

## Quality Assurance: Best Practices in Clinical SAS® Programming

Parag Shiralkar

eClinical Solutions, a Division of Eliassen Group

### Abstract

SAS® programmers working on clinical reporting projects are often under constant pressure of meeting tight timelines, producing best quality SAS® code and of meeting needs of customers. As per regulatory guidelines, a typical clinical report or dataset generation using SAS® software is considered as software development. Moreover, since statistical reporting and clinical programming is a part of clinical trial processes, such processes are required to follow strict quality assurance guidelines. While SAS® programmers completely focus on getting best quality deliverable out in a timely manner, quality assurance needs often get lesser priorities or may get unclearly understood by clinical SAS® programming staff. Most of the quality assurance practices are often focused on 'process adherence'. Statistical programmers using SAS® software and working on clinical reporting tasks need to maintain adequate documentation for the processes they follow. Quality control strategy should be planned prevalently before starting any programming work. Adherence to standard operating procedures, maintenance of necessary audit trails, and necessary and sufficient documentation are key aspects of quality. This paper elaborates on best quality assurance practices which statistical programmers working in pharmaceutical industry are recommended to follow. These quality practices are directly referred from the regulatory guidance and are illustrated with examples in this manuscript.

### Introduction:

Programmers working in pharmaceutical industry perform various tasks including but not limited to data management programming, data validation and mapping, data analysis, report generations, and performing queries on data. Although programming staff does lot of clinical as well as statistical programming, because of de-facto use of SAS® software in statistical programming such programming staff is often referred as 'SAS® programmers'. Some of the tasks in which programmers are involved are more 'supplemental' tasks to support the data management function. These tasks include data validation, data queries, and edit check programming. On a biostatistics side, programmers are involved in more analysis oriented tasks like generation of statistical reports, and development of analysis datasets. Time spent by SAS® programmers on these assignments depends on various factors like complexity of tasks, quality expectations, and availability of time to complete the request. Even though in most of the cases programmers meet customer expectations by providing best quality results, they often tend to loose their focus on bigger picture in terms of maintaining process integrity. In following discussion, we will look into some of the core process documentation in clinical trial and see how the SAS® programming is related to those processes.

