PharmaSUG 2020 - Paper DS-031

Ensuring Consistency Across CDISC Dataset Programming Processes

Jennifer Fulton, Westat

ABSTRACT

Whether you work for a small start-up, a mid-level CRO, or Fortune 500 biopharma company, CDISC compliance is a daunting prospect at the start of any project and requires creative solutions and teamwork. Consistency is the hallmark of a CDISC project and was the impetus for the formation of the consortium. Each CDISC project should be approached consistently as well, leading to improved accuracy and productivity, regardless of staff skill and experience. This approach leads to quality final product and FDA approval, ultimate goal. Westat approached this goal by developing an overarching CDISC development and delivery checklist. It provides a visual of the scope of a CDISC project, organizes work instructions and templates for users, and helps assure critical steps are not missed. Like the 26 miles in a marathon, our paper will lay out 26 steps to CDISC compliance, along with tools and techniques we have developed, to help others reach the finish line.

INTRODUCTION

In today’s CDISC programming environment, there are many tools available to help develop and validate CDISC datasets. Whether these tools are pulled from private vendors or CDISC forums or are homegrown within individual companies, they call for consistent usage and handling. Complications occur when one programmer uses a tool inconsistently from another. Managing each CDISC dataset similarly supports production of clean and compliant CDISC datasets. This paper will discuss processes to promote consistency across dataset programs and tools that can be used by programmers and other project staff responsible for ensuring CDISC-compliant deliverables.

Westat approached this goal by developing new Standard Operating Procedures (SOPs), which resulted in new work instructions, and culminated with an overarching CDISC development and delivery checklist. Benefits of implementing this checklist are:

- Supervisors and project managers determine the scope, timelines and costs of a CDISC project more efficiently.
- A large number of work instructions, templates, generalized SAS macros, and other references are encapsulated for easy retrieval and appropriate use.
- Step-by-step instruction assures critical data/information is not missed while providing a plan to navigate the CDISC project from start to finish.

The checklist items are high level and direct the user to other tools and references and for each we indicated the responsible party/parties, and target start and end date. Below we will walk the reader through our checklist of 26 key CDISC tasks and discuss some items in detail. Your company may elect to adapt the checklist to fit your needs and processes.

CHECKLIST ITEM #1: TRAIN PROJECT STAFF

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
1.	**Train Project Staff** The staff assigned to the project will complete the requirements in the CDISC Training Guide, in consultation with the Project Manager (PM) and other project leads.	All Project Staff as detailed in the CDISC Training Guide	Before any project work initiation	Before any project work initiation

A critical first step in the process of CDISC implementation is training on CDISC for project staff. Expanding your organization's CDISC expertise will increase your company's capacity to respond to CDISC implementation requests. Westat developed a training program on key aspects of CDISC, such as "Setting up CDISC Project Specifications", "Trial Design and Special Purpose Domains", and "define.xml and Supporting Documents for SDTM". Our CDISC experienced programmers mentor programmers new to the CDISC processes. The training program is continually updated based on staff experience, online CDISC webinars, and consultant expertise. Our designated CDISC Curator responds to all CDISC-related questions. Similar to an IT helpdesk, the Curator organizes the CDISC information and directs staff to additional resources within the company or online.

The detailed work instructions give project teams an organized set of references to initiate a CDISC project. These work instructions are referenced in several checklist steps. We recommend an abridged master checklist that is supported by comprehensive instruction documents.

We created a CDISC Training Guide for Checklist Item #1, which catalogues recommended trainings for staff at the start of each new project, based on their level of expertise. The CDISC Training Guide aids the project manager with recruiting staff for the project, and planning timelines and costs to ensure a team knowledgeable about CDISC processes.

CHECKLIST ITEM #2: SELECT CDISC TERMINOLOGY

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
2.	**Select CDISC Terminology to be used throughout the project** Determine the CDASH terminology, SDTM terminology, and ADaM terminology versions to be applied to the project. Save applicable terminology files to the project area and record version dates.	Data Manager or SAS Programmer	Before CRF development	Before CRF development

We have appropriate project leads consult with the CDISC Curator to determine the CDASH Terminology, SDTM Terminology, and ADaM Terminology versions for application to the project. The appropriate terminology files will be saved to the project folders and version dates recorded. The terminology version selected at the start of the project is the version used throughout the project. Custom terms may be added for extensible terminology as needed for the project. The terminology file is incorporated into a specifications document that serves as the basis for much of the programming of the CDISC domains and the define.xml. Our checklist also directs the user to current online terminology files.

CHECKLIST ITEMS #3-4: CDASH

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
3.	**Create CDASH-Compliant CRFs** Develop CDASH-compliant CRFs according to the Getting Started Guide for Data Managers.	Data Manager	At start of CRF development	Completion of CRF development
4.	**Create a CDASH-Compliant Database** Develop a CDASH-compliant database according to the Getting Started Guide for Data Managers.	Database Developer and Data Manager	At start of database development	Completion of database development

Checklist item #3 develops CDASH-compliant Case Report Forms (CRFs), and checklist item #4 develops a CDASH-compliant database. Our data management department developed a "Getting Started Guide for Data Managers" work instruction, which serves as a training and reference tool to accomplish CDASH-compliance. The SAS programmer at this stage reviews the CRFs to ensure the use of correct terminology throughout, and assesses other issues that may contribute to inconsistencies or unnecessary work for the SAS programmer.

CHECKLIST ITEM #5: DEVELOP SDTM SPECIFICATIONS

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
5.	**Develop SDTM Specifications** Initiate the SDTM specifications document according to the Work Instruction for Development of SDTM Specifications This document must be initiated prior to SAS programming, but it will remain a fluid document throughout the project.	Data Manager or SAS Programmer	At start of database development or prior to SAS programming	After database lock

Checklist item #5 develops project-specific SDTM specifications using a standard template. This template is based on SDTM specifications downloaded directly from CDISC and is the foundation for any CDISC project. Specifications include domain and variable lists. Additionally, there are worksheets for project teams to enter study-identifying information such as the name and version of the CDISC model and Implementation Guide (IG), terminology versions, trial design information, trial summary parameters, and data sources for developing SDTM data sets, define.xml, and the Annotated CRF (aCRF).

The SDTM specifications template produces a centralized document for project staff to access. These specifications drive the production of SDTM domains and define.xml since SAS programs pull information directly from the document. Generalized SAS macros that facilitate CDISC data development rely on the structure of the standard specifications, resulting in significant efficiencies.

CHECKLIST ITEM #6-7: PROGRAM SDTM DOMAINS AND CREATE XPT FILES

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
6.	**Program SDTM Domains** Develop all SDTM domains required for the study, per the Work Instruction for Programming SDTM and ADaM Datasets Note that primary SDTM data-generating SAS programs may be included in the final submission.	SAS Programmer	As soon as credible test data or study data is available	Initial programs completed 3 weeks after programming start; programs not considered final until after database lock
7.	**Convert SAS datasets to V5 XPT files** Convert SAS datasets to V5 XPT files using the sastoxpt.sas SAS program explained in the Work Instruction for Programming SDTM and ADaM Datasets Datasets must be converted to XPT prior to every iteration of Pinnacle 21.	SAS Programmer	Once SDTM domain programs are developed	XPT files must be created for Pinnacle testing; XPT files not final until SDTM domains are final

Checklist item #6 is comprehensive. This one step links to multiple instructions and tools that guide the production of SDTM datasets. SAS programs follow a standard design and incorporate a series of generalized SAS macros based on SDTM specifications.

The first and simplest macro is a program that creates the Trial Summary (TS) using study-specific Trial Summary parameter information entered into the SDTM specifications document. The project management staff provide information needed for TS. Stipulating the information in the SDTM specifications document, rather than in a SAS program, makes it easier to gather input on these parameters from project team members. An additional benefit is SAS programmers do not need to search for the information while developing the SDTM programs.

The next three SDTM domains establish Trial Arms (TA), Trial Elements (TE), and Trial Visits (TV) by importing the Trial Design section of the SDTM specifications into SAS programs. Identifying this information before the programming allows project managers to provide input on ELEMENT and EPOCH definitions for a clinical trial.

A macro to generate Subject Elements (SE) and Subject Visits (SV) is the most complex of the tools in the CDISC SAS programming repository.1 This tool underpins and incorporates all subject-level data in a study. The macro accommodates unscheduled visits, defines reference dates, and assigns ELEMENTs and EPOCHs for every subject visit.

Once the aforementioned domains are complete, the remaining subject-level domains are developed. SAS programs follow a standard design structure, which incorporates the use of a single macro to aid with multiple facets of domain creation. This tool creates variables common to all domains, such as STUDYID and USUBJID to reduce repetitive codes across programs. Additionally, it manages simple yet tedious processes, such as assigning variable labels, dropping unnecessary variables in the final data set, ordering variables, and sorting data sets by key variables as defined in the specs. It formats dates as ISO8601 (an international standard for covering the exchange of date- and time-related data) and calculates study day variables based on references dates. This tool also performs many CDISC and FDA checks for programmers, such as checking:

- 1:1 ratio of ARM/ARMCD, ELEMENT/ETCD assignments in specs
- Specs for missing variable origins and definitions
- Value Level metadata
- Missing variables in data compared to specs
- Variable attributes match between data and specs

- All data values match a value in a defined codelist, where applicable
- Required variables have no null values
- Key variables identify unique observations as efficiently as possible.

After all SDTM domains are generated, a post-processing macro rewrites every SDTM data set, assigns the maximum variable length for a single variable across domains per SDTM requirements, converts SAS data sets to V5 XPT files and checks file sizes.

CHECKLIST ITEM #8: CREATE DEFINE.XML FOR SDTM

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
8.	**Create define.xml for SDTM** Create define.xml for SDTM according to the Work Instruction for Development of define.xml and Reviewer's Guides Sent finalized document (near the end of the project) to the QA department for review according to the Guide for QA Review of define.xml and Supporting Documents	SAS Programmer	Once SDTM domain programs are developed	Initial define.xml completed with initial SDTM domains for Pinnacle testing; define.xml not considered final until after database lock

There are several approaches to creating define.xml, but checklist item # 8 includes a SAS macro that facilitates development of define.xml. The macro incorporates the SDTM datasets and the SDTM specifications previously described. Any time new data are available or specifications are updated, the macro can be run to update define.xml. This allows the programmer to create define.xml early in the process. It can be included in Pinnacle 21 checks to conduct a more thorough review of SDTM data and define.xml during development and throughout data collection and programming. This macro also frees the SAS programming teams from the burden of maintaining XML code.

CHECKLIST ITEM #9: RUN PINNACLE 21

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
9.	**Run Pinnacle 21 to validate SDTM datasets and define.xml** Run Pinnacle 21 to validate SDTM datasets and define.xml and review the report. Address findings where possible according to the Work Instruction for Programming SDTM and ADaM Datasets	SAS Programmer	After SDTM domains are created	Final Pinnacle reports are run and saved after database lock and all SDTM programs and define.xml are finalized.

Publicly available software, Pinnacle 21 (Community), is an industry standard to assess the compliance to model specifications of CDISC data. Pinnacle replicates many of the macro checks earlier described. However, we find it beneficial to be alerted to some issues and address them as the programs are being developed; then run Pinnacle to confirm the issues were addressed properly. Pinnacle includes an extensive set of additional checks.

CHECKLIST ITEM #10: VALIDATE DATASET-GENERATION PROGRAMS

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
10.	**Validate Dataset-Generation Programs through Double Programming** Independently validate SDTM programs through the double programming process outlined in the Work Instruction for Double Programming of CDISC datasets	Secondary, independent SAS Programmer	After SDTM domains are created	Double programming is considered complete after database lock and all SDTM programs and define.xml are finalized.

The SAS programs that produce the CDISC data need to undergo validation to confirm that they are correct and complete. This validation process includes double programming in addition to a logical code and dataset review. The SAS validation method of double programming requires two independent programmers to each write a SAS program with the goal of attaining identical results.

CHECKLIST ITEMS #11-13: CREATE SDTM SUPPORTING DOCUMENTATION

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
11.	**Create the Complex Algorithms document for SDTM (if needed)** This document is needed only if all applicable variables in the define file are not clearly described in the derivation of all applicable variables. SDTM should have few, if any, complex derived variables. Therefore, it is likely that this document is not needed for SDTM.	SAS Programmer	During SAS Programming of SDTM Domains	After database lock and all SDTM programs and define.xml are finalized.
12.	**Create the Study Data Reviewers Guide (SDRG)** Create the SDRG according to the SDRG template and samples This document is not finalized until after database lock. It is vital that every error and warning that will remain in the final Pinnacle 21 report is explained in the SDRG.	Data Manager or SAS Programmer	During SAS Programming of SDTM Domains	After database lock and all SDTM programs and define.xml are finalized.
13.	**Create the Annotated SDTM CRFs (aCRFs)** Create the aCRFs for SDTM according to the Work Instruction for Annotating Case Report Forms (CRFs) for Study Data Tabulation Model (SDTM) Datasets	SAS Programmer	At database lock when collection forms will not change and no new forms added	At database lock when collection forms will not change and no new forms can be added.

These items of the checklist guide the programmers with creating supporting documentation such as Complex Algorithms, cSDRG/nSDRG, and aCRFs.

The Complex Algorithms and cSDRG/nSDRG are not generated programmatically. Instead, each template and examples from other projects are provided to promote efficiency and consistency across projects. A SAS macro pulls CRF location information from the specifications to annotate the SDTM CRF according to aCRF guidelines. CRF pages are annotated to ensure the specs correspond to the final annotations. The SAS program drives the annotation colors, fonts, and sizes to automate efficiency and consistency.

CHECKLIST ITEMS #14-21: ADAM DATA AND SUPPORTING DOCUMENTATION

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
14.	**Develop ADaM Specifications** Create the ADaM dataset specifications document using the process outlined in the Work Instruction for Development of ADaM Data Specifications	Biostatistician or SAS Programmer	After SAP is finalized	After analysis is complete.
15.	**Program ADaM datasets** Create ADaM datasets according to the process outlined in the Work Instruction for Programming SDTM and ADaM Datasets Note that primary ADaM data-generating SAS programs will be included in the final submission.	SAS Programmer	As soon as SDTM datasets are available and ADaM specs are developed	Initial programs completed 3 weeks after programming start for application to TLFs; programs not considered final until analysis is final.
16.	**Create define.xml for ADaM** Create define.xml for ADaM according to the process outlined in the Work Instruction for Development of define.xml and Reviewer's Guides	SAS Programmer	Once ADaM dataset programs are developed	Initial define.xml completed with initial ADaM datasets for Pinnacle testing; define.xml is not considered final until analysis is final.
17.	**Convert SAS datasets to V5 XPT files** Convert SAS datasets to V5 XPT files using the sastoxpt.sas SAS program explained in the Work Instruction for Programming SDTM and ADaM Datasets Datasets must be converted to XPT prior to every iteration of Pinnacle 21.	SAS Programmer	Once ADaM dataset programs are developed	Every time SAS datasets are generated, XPT files must be created for Pinnacle testing. But XPT files are not final until ADaM datasets are final.
18.	**Run Pinnacle 21 to validate ADaM datasets and define.xml** Run Pinnacle 21 to validate ADaM datasets and define.xml and review the report. Address findings where possible according to the Work Instruction for Programming SDTM and ADaM Datasets A final Pinnacle 21 report after database lock should be saved to the project area with all findings documented in the ADRG.	SAS Programmer	After ADaM datasets are created	Final Pinnacle reports are run after analysis completion and all ADaM programs and define.xml are finalized.
19.	**Validate Dataset-Generation Programs through Double Programming** Independently validate ADaM programs through the double programming process outlined in the Work Instruction for Double Programming of CDISC datasets	Secondary, independent SAS Programmer	After ADaM datasets are created	Double programming is considered complete after database lock and all ADaM programs and define.xml are

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
				finalized.
20.	**Create the Complex Algorithms document for ADaM (if needed)** This document is needed only if the description of all applicable derivations in the define file are not clearly described.	SAS Programmer	During SAS Programming of ADaM datasets	After analysis completion and all ADaM programs and define.xml are finalized.
21.	**Create the Analysis Data Reviewers Guide (ADRG)** Create the ADRG according to the ADRG template and samples This document will not be finalized until after database lock. It is vital that every error, warning, and notice that will remain in the final Pinnacle 21 report is explained in the ADRG.	Biostatistician or SAS Programmer	During SAS Programming of ADaM datasets	After analysis completion and all ADaM programs and define.xml are finalized.

Although Checklist items 14-21 could be summarized by simply saying “repeat what you did for SDTM”, we found it beneficial to delineate the ADaM data steps separately. This helps to provide a visual of the true scope of a complete CDISC project, it enables the checklist to be split between separate SDTM and ADaM programming groups if needed, and the Target Start and Target Stop columns of the checklist can clearly indicate where the ADaM programming fits into the timeline.

Similar tools as described for SDTM are used to promote consistency across the ADaM programming. These include an ADaM specifications document and a series of generalized SAS macros and program templates. The ADaM programs are tested using the double programming method. Templates and examples of the Complex Algorithms and Analysis Data Reviewers Guide (ADRG) are provided.

CHECKLIST ITEMS #22-25: FORMATTING AND QA REVIEW

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
22.	**Bookmark pdf documents containing more than 5 pages** Add bookmarks to any document described above, or to be included in the final delivery package, that contains more than 5 pages according to the Work Instruction for Word Processing CDISC Documents	SAS Programmer	After database lock and analysis completion	Prior to QA review
23.	**Check file sizes** Ensure that files meet the criteria for size as explained in the FDA Study Data Technical Conformance Guidance. Datasets greater than 5 GB in size should be split into smaller datasets no larger than 5 GB. Sponsors should submit these smaller datasets, in addition to the larger non-split datasets. A clear explanation regarding how these datasets were split needs to be presented within the relevant data reviewer’s guide.	SