

Improving the Relationship between Statisticians and Programmers in Clinical Trial Studies

Mai Ngo, Mary Grovesteen, and Vaughn Eason, Catalyst Clinical Research, LLC

ABSTRACT

Successful deliveries of analysis outputs for a clinical trial depends on a strong biostatistics team, which typically includes a study statistician and a programming team with several statistical programmers. As trials get more complex and biostatistics teams face increased pressure to produce outputs efficiently and on a timely basis, a strong working relationship between the study statistician and the programming team is vital to the success of the analysis project. Yet with the time pressure and the increased complexity of the analysis as well as challenging data issues common in clinical studies, the communication between the statistician and programmers tends to break down when it is needed the most. This results in frustration from both sides, inefficiencies that could have been avoided, and stressful last-minute work and rework. Having worked both as statistical programmer and statistician, we have been fortunate to gain valuable hands-on perspectives from both sides. Based on personal reflections as well as conversations with our colleagues, we will present some of the key areas of frustration in the working relationship between a study statistician and the programming team, touch on perspectives from both the programmer and statistician, and offer suggestions for alleviating these issues.

INTRODUCTION

There are many parts to a clinical trial, including writing the protocol, enrolling subjects, identifying sites for drug administration and data collection, and safety monitoring by clinical professionals. Due to the intense focus on setting up the trial and making sure it runs smoothly without any safety issues, there is little attention left over for the final analysis, which occurs after all the data has been entered and the trial is over. If the trial drags on longer than expected due to low enrollment or other issues, the timelines for the final analysis can be delayed and even shortened. The uncertainty of the clinical trial timelines creates pressure for the statistical teams, which are made up of statisticians and programmers, that are expected to produce the final analysis. If the timelines overlap with other trials that the same teams are working on, the teams feel additional pressure to figure out ways to meet the competing deadlines.

Because of the intense pressure statistical teams can face to produce the analysis outputs of a clinical trial, tension often occurs between the statisticians and programmers. Programmers and statisticians have common complaints about how the failure of one side or the other to be effective at their jobs can make their own jobs more difficult. Based on our own experience as programmers and/or statisticians and drawing from the insights of our colleagues, we have identified some common complaints programmers and statisticians have about each other. The first and second main sections of this paper presents common complaints of programmers and statisticians, respectively, along with each role's perspective and a common ground solution. We hope that this will help both programmers and statisticians begin to understand each other's perspective and work together more effectively as a team.

PROGRAMMERS' COMMON COMPLAINTS

1. THE STATISTICAL ANALYSIS PLAN (SAP) OR ADAM DATA SET SPECIFICATIONS ARE TOO VAGUE

The Programmer's Perspective

It is frustrating for a programmer to write a program when the SAP or ADaM data set specification is vague. For example, there are times when programmers have to program based on a SAP that does not have a clear operative definition of the main analysis variables, lacks a definition of how to handle missing data or partial dates, or specifies analysis variables that can't be programmed based on the data collected

The Statistician's Perspective

The study statistician cannot think of every possible scenario that might show up in real data. Sometimes, because they are pressed for time, statisticians may overlook listing annotations, thinking they are straightforward. The study statistician may also think that programmers should be proactive and bring up questions if they see holes in the specifications or things in the data that would affect the specifications. Even when the statistician has the best intentions, some part of the specs may still be unclear to another set of eyes. Also, data issues that were hard to foresee at the time the SAP or ADaM specifications were written require further fine-tuning of the specifications for the derivations of the analysis variables.

A Common Ground Solution

A good SAP needs teamwork and iterations between the statistician and the programmers. The statistician should put on the programmer's hat while writing the SAP and ADaM specifications so that they are programmable given the data collected and cover anticipated questions from programmers. They should become familiar with CRF forms and raw data when they write the analysis specifications. On the other hand, programmers should take more ownership of their programs and proactively bring up questions if they see holes in the analysis specifications or data issues that may affect the specifications. If time and budget allow, a review of the SAP and mock shells by the lead programmer from the programming perspective is always helpful.

2. STATISTICIANS ACT LIKE HERMITS

The Programmer's Perspective

It is frustrating for programmers to work with a "hermit" statistician. The hermit statistician lacks communication to the programming team on several fronts. They do not communicate deliverable timelines or expectations, e.g. the number of outputs until the end of the study, how many outputs including how many similar versus unique outputs, etc. They do not communicate upfront the expected general presentation formats, such as date format, proper versus all upper case, display of Unscheduled visit labels in listings, etc. They are not upfront about which issues they need time to think about or potential upcoming changes which can leave programmers in the dark, not knowing when and what they can program and what their workload for the study will be. They are frequently slow or forget to reply to programmers' questions.

The Statistician's Perspective

From the "hermit" statistician's perspective, they do not deliberately set out to annoy or create extra works for programmers. Often, the statistician may be shy or introverted and has numerous project-related tasks for which they are responsible. In addition, they may need to block out some time to think deeply about some difficult analysis problems. This sometimes results in slower responses to programmers' questions and unintentional forgetfulness in relaying important information to the programming team. Frequently, the lead statistician is not the one who decides on the timeline and may be under pressure from a client or management to meet constantly changing and tight timelines. In addition, the lead statistician may not want to interfere with the lead programmers in communicating to the programmers about workload and work assignments. As they are not as deep in the data as programmers, they may not be aware of smaller details such as the value in the raw data for Unscheduled visits or the date formats, etc. in the data.

A Common Ground Solution

If the budget and time allows, a regularly scheduled short team meeting can help solve communication blockages. A lead statistician should make an effort to find out the programming team's preferred communication styles and meet them at least halfway. In addition, a lead statistician should warn programmers of possible future requested additional data sets, or the way in which a deliverable should be formatted so programmers, especially the lead programmer, can plan ahead and develop macros to help save time. For example, the study statistician should let programmers know upfront if outputs are needed in a format that are not the common output formats in the company (e.g. RTF, Excel vs. PDF), or if all outputs will need to be split for certain subsets of the data, such as by region or subject population.

On the other hand, programming team members should try to understand the statistician's workload, perspectives and preferred communication methods and respect that preference when possible. Lead programmers can also help share the load of communicating to the team about work assignments and deadlines and field programming-related questions.

3. SPECS ARE WRITTEN WITHOUT CONSIDERATION OF LISTING PROGRAMMING

The Programmer's Perspective

Listings are generally considered easy tasks and assigned to the least experienced programmers. In some companies, these are assigned very limited time targets. Therefore, programmers often expect the programming of the listing to be straightforward to meet the limited time target.

Programmers can become frustrated when either the mock listings are not provided or the mock listing shells are inconsistent with the SAP text, lack clear annotations, or do not reflect what is collected on the CRF. Left having to guess about what the listing should look like or spend extra time trying to find non-existent variables can eat up a programmer's time and jeopardize the timelines.

The Statistician's Perspective

It requires much more work to add variables that are needed for listings but not needed for tables in the ADaM data set specifications. This also requires additional thinking about the structure of the ADaM data set to accommodate listing programming. For example, if lab listings need to list visits that are not done, the corresponding ADLB needs to include the visits that do not have lab results instead of just including non-missing results. Sometimes the budget and timeline for writing the SAP and accompanying mock shell is underestimated and the statistician must forgo some attention to details in listing shells to meet these targets.

A Common Ground Solution

Lead statisticians should pay attention to make listing programming as straightforward as possible by adding derived variables for listings to the ADaM data sets when it doesn't require too much change in the ADaM data set structure. They also should consult the CRF as much as possible when designing mock listings to make sure the shells reflect the actual data structure. A lead programmer or an experienced programming team member who is deeply involved with the data in the study can help review the listing shells to make sure the mock and annotations are data-driven and easy to follow for programming. Management can help make this easier by not underestimating the amount of work required to create the shells and understand that while this takes more time to create good mock shells upfront, it saves programming time later.

4. THE ANALYSIS OR MOCK SHELL DESIGN DOES NOT TAKE INTO ACCOUNT PROGRAMMING EFFORT

The Programmer's Perspective

It can be frustrating for programmers to program complicated data set structures and table and listing layouts. As programmers are often under pressure to program outputs without going over budget, inefficient analysis data set structures and overly complicated layouts for tables increase the programming time, leaving programmers unduly stressed.

The Statistician's Perspective

The statistician is not very involved in programming and may not be aware of which way of presenting data takes more programming efforts. Other times, the statistician may not want to compromise the best way to present data just because that requires more programming efforts.

Sometimes, as the study analysis strategies evolves, there's a tradeoff between completely changing the structure of some data sets to accommodate new strategies more efficiently versus modifying the existing

ADaM data set structure. The former is cleaner from the design of data sets but could create much more programming rework.

A Common Ground Solution

Whenever possible, statisticians should design shells and data set structures in the simplest way that can achieve the analysis goal. If there are several acceptable ways of presenting the data, lead statisticians should seek the lead programmer's or affected programming team member's input about which one requires the least programming effort.

On the other hand, if programmers are aware of an analysis design or specifications that can achieve the same analysis goal with much less programming effort, they should bring that suggestion up to the lead statistician.

5. STATISTICIANS ARE UNAPPRECIATIVE OF PROGRAMMERS' ROLE IN THE PROJECT

Some statisticians think programmers are just there to carry out the grunt work. They do not show appreciation of the programming effort it takes to make the deliverables possible and expect programmers to carry out all the specifications changes at the drop of a hat, without consideration for programmers' workload and work-life balance. They are quick to get frustrated and view programmers as lazy when they see programming lagging behind the project timeline.

The Programmer's Perspective

Programmers care about the quality of their work as much as statisticians care about with theirs. For example, when a statistician commits to an aggressive deliverable timeline, the quality and integrity of the programs used for creating the deliverables are compromised. How a program is written and serves its initial purpose has a lasting effect on the robustness and portability of the program. Over the course of the study statisticians make many minor changes to serve minor preferences, meanwhile rushing the work of the programmer, who is expected to make efficient programs while staying within the parameters of both the sponsor and personal preference of the statistician. The quality of a programmer's work is diminished, along with their morale, when they are not included in the conversation involving the changes and are expected to say yes to all changes, especially at the last minute. The programmer understands that sponsors can be nitpicky, and changes can suddenly happen, however the issue is being excluded from the conversation all together and being expected to react on a moment's notice. Quality, portability and robust programming are very important to the programmer and when this is no longer considered, friction in the workplace can occur.

The Statistician's Perspective

As the statistician may not directly be involved in the programming, they may be overly optimistic about meeting a deadline and underestimate the programming efforts. As a result, they fail to see the programmers' hard work to show enough appreciation. By nature, statisticians tend to be critical thinkers who have high standards and at the same time, introverted. As a result, they may not be effusive with their praises and fail to show their true appreciation of the programmers' hard work.

A Common Ground Solution

Both sides, programmers and statistician, need to show more appreciation of each other's work and consult each other about the feasibility of analyses and timeline and the best way to achieve the analysis goals. It is important that both sides exercise patience and seek to understand the other side's perspectives and how they work.

STATISTICIANS' COMMON COMPLAINTS

- 1. PROGRAMMERS, ESPECIALLY THOSE WHO PROGRAM ADAM DATA SETS, DON'T READ THE SAP AND ADAM SPECIFICATIONS CLOSELY AND WITH A CRITICAL MIND**

The Statistician's Perspective

The study statistician may want input from programmers. They also want ADaM data set programmers who take a proactive role in understanding the analysis goals and important analysis variables, especially efficacy endpoints. Without this understanding, programmers cannot catch holes in the analysis specifications or bring up data issues that impact important analyses. It also adds a lot of work for the statistician if the programmers just program blindly or take the specs too literally.

The Programmer's Perspective

Programmers may think it is solely the lead statistician's responsibility to make sure the specifications are written correctly and cover all bases. They are under timeline and budget pressures to program as fast as possible, so they may not have the time to delve deeply into the details of the statistical analysis itself.

A Common Ground Solution

This requires teamwork between statisticians and programmers. Statisticians should take extra care to write specifications and annotate the shells with enough details and when possible, cover all possible scenarios to make it easy to program. They should not expect the programmers to read their mind or between the lines when writing the specifications. While the statistician is mainly responsible for SAP and ADaM data set specifications, programmers should also take an active role and use their experience, common logical sense, and programmers' mindset to raise questions about areas in the SAP or ADaM specifications that are vague or have holes in the logic that needs further fine tuning.

2. LEAD PROGRAMMERS DON'T COORDINATE PROGRAMMING RESOURCES WELL AND ARE NOT PROACTIVE ABOUT TIMELINES, LEAVING INSUFFICIENT TIME FOR STATISTICAL REVIEW

The Statistician's Perspective

It's frustrating for statisticians when they are working with passive lead programmers. If a lead programmer doesn't coordinate resources well through either inefficient assignment of programming tasks so that similar programs are assigned to the same team member or fail to assign tasks to take advantage of programming team members' unique strengths, the ability to meet deadlines is hampered. Furthermore, when the lead programmer does not proactively work on creating internal programming timelines and communicate that to the programming team and the study statistician, the study statistician is left not knowing what is going on, when to block out their schedule for statistical review of outputs, and whether the deadline can be met. This also affects the programming team members' ability to plan their workload and can potentially impact the quality of the deliverables and risk timeline. As the study statistician is often the contact person for many non-statistical things such as budget and timeline issues, a non-proactive lead programmer and programming team causes additional stress when having to answer more questions from a client about the budget and timeline. If the study statistician decides to fill in the gap for the lead programmer, this essentially distracts them from being able to focus on solving analysis issues.

The Programmer's Perspective

Programming leads have to juggle many responsibilities, such as communicating with each team member about their tasks, assigning tasks based on programmers' abilities, setting up and maintaining the study program tracker and corresponding folders, keeping track of timelines and also programming the bulk of the study outputs themselves if resources are scarce. It is hard to anticipate issues that arise during the programming schedule and if timelines are constantly shifting, planning ahead is difficult. If a programmer is leading multiple studies, they may have competing priorities, causing deadlines to slip unintentionally. If they have no choice about the resources they are given for the study, they might have to work harder to maximize those resources, making it even more difficult to meet a deadline.

A Common Ground Solution

If study statisticians are more experienced with project management, they should take a more assertive role in suggesting assignments that utilize programming resources effectively while remaining respectful of the leader programmer's input. Additionally, companies can benefit by offering in-house project management training to lead programmers and lead statisticians.

To ensure meeting deadlines, it's helpful to have a clear and early assignment of tasks by the lead programmer, as well as built-in internal timelines and regular check-ins with programmers if programming progress is slow. Lead programmers should also keep the lines of communication open and keep the lead statistician in the loop of the planned programming timeline, programming progress, and when the outputs will be ready for statistical review.

3. PROGRAMMERS LACK ATTENTION TO DETAIL

The Statistician's Perspective

Because of tight timelines, usually when it comes to the time for the study statistician to do statistical review, it's already very close to the deadline. Programmers who are not detail-oriented produce outputs that distract the study statistician from focusing on the results since they must give many comments on presentation, typos, page breaks, etc. Sometimes programmers vary their presentation across similar outputs, which means the statistician must focus harder on pointing out those differences where they could have spent less time commenting on similar outputs. Other times, programmers may overlook the population or certain subset conditions, leading to incorrect results.

The Programmer's Perspective

Programmers may think it's a statistician's job to catch all the presentation issues and check if the analysis variables and outputs are correct. They are often under pressure to spend the least time possible on an output while working under multiple tight deadlines. As they are focused on programming, they do not have the same general perspective as the statistician whose role is to review the outputs.

A Common Ground Solution

Production programmers should review their outputs to see whether the presentation is legible and spacing and page breaks make sense, and that outputs use a consistent display format. Validation programmers should also give comments on the layout and formatting in addition to validating the contents of an output.

If programmers are assigned to multiple similar outputs, they should make sure to check consistencies across the outputs and if that inconsistencies are due to the shell, bringing them up to the statistician. The study statistician could start rolling statistical reviews to catch things that need fixing across multiple outputs and be upfront about their expectations for output formats. They can also proactively spot check for areas the programmer may overlook to make sure the programming is correct.