**Clinical Trial Core Documentation:**

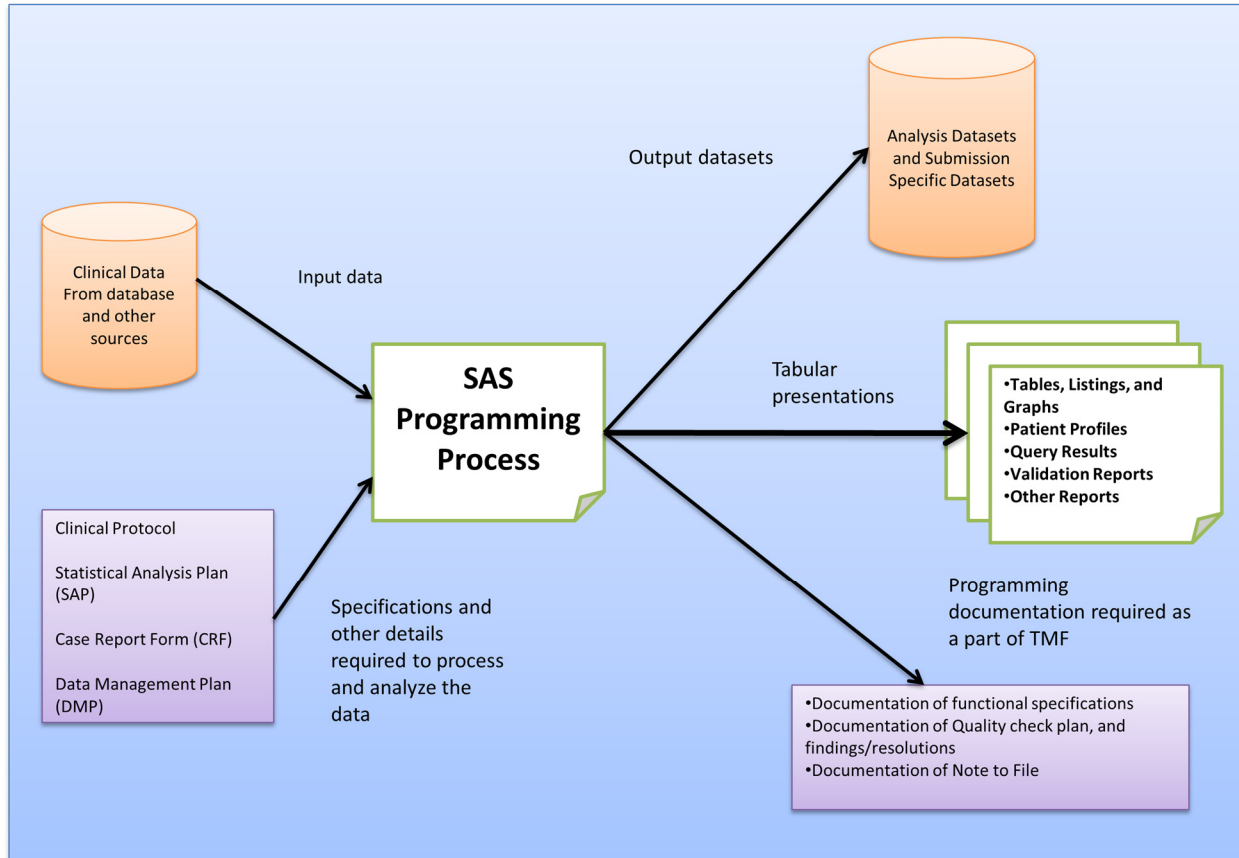
Every sponsor organization conducting or sponsoring a clinical trial is required to maintain a 'trial master file' or TMF. This core set of documentation consists of lot of trial specific regulatory documentation. This requirement is enforced by regulatory authorities to ensure that clinical trial sponsoring organizations follow regulatory guidance and good clinical practices (GCPs). In case of regulatory audit and submissions, TMF is the core focus of both internal and external assessments. TMF includes lot of core documentation of clinical trial. Since the work of SAS® programmers are primarily focused on data analysis and reporting, let's look into some of the core documentation related to data management and biostatistics functions which is part of TMF:

**Data Management Plan:** Data management plan or DMP consists of lot of details relevant to data collection, storage, archival, and overall data management. This includes the data standards used, data integrity checks and validation mechanisms used to manage the data. Programmers working on edit checks or data queries are often required to refer to data management plan for more input. At the same time, in many circumstances data management or clinical programming deliverables may get included as an input to data management plan.

**Statistical Analysis Plan:** This document often referred as SAP, elaborates on details of statistical analyses and reporting of the clinical trial data. SAP includes details including but not limited to analysis populations, windowing, imputations, baseline computations, and guidelines for other complex analyses. List of tables and table shells which outline the guidelines of tabular presentation needs for trial reporting are considered as a part of SAP. Any SAS® programs developed to produce these tabular presentations are required to follow the software development life cycle (SDLC) principals and related regulatory guidelines as listed in references. While following SDLC, programmers are required to document the detailed algorithm along with functional specifications. Many sponsors include such documentation along with quality check documentation as a part of trial master file.

So, looking at the above core documentation of clinical trial, it is evident that the scope of programming work SAS® programmers do in clinical trial is not confined to a micro level within biostatistics or data management functions. Rather, it is part of a broader requirement of trial level documentation and processes which get closely monitored and audited by regulatory authorities. While programmers focus on 'what needs' to be produced, there should be sufficient focus on documenting 'how it is produced and quality checked'. Although these two perspectives are 'required', how much focus should programmer give on these two aspects depends on management guidelines and the time and budget constraints imposed on the programming staff. Regardless of such constraints, it is strongly recommended that programmers pay necessary attention to documenting functional specifications and quality check related details as such details are part of trial master file of the clinical trial.

Overall clinical process where SAS® programming is a core focus is shown below:



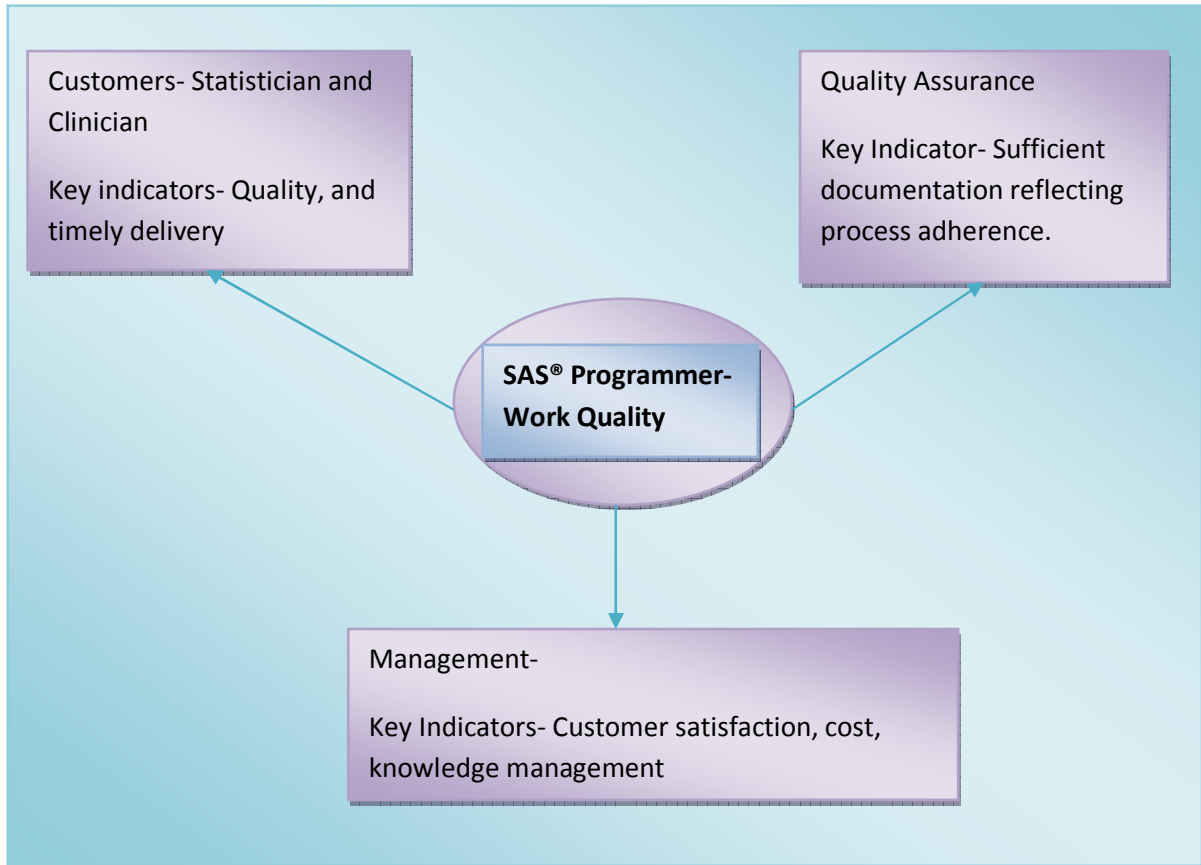
Clinical trial sponsor management defines the standard operating procedures and work instructions related to programming function by keeping above processes and regulatory guidance in considerations. While working on any assignment, SAS® programming staff is required to follow such processes, operating procedures and work instructions.

### Quality: How is it perceived?

Looking at above macro perspective, 'quality' of programming deliverable has multiple dimensions and performance of programming service is perceived by different stakeholders in clinical trial in different manner. Customers, management, and quality assurance are key stakeholders who get impacted with quality of programming deliverables.

- Customers: Depending on type of programming work programming staff is involved in, customers of programmers could be data managers, biostatisticians, medical writers, clinicians, or other analyst staff including pharmacokinetics and pharmacometric analysts. In most of the cases clinicians and statisticians are key customers of statistical programmers. In other words

clinicians and biostatisticians are primarily interested in the quality and timely delivery of reports and datasets produced by programmers. These customers perceive quality of programming service and staff by measuring how efficiently and in timely manner do programmers produce the outputs.



- **Management:** When it comes to quality of programming service, management perceives it from a macro perspective and considers customer satisfaction and cost effectiveness of the service. Quality of output and timely delivery as perceived by customers leads to customer satisfaction. However, cost effectiveness always puts a constraint in terms of how many programming resources can be utilized for certain tasks. Such budget constraints often have conflicting effect on programming function.
- **Quality Assurance:** While programming function and staff pays more attention to efficiency and cost, quality assurance needs often get lesser focus. In earlier section we looked at the overall clinical reporting process. The focus of Quality Assurance is to ensure that programming function and staff adheres to the process defined by management. This process adherence is often measured in terms of maintenance of process documentation and timely actions taken by programming staff over the course of trial.

### **Difference Between Quality Control (QC) and Quality Assurance (QA):**

Quality control of programming process focuses more on ensuring efficient output delivery and ensures satisfaction of key customers of statistical programming function. While ensuring efficient delivery, programmers may not necessarily follow the processes perfectly. Quality assurance on the other hand focuses on process adherence. If programming staff does not follow process completely, quality assurance expects that such instances are documented as process deviations. Although these two terms sound somewhat different they are inter-related and SAS® programmers are required to follow both quality control and quality assurance best practices. For clear distinction between QC and QA consider the following cases for illustration:

- Case when process followed QC but not QA:

SAS® programmer develops a SAS® code to produce a table based on the table shells outlined by biostatistician. After delivery of this table, biostatistician requests change in the table shell and asks the programmer to modify the code, and re-deliver the table based on modified table shell. Programmer follows these instructions, quality checks the table and submits it to biostatistician. In this process because of time pressure, programmer does not document the modified algorithm in the functional specifications document. The delivered table is of best quality and biostatistician is satisfied with the quality of programmer's work. This is an example of the instance when programmer is doing quality check but does not follow the process and do not meet the quality assurance guidelines.

- Case when process followed QA but not QC:

In one of the listing delivered to biostatistician, it is identified that the sort order of records is incorrect. After this is being noticed by biostatistician, primary programmer found that there is a section of code where the sorting of data isn't carried out as expected. Programmer fixes this issue and re-delivers the listing to biostatistician. While doing these fixes, primary programmer takes more time than planned to deliver the listing to biostatistician. Programmer documents this finding, records the date of identification of this issue, states the resolution and records the date when the issue is resolved. After this, programmer follows up with management and QC programmer and discusses this issue for future prevention of errors. In this case, although programming followed the quality assurance process and procedure, there is customer dissatisfaction as delivery of this listing to biostatistician was erratic and did not meet delivery timelines.

### **Planning QC Strategy: Within Framework of QA**

Based on above cases , it is clear that although programmers need to focus on quality control, such quality control activities should be within framework of the processes laid out and approved by management and quality assurance. In general if programming staff is being assigned a routine

responsibility or a milestone based deliverables, then it is strongly recommended that programmers develop a QC plan and clearly outline quality control expectations. Such QC plan should adhere to the existing SOPs and programming processes. QC plan should provide more details about what will be quality checked. This should include details about methods followed for quality check. As an example, certain checks can be done manually/visually by looking at reports, other documentation, and data. On the other hand, certain checks need to be done by doing data query or verifying the algorithm programmatically. QC plan should provide adequate details about QC methods. It should also provide clear guidelines about how QC findings will be reported, to whom will those be reported, and how the resolutions and actions taken will be documented. Such QC plan should be approved by the stakeholders involved in the process. As an example, for all the reports produced by programmers for the clinical study report deliverable, the QC plan should be mutually approved by biostatistician, primary programmer working on the trial, and by the QC programmer working on the trial. If needed, for certain aspects data manager should be consulted for data validation and QC. Such mutually approved QC plan needs to be followed by programming staff for rest of the course of QC activity. Management needs to take pro-active role in enforcement of use of such QC plan during the course of programming activities. While following the QC plan, programmers always face circumstances where it is difficult to judge deviation from process adherence. Following are some of the guidelines which programmers can follow to ensure QA process adherence:

