are developed, new features are demonstrated weekly in meetings that involve Development, Documentation, Testing, Technical Support, Education, and Sales and Marketing. These demos include discussion and collaboration regarding coding and implementation for proposed changes that have wide impact. The demos also enable discussion and understanding of requirements by everyone on the JMP team.

Problems are reported and managed in the Atlassian Jira system. Automated reports categorize problems by priority, developer name, area of code, and version number where the problem was found. As fixes are made and verified, the current status is updated in the Atlassian Jira system. Verification of code fixes is done by the testing group using Atlassian Jira verification tools. Documentation regarding the verification of the fix is maintained in the Atlassian Jira system.

Release Completion and Package Validation

When development and testing of a new version of JMP are complete, a production release candidate version is delivered to the release engineering groups who then create release package candidates.

The JMP testing and validation team, development, technical support, and product management review the package candidates to ensure that all requirements are met. After sign off by all reviewers, under the direction of the SAS director of quality assurance, the software is released to production.

Minor Maintenance Releases

After a major version is available to customers, customer-reported bugs and issues are reported primarily through the JMP Technical Support Division. These issues are tracked in the Atlassian Jira system. When appropriate, fixes to these problems are made in the next major release track in the source management system. The approval board also evaluates fixes for inclusion in a scheduled maintenance release.

Occasionally, maintenance releases may include minor feature enhancements based on user requests.

All proposed code changes for maintenance releases are reviewed by the developer who coded the fix and the peer developer. Maintenance releases also include a Code Approval phase as described in the Major Release Development and Testing section above where fixes must be approved by the approval board. Fixes are tested before inclusion in the maintenance track, and automated tests are updated appropriately. JMP performs automated regression testing on the maintenance version throughout a maintenance cycle to reduce the risk of regressions occurring when fixes are made in the maintenance track. Maintenance releases occur every four to six months as needed to address customer reported bugs and issues.

Security Branches

This release stream focuses on delivering critical security updates to address vulnerabilities that are discovered in JMP software. These frequent releases ensure customers have the latest security patches to minimize potential risks. Security branches occur every two months as needed.

Technical Support

The JMP Technical Support Division supports JMP software. See Appendix 2: "Quality in SAS Technical Support" in "The Quality Imperative" for details.

Training

A comprehensive curriculum of JMP courses, prepared for novice to expert JMP users, is available through the JMP Education Division. Instructor-based training is provided in public courses at JMP regional training facilities across the United States or through JMP Live Web. On-site courses can also be provided at the customer's location. Developed and taught by JMP technical training specialists, these courses use the most current JMP software. Content lectures, software demonstrations, hands-on computer workshops, and a copy of course notes are included with each course.

The Training portion of the JMP website (jmp.com/training) provides detailed information about all facets of JMP training courses.

FDA-Regulated Industries and Compliance with CFR Title 21 Part 11

For purposes of U.S. Food and Drug Administration (FDA) validation, JMP should be considered a tool.

JMP customers who are FDA-regulated might be expected to validate the systems that they build with JMP. Because JMP is a tool, the user must show the FDA that JMP is being used correctly. See "Complying with United States Code of Federal Regulations, Title 21 Part 11" in Appendix 1: "Regulated Industry Issues" in "The Quality Imperative" for more information.

Customers can re-create analyses by saving JMP Scripting Language scripts with a JMP data table. Customers can then use JMP to explore and analyze. When the analysis is complete, customers can save the JMP script to the data table. The JMP report can have date and time information embedded. Customers should keep the data table with the JMP script attached and the JMP report in the source control system as a read-only file. If any changes are needed in the future, customers can use the source control system to check the data table out, make the changes, and check the file back in.

The FDA has adopted a SAS transport format as the standard for accepting and archiving data sets. Note that JMP software can read and write the transport format.

Because JMP provides interfaces to SAS software, all of the techniques that are described in Appendix 1: "Regulated Industry Issues" in "The Quality Imperative" are available to JMP customers who also have SAS installed.

Operational Qualification (OQ)

To facilitate customer validation, JMP provides both a set of baseline tests and also an extendable test framework that are written in the JMP scripting language (JSL). The tests are written to an external standard for validation of statistical software, provided by the National Institute of Standards and Technology (NIST). These tests are not designed to cover every statistical function in JMP. Customers are encouraged to use these tests and framework as a basis for their own validation work. The JMP NIST tests are available on the JMP website (jmp.com/qualitystatement) and described in a white paper that is also available on the JMP website: "Assessing the Numerical Accuracy of JMP".

Additional Assistance

Questions regarding the JMP quality initiatives and validation should be directed to qualitypaper@sas.com.