PharmaSUG 2017 - Paper IB06

Data Monitoring Committee Report Programming: Considering a Risk-Based Approach to Quality Control

William Coar and Amber Randall, Axio Research

ABSTRACT

Quality control (QC) is fundamental to ensuring both correct results and sound interpretation of clinical trial data. Most QC procedures are a function of regulatory requirements, industry standards, and corporate philosophies. However, no one should underestimate the importance of independent, thoughtful consideration of relevance and impact at each step in the process from data collection through analysis. Good QC goes far beyond just reviewing individual results and should also consider monitoring data throughout the course of a study. In particular, QC is essential when supporting a Data Monitoring Committee (DMC). Given the nature of interim and incomplete data, inherent challenges exist when it comes to generation of DMC reports. Many of the usual practices associated with quality control need to be adapted to accommodate the repetitive nature of DMC review on accumulating data that may have outstanding queries.

This presentation will explore adaptations to a typically rigid QC process that are necessary when reviewing interim/incomplete data. Such adaptations focus on a risk-based approach to QC to ensure that a DMC can make informed decisions with more confidence in the data and programming.

INTRODUCTION

Quality control (QC) in programming is fundamental to good decision making. If businesses and scientists are willing to invest valuable time, money, and resources, then it follows that they would want to ensure that the results are correct. That may sound simple enough, but digging a little deeper uncovers a labyrinthine list of considerations and approaches [1].

Most procedures to ensure quality control can rapidly become very intensive, in time, effort, and money. Further complications arise when critical decisions are being made based on iterative reporting. Because of the repetitive nature of interim reporting on incomplete and erroneous data, traditional methods of QC assuming a clean/locked database may not be optimal or even feasible. A thoughtful plan is imperative, and may even evolve over time.

Such is the case with supporting a Data Monitoring Committee.

DATA MONITORING COMMITTEES

The Data Monitoring Committee is a group of experts tasked with oversight of a clinical trial. They typically consist of 3-6 experts in the disease area, including one statistician. Most often they are external to pharmaceutical companies (i.e., Independent), but there are cases where internal DMCs are utilized. Not every study requires the use of a DMC, though they are becoming more common in rapidly growing fields like Oncology.

DMCs operate under a formal Charter which acts as operating guidelines. The committee's primary responsibilities are to safeguard trial participants and protect trial integrity. They also provide insight on scientific and practical issues. They serve as advisors to a Sponsor drug company, and do so through formal recommendations. The DMC forms recommendations through periodic data review meetings. Each meeting typically consists of at least two sessions: an Open Session and a Closed Session.

Open Sessions allow the DMC to gain insights from the study team. However, by-arm results are generally not discussed, even in open label trials. An abbreviated set of 'total-only' tables, listings, and figures (TLFs) are typically provided to support the Open Session discussion. This is called an Open Report.

All by-arm discussion is generally reserved for a Closed Session. This only includes the DMC and an independent supporting statistician from a Statistical Data Analysis Center (SDAC). No Sponsor representatives attend the Closed Session. Data are reviewed through a more comprehensive set of TLFs called the Closed Report. Programming support to generate the TLFs comes from the SDAC.

STATISTICAL DATA ANALYSIS CENTER

The SDAC is essentially an independent group of statisticians and programmers whose role is to support the DMC. While they are contracted through the Sponsor, they ultimately serve the DMC. They prepare the reports and provide logistical support for each DMC meeting.

In preparation for each meeting the SDAC receives data from a data management group to generate analysis datasets and TLFs for both the Open and Closed Session. Historically, only summary tables of safety data were presented. However, other TLFs are now commonly added to allow the DMC to monitor study progress. An important feature of the DMC report is that it is not simply a subset of TLFs from a study report. There are often tables or figures provided to allow the DMC to assess study conduct (such as summarizing data capture) that are never considered for a study report.

Deliverables to the DMC are iterative by design. Early analysis data and outputs generated often with incomplete/dirty data. Even the best defensive programming can't predict the future so programs often require continual updates and QC.

A PROCESS PERSPECTIVE

Figure 1 provides a general overview of a clinical trial from design to the final reporting of results. A clinical trial is a designed experiment with objectives and goals to further science. After the protocol is final and analysis plan established, patients enroll in the study. With that begins data collection and data cleaning. In more recent studies, CDISC standards are considered for implementation. This step is not typical in historical studies where analysis data were created using CRF data. With CDISC standards, the programming team must first create tabulation data (SDTM) from the CRF data. Analysis datasets (ADaM) are then derived from tabulation data rather than the original CRF data. Programming of TLFs provide the analyses to support the final study report. Specifying, programming, and testing can all be considered as "analysis specification, programming, and validation". The programming process may include a final step to assemble the final set of TLFs into a single file for review and electronic distribution. This may involve SAS®, scripts from 3rd party software, or other custom processes. Regardless of the approach the end goal is often to have a single file with hyperlinked table of contents and bookmarks. DMC reports may be produced at several points in time throughout this process as deemed necessary by the study team and the DMC members.

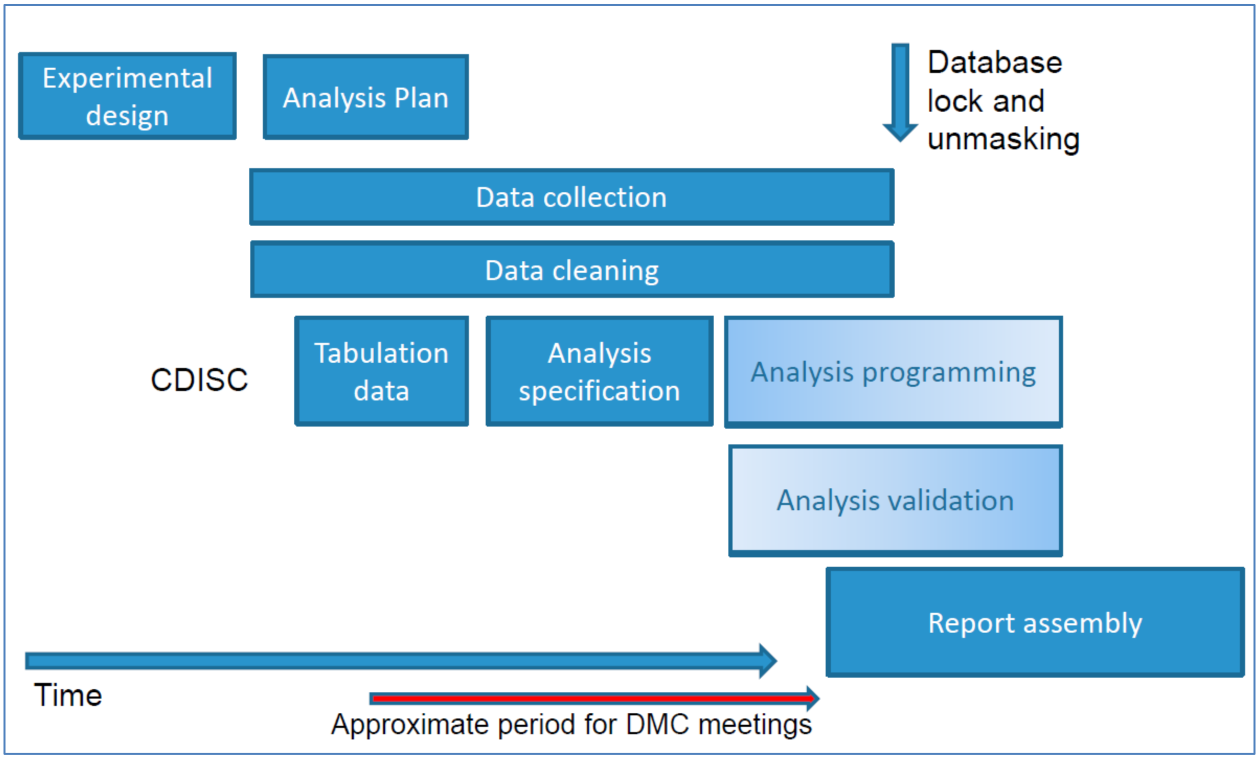

Figure 1: Example Process

Figure 1 also notes that this process takes time, often years for many studies. However, it is essential to understand how time impacts the timing of the final deliverable. SAS programming is used throughout the process, and sequential dependencies are unavoidable. Even with the best programmers, a project team simply can’t wait until database lock to begin development of SAS programs for SDTM, ADaM, and TLFs.

Ideally, most SAS code is in place prior to database lock. Some validation of SAS programs may also be completed prior to database lock so final results are more quickly attainable. However, final validation of critical components (such as primary safety and efficacy tables) will likely occur after database lock. A strategic (and wise) approach to programming and validation may be to begin SAS program development early, but wait until mature data exist to formalize the validation of each SAS program. A good project plan will ensure that this is feasible, but what if there are iterative data reviews where critical decisions are being made prior to database lock and final reporting? Whether the iterative reporting is to support a DMC, a formal interim efficacy analysis, or to support key business decisions, adaptations to traditional QC measures may be necessary.

Figure 2 below aims to focus on key components that utilize SAS programming throughout the conduct of a study as they apply to a standard CDISC data flow.

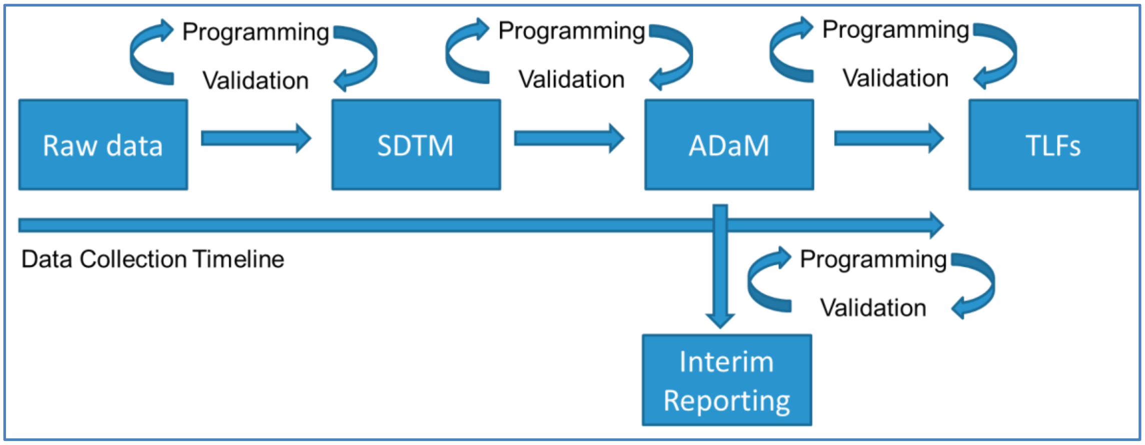

Figure 2: Data Flow Process with CDISC

When CDISC is part of the data flow model, extra steps and additional data dependencies are required. Early drafts of SDTM may prove to be inadequate as the raw data mature. Updates and modifications to SDTM programs are common as a trial progresses. Given the inherent dependencies in this data flow model, modifications to components of SDTM may have direct impacts on existing subsequent analysis datasets and/or TLFs. These subsequent modifications may invalidate any existing ADaM or TLF code that may have been previously verified.

In a process with iterative reporting and key decisions being made regarding the safety and wellbeing of patients under study, some form of validation is necessary with each iteration. This is where traditional methods may not be optional, or even feasible. Adaption is required to maintain an acceptable level of quality, and adaptation itself may need to evolve over time. Validation is a critical component and iterative with each deliverable, but may differ from that when reporting final study results on complete/locked data.

Depending on the nature of the (iterative) deliverable, the cyclic nature and inherent dependencies can easily become overwhelming. Programming itself, being on interim/incomplete data, can rapidly become complicated as well. For example, it may be known that one or more subjects have died, yet death dates are missing. In a study evaluating cardiac events, some events may have been through adjudication (thus confirmed), while other have not (remaining unconfirmed). There may be patients with data suggesting they ended treatment, however there is no exposure data entered yet so typical programming rules flag them as untreated. Could you really discontinue treatment if you weren't treated? Is the issue that there is a lag in data entry of exposure data, or actually an error by the site filling out an End of Treatment page incorrectly? Interim and incomplete data isn't able to distinguish between the two.

In order to ensure quality is maintained ***at each iteration*** of a DMC report, programmers need to decompose the process, evaluate new and existing dependencies, and assess the impact and likelihood of error for each SAS program including the overall impact on the final deliverable.

DEVELOP A QC PLAN

A Quality Control (QC) Plan will help simplify a potentially complex process without adding complexity itself. The QC Plan should:

- Assess each step in the workflow and how it impacts the quality of the final product
- Allow flexibility to consider risks/consequences of making a mistake
- Allow flexibility to consider the external factors that may influence decisions
- Allow flexibility to adapt as data accumulate and focus areas change
- Focus on what is most important at each step and at each iteration

As the project evolves, so may the QC Plan. At each step of the process at each iteration, evaluate each component of the QC triangle (Figure 3).

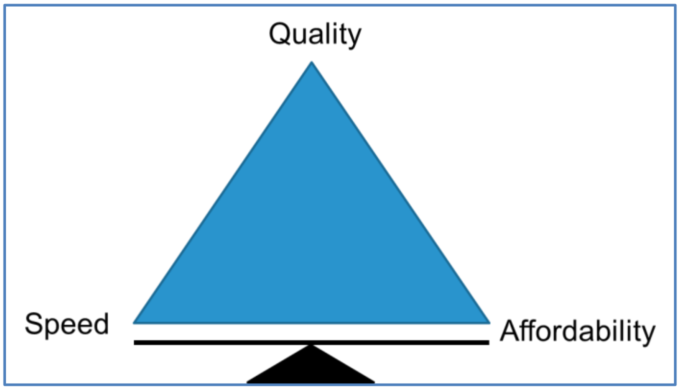

Figure 3: Balance in the QC Triangle

An effective QC Plan should seek to achieve balance between speed and affordability yet maintaining a high degree of quality. For each task (at each iteration), ask WHO, WHAT, WHEN, WHERE, and HOW while considering the QC triangle.

The nature of iterative reporting of incomplete and dirty data may dictate a QC Plan that is quite different than what might be used for final reporting on clean, locked data. In addition, not all results hold equal value in a DMC report review. More emphasis is often placed on serious adverse events, endpoint review, and study conduct than on topics such as concomitant medications. Your QC Plan should also take this into consideration. When the iterative review is to support a DMC, the final deliverable is likely two sets of TLFs. The first set is the *Open Report*. The second set of TLFs is a *Closed Report*. Recall that the Open Report is an abbreviated set of TLFs with a total only column whereas the Closed Report is a set of TLFs that presents by-arm information. The set of TLFs are initially agreed upon by the DMC, SDAC, and study team at the beginning of the study by looking at shells. However, these often change with each data review based on the data seen and the requests of the DMC to look more closely at topics of interest or concern. For the purposes of this discussion, a deliverable for each DMC meeting is a set of TLFs that have by-arm results with sufficient information to allow a DMC to make a recommendation regarding the continued conduct of the trial to the Sponsor organization. Typical recommendations are: continue the study, continue the study with protocol modifications, or stop the study.

While the DMC issues recommendations, in most cases Sponsors accept the recommendations. Protocol amendments can be time consuming causing delays, and stopping a study can be detrimental to the development program. In the event of a safety signal or some other signal suggesting a protocol modification, it is critical to have an (accurate) DMC report that represents the data in its current form.

ADAPTATION FOR ITERATIVE REVIEW

Adaptation of the QC process allows each task at each step in the process to evolve. Two considerations should be made – the effect of any upstream dataset changes on downstream programming and an assessment of the most important aims for each deliverable at that point in time. For a given iterative data review, the first step is to *clearly define the deliverable*. The DMC may have requested modifications to existing TLFs, or maybe even requested new TLFs. In most cases, there are only 2-3 weeks between a data cut and delivery of the TLFs to the DMC. It is important to identify the exact deliverable for that iteration early so resources can be allocated accordingly.