4. PROGRAMMERS ARE NOT RESPECTFUL OF THE STATISTICIAN'S TIME

Some programmers include the statistician in every validation email, expecting the statistician to provide input for every type of mismatch between the production and validation side, who don't provide data examples or details for their question, or who expect the statistician to provide immediate replies by instant messaging the statistician frequently throughout the day.

The Statistician's Perspective

As lead statistician, the statistician often deals with numerous emails from sponsors, project managers, data managers, as well as programmers. At the same time, they are expected to spend time thinking of solutions and researching analysis methods. Having to be on the receiving end of multiple emails in which they do not need to be included disrupts the statistician's chain of thoughts, unnecessarily increases their mental load, and reduces the time they have for their main responsibilities.

The Programmer's Perspective

Programmers want to ask statistician questions as they arise as this saves them time from rework or allow them to continue the specific program without having to resume once receiving answers. They may also want the statistician to help decide on data issues and resulting mismatches between primary and validation programs because of these issues.

A Common Ground Solution

Programmers should ask questions if they are not clear about the analysis specifications or if they find data issues that may affect analyses. However, it would be more helpful if they batch their questions and follow email etiquette to show respect for the statistician's time. This etiquette includes clear and to-the-point email subject line, only copying the statistician when necessary, providing enough details and data examples for their question, formulating their questions well, and including suggested solutions if they have any.

Statisticians can be more assertive about needing uninterrupted time to think and ask to be excluded from email discussions between programmers. It is important that statisticians communicate with the team about hours they plan to block out to focus on deep work and hours where they are open for questions. They can process questions from programmers at designated times during the day to ensure reasonably prompt responses while still being able to focus on other statistical lead's tasks. Also, the statistician needs to understand that the seemingly trivial task of changing the display or formatting sometimes turns out to be time-consuming programming rework. Therefore, they should as much as possible specify formatting rules upfront, thereby saving both programmers' time and their own time answering programming questions.

CONCLUSION

It truly takes a team effort to produce and deliver quality analysis outputs of clinical trial data on time. As in any line of work, team communication, project management and leadership play very important roles in the success of clinical data analysis projects. In addition, there is another layer to the relationship between statistician and programmers in clinical trial data analysis projects that is specific to the nature of the data, CDISC requirements and tight timelines. In this paper, we have highlighted some common issues and suggested solutions for these in this specific context. Application of common-sense project management and team communication approaches to this specific setting, respect for each other's time, the ability to see perspectives from the other sides, and a proactive and positive mindset will help a statistics team achieve their analysis goals and meet their deadlines.

RECOMMENDED READING

Ferraro, Jack. 2012. Project Management for Non-Project Managers. New York, NY: AMACOM.

Gustavson, Paul and Liff, Stewart. 2014. A Team of Leaders: Empowering Every Member to Take Ownership, Demonstrate Initiative, and Deliver Results. New York, NY: AMACOM.

Markova, Dawna and McArthur, Angie. 2015. Collaborative Intelligence: Thinking with People who Think Differently. New York, NY: Spiegel & Grau.

Mitchell, R.M. 2001. "Fast and Easy Ways to Annoy a Statistician: The Sharing and Presentation of Data Between a SAS Programmer and a Statistician". Proceedings of the 26th Annual SAS Users Group International (SUGI) Conference. Available at <https://support.sas.com/resources/papers/proceedings/proceedings/sugi26/p243-26.pdf>.

Mitchell, R.M. 2001. "Fast and Easy Ways to Annoy a SAS Programmer: A Statistician's Revenge!" Proceedings of the 28th Annual SAS Users Group International (SUGI) Conference. Available at <https://support.sas.com/resources/papers/proceedings/proceedings/sugi28/266-28.pdf>.

CONTACT INFORMATION

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

Mai Ngo
Senior Biostatistician
MaiNgo@catalystcr.com

Mary Grovesteen
Senior Biostatistician
MaryGrovesteen@catalystcr.com

Vaughn Eason
Statistical Programmer
VaughnEason@catalystcr.com