- What to document?: In pharmaceutical industry, it is implied that 'if it is never documented, it didn't happen'! While considering need of documentation in programming process, it is important that all programming functional specifications as derived from SAP needs to be documented with necessary and sufficient details. Any changes in programming specifications as requested by biostatistician and other stakeholders need to be documented as a part of functional specifications. As per 21 CFR part-11, all major revisions in SAS® code as well as changes in functional specifications are required to be documented. Similarly, it is important to document QC findings, dates, and resolutions. All approvals must be signed off, and should be appropriately dated.
- How much to document?: If the documentation is so much important, then obviously next immediate question that comes in mind is how much to document? From QA perspective there is some 'necessary' set of documentation which includes documentation of programming functional specifications, and decision logs. Besides that any important deviations and changes need to be documented as a part of 'note to file'. All approval forms should be documented in a timely manner with necessary details. Besides this necessary documentation, some other documentation which provides more detailed information about specifications, audit trail, and issue logs can be considered as 'sufficient' documentation. Before deciding the necessity or sufficiency of the documentation, it is important that the programmer consults with appropriate management representative and if needed with the quality assurance representative regarding documentation needs.

- Programming Activities and documentation: Once programmer knows what to document and how much needs to be documented, management needs to take a thorough assessment of work load of the programmer and see if the assignment can be done by the programmer. Programmers should also do this type of pulse check and proactively approach management if the documentation needs put excessive burden on programmer and may jeopardize the ability of programmer to perform his/her programming activities efficiently. This can lead to wider discussion resulting into appropriate resourcing for the programming activities.

### **Conclusion:**

Programming function within pharmaceutical industry is often viewed more as a software developer or support function to core biostatistics and data management services. Despite such status of programming service, most of the work performed by statistical programmers is part of core deliverables of clinical trial. While programmers focus solely on meeting customer needs, it is important for programming management and staff to plan QC strategy and conduct the quality assessment of deliverables and processes accordingly. Although above review provides some guidelines about best practices in QA, ideally management should layout best practices by considering work environment and constraints within the organization. The concept of 'think global act local' applies to programming function as well. While programmers need to provide best quality outputs, such work should be aligned to the trial level processes and documentation needs. Delivery of best quality output in timely manner is definitely an important objective of clinical programmer. However, maintaining process integrity and consistency is a broader yet core requirement as per regulatory guidance in pharmaceutical industry.

### **Acknowledgement**

Author would like to acknowledge following individuals:

Helene Sabia, Associate Director, Taisho Pharmaceuticals R&D, Inc.: For her thorough insight and perspectives on clinical operations which helped me understand the challenges faced by sponsor while managing the clinical trial in a fast paced research oriented environment.

Cheryl McCarthy, Associate Director, eClinical Solutions, a division of Eliassen Group: For her guidance and mentoring in quality assurance best practices which gave me broader perspective of implications of programming service across the trial level processes.

### **References:**

- 1) CFR- Code of Federal Regulations Title 21, on <http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?fr=11.10>

- 2) Food and Drug Administration- Guidance for Industry- Process Validation: General Principles and Practices. On <http://www.fda.gov/downloads/Drugs/GuidanceComplianceRegulatoryInformation/Guidances/UCM070336.pdf>
- 3) Good Clinical Practices- Guidance for Industry, on <http://www.fda.gov/downloads/Drugs/GuidanceComplianceRegulatoryInformation/Guidances/ucm073122.pdf>

**Contact Information:** Your comments and questions are valued and encouraged. Author can be contacted at [pshiralkar@eliassen.com](mailto:pshiralkar@eliassen.com) or at [pcshiralkar@gmail.com](mailto:pcshiralkar@gmail.com)

SAS® and all other SAS® Institute Inc. product or service names are registered trademarks or trademarks of SAS® Institute Inc. in the USA and other countries. ® indicates USA registration.