Once the deliverable is established, the programming team should *identify critical components for decision making* at that iteration. Choose validation methods that are appropriate for that iteration. Recognize that critical components may change. Moreover, complexity of SAS programming may change as well. The DMC may find potential signals emerging as data evolve, and the seriousness of those signals may warrant changes in critical focus areas. For example, the DMC may find potential drug interactions resulting in concerning liver abnormalities that were unexpected. It may be the case that additional requests are made to provide reports of potential Hy's Law cases. It is important to focus on what matters most *at that point in time*.

VALIDATION

The gold standard for validation methods is reproduction of results by an independent programmer. This is the most rigorous approach, and the most expensive. However, it is not foolproof. Mistakes still happen for a variety of reasons such as programming strictly to the specifications or improper allocation of resources.

At the opposite end of the spectrum is code review. This too should be performed by an independent reviewer [2]. This certainly takes fewer resources, but may not be appropriate when there are substantial (or any) data manipulations. It most certainly is not appropriate for TLFs that may be used to make critical decisions. Any number of hybrid approaches also exist, including the use of automated tools (such as macros) and third party software (e.g., Pinnacle 21).

As seen in Figures 1 and 2, analysis datasets are created to allow more efficient generation of TLFs. Validation of analysis datasets typically involves a second programmer reproducing the analysis dataset and performing an electronic compare procedure (Proc Compare). A clean Proc Compare suggests a second programmer independently reproduced the analysis data, thus providing assurance that the dataset is correct.

Note that programmers may not have much (or any) control over the inputs to analysis datasets (or even SDTM). However, programmers may have much more control over the inputs to the TLFs through the use of analysis datasets. Moreover, as data mature and critical code continually passes validation without modification, programmers may have more confidence in the output without requiring independent verification of every data point in a table or listing.

It is essential to keep these in mind as you consider adaptations to the QC Process.

DEMONSTRATION OFADAPTIVE QC PLAN

A typical DMC report includes 20-30 summary tables, 3-5 listings, and a few figures. To illustrate how validation methods can/should evolve, we consider QC based on early data and contrast it with QC of those same TLFs when data are more mature. The critical components for decision making are initially identified as:

- Study conduct: Are stratification factors balanced?
- Fatal Adverse Events: does the study drug have adverse reactions with outcome of death
- Is there benefit to offset risk: are patients living longer in the presence of (tolerable) side effects?

There are many other critical factors, but we choose these three for illustration only.

Early Data Review

Early data reviews are often done with a small number of data points. Data are missing, incomplete, and may have outstanding queries. From the beginning, programs easily become far more complicated to accommodate sparse data. As complication increases, so does the likelihood for data.

Stratification factors may not be the first thing that comes to mind when it comes to study conduct. However, stratification factors are part of the protocol design. If not implemented correctly they can impact the efficacy analyses, thus are high risk. These variables are generally in ADSL, which will undoubtedly go through rigorous QC being reproduced independently.

Generation of frequency counts can be done quite easily given a validated ADSL. Regardless, independent programming is suggested to assure randomization is balanced within stratification. While there is a high degree of accuracy with IxRS systems (interactive voice or web randomization system), mistakes can happen. Early detection can allow corrections, which may actually allow the study to continue while maintaining integrity.

Other tables to assess study conduct are: enrollment over time, enrollment by site, time from randomization to first dose, rate of protocol deviations (including inclusion and exclusion criteria) and completeness of data capture.

Regarding fatal adverse reactions while on treatment, undoubtedly these are closely monitored. Deaths as an outcome of an adverse events while a patient is on treatment may be viewed differently from a death due to disease progression after a subject is likely on subsequent therapy. Fatal adverse events while on treatment is not the same as all-cause mortality. As signals emerge in DMC reports, the DMC members use their collective expertise to assess the imbalance from both a statistical and clinical point of view. The DMC has to assess whether it is ethical to continue patients in the study and put (future) patients at risk if the drug increases the likelihood of death related to study drug. In early data reviews, there may only be one or two, if any, patients with fatal adverse reaction. This information is presented in both a summary table and a data listing.

Given the critical nature of fatal AEs, the first reaction may be to have the table independently reproduced. Independent production of ADAE is recommended. However, in these early data reviews, time spent on reproducing a summary table may not be necessary. If there is only 1 patient with a fatal event, this can easily be verified manually with very little time and effort invested. While this manual review may be sufficient for this early validation, it won’t be sufficient as data mature and become more complete and complex.

To allow a DMC to assess potential benefit in the presence of risk, a Kaplan Meier curve of time-to-death is often presented. Deaths are undoubtedly are a key safety concern. Analysis datasets (ADSL or ADTTE) will contain the necessary information (event/censoring flags, duration variable). These are typically reproduced by independent programmers. Censoring rules for other efficacy measures such as progression free survival can become rather complex and difficult to program at the final stages of a study. However, in the absence of mature data, simplified censoring rules may be required. In addition, data that might normally be consistent may not be with immature data. The programmers may need to search disposition, AE, and other datasets to locate all clues that a death occurred whereas at the end of the study the deaths may be reliably located in just one dataset.

Extensive efforts to validate the analysis data may allow a programmer to utilize a hybrid approach to verification of a Kaplan-Meier (KM) figure. For example, it is not uncommon for programmers to have some standard code (or macro) to generate such a figure. Since KM figures with few events may provide extremely limited information anyway, code review by an independent programmer (or statistician) may be acceptable to verify the macro is called correctly. As with early analysis of fatal adverse events, this validation approach may need to be reconsidered and replaced with independent programming as events accumulate.

Iteration with Mature Data

Assessment of stratification factors continues throughout the course of a DMC. However, with later reports, methods of validation may change without sacrificing the quality of report. ADSL continues to be revalidated with each data review. This is especially true when it is known that input datasets change (as is the case when SDTM specifications change). These changes are more common for earlier reviews, but later reviews are not immune to such changes either. Given the control programmers have over the inputs to the summary table, it may be sufficient to have independent code review, or have an independent programmer verify that no changes were made to the program that generates the summary table. As enrollment completes, less emphasis is given to the review of stratification factors with respect to study conduct. Even if issues were found within the IxRS system, enrollment is complete so nothing can be done. Since less emphasis is put on such variables in these later reviews, the consequences of making a programming mistake at this point are low.

Fatal adverse events will continue to require independent revalidation of ADAE as the risk remains high. With enough data, independent programming of the summary table of fatal adverse events is warranted. If imbalances emerge, additional requests to the programming team from the DMC are likely.

A KM approach to analyzing the overall survival endpoint becomes more reliable as data mature. However, it may be the case that the censoring rules for a final analysis may require complex programming given that the data may still be missing or incomplete. While it is not advisable to modify censoring rules, it may be required (see below). It is advisable to establish reasonable censoring rules early in the study that may allow the DMC to make informative decisions, yet may differ from those used in the final planned analysis. As Overall Survival data mature, it is advisable to have a second programmer (and maybe a statistician) independently produce the figure or perform additional measures of QC to verify that the KM figure accurately reflects the data.

An exception to the recommendation regarding modification to censoring rules is when a formal interim analysis of efficacy is conducted. When this happens, all efforts should be made so that censoring is consistent with the actual Statistical Analysis Plan. Censoring rules for a formal analysis may be far more complicated when early stopping for efficacy or futility is planned. Programming of these rules may be more complicated as well as incomplete dates often interfere with determining censoring times.

While these three may be straightforward, they serve as good candidates to demonstrate the need to evaluate adaptations in validation methods.

OTHER ADAPTATIONS

A common source of error is inconsistency in data entry. Queries may eventually resolve fixing the inconsistency in the (interim) data, but seemingly correct code may need additional maintenance. For example, sites may fill out an End of Study CRF page without filling out an End of Treatment CRF page. Alternatively, a patient may have a record in the End of Treatment CRF page but not yet have exposure data. In either of these cases, initial programming may need to be updated to resolve inconsistencies in a disposition summary table. Depending on the number of inconsistencies, a programmer may choose to save a version of ADSL prior to the modification. This can be used later for a Proc Compare to show that only the expected changes occur. This is yet another form of verification that the modifications are expected and correct. Again, depending on the number of inconsistencies, this may be sufficient when resources are strained. An alternative approach is to leave the inconsistencies it the summary table and explain them to a DMC. In this case, validation procedures may not be “clean”, yet the ADSL and disposition table are deemed acceptable. It is important to recognize that this may be acceptable as long as they are documented, declared to the DMC, and don’t impact interpretation of the final results. (This makes them low risk.)

The risk of finding inconsistencies in the data is actually higher when a DMC requests “current’ data rather than “clean” data. Most DMCs recognized that clean data for an interim review simply isn’t possible, and accept that inconsistencies may arise. Inconsistencies should not be viewed as a poor reflection of a Sponsor or an SDAC as both are making significant efforts to provide results most represented of the data. Rather they should be recognized as a consequence to a request for data as current as possible.

Other source or programming errors are changes in the source data structure. As noted earlier, programmers have less control over the inputs to analysis datasets. Early in the study, it is not uncommon for database changes to occur. These may have downstream effects on subsequent SAS programs. A QC Plan should include assessment of data structure between successive interim reviews. It is important to identify these problems early in production of the iterative review so that proper validation methods are chosen to accommodate the changes.

The last adaptations discussed here is the use of automated tools. These can be expensive, yet provide great efficiencies in the DMC process. We note that automated tools are only as good as their inputs. There is no substitute for defensive programming within custom tools.

Adaptations with CDISC

To some degree, there is more control of input to analysis datasets when SDTM is provided to support a DMC. However, for how rigid the standards are, there is still an art to implementation. There will always be areas open for interpretation. Programming rigid standards using sparse data make programming complex. Specifications need continual updates to accommodate dirty data. Striving for compliance may not be essential to support a DMC unless a Sponsor is performing formal interim analyses for efficacy.

CONCLUSION

When critical decisions are made based on data, validation and quality control are essential. Programming with interim/incomplete data only complicates matters. Early data are often sparse while later data is more mature. As a project evolves, trust is developed with existing SAS programs. QC efforts continue, but may ultimately have evolved since early data reviews. At some point, additional efforts are placed on independent review rather than independent reproduction. Processes may be in place to continue the independent reproduction, but independent review is essential to provide critical thinking. Without it, complacency can overshadow quality control.

When supporting at DMC (or any other decision making process involving iterative reports), recognize that programming will evolve over time. Key outputs may change based on emerging results, and it may be necessary to reprioritize. It is essential to document and track what changes over time. It is not always possible to be perfect. As long as humans are involved, mistakes will happen. Thus, it is essential to focus on what matters most at each iteration, and adapt validation methodology to do so while maintaining the overall integrity of the report.

REFERENCES

[1] Randall, A, and Coar, W. 2017. “Quality Control Programming: A Lost Art?” Proceedings of the SAS Global Forum 2017 Conference, Orlando, FL. Available at http://support.sas.com/resources/papers/proceedings17/.

[2] Marchan, C. and Tinazzi, A. 2016. “What Auditors Want.” Proceedings of the PHUSE 2016 Conference, Barcelona, Spain. Available at http://www.phusewiki.org/docs/Conference%202016%20RG%20Papers/RG04.pdf.

ACKNOWLEDGMENTS

The authors would like to thank the programmers and statisticians at Axio for their discussions contributing to this paper. The authors would also like to thank Joshua Sanders for his insights from a Quality Assurance perspective.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

William Coar, PhD
Biostatistician/Director of Statistical Consulting
Axio Research, LLC
2601 4th Avenue, Suite 200
Seattle, WA 98121
Email: williamc@axioresearch.com

Amber Randall
Director of Statistical Programming
Axio Research, LLC
2601 4th Avenue, Suite 200
Seattle, WA 98121
Email: amberr@axioresearch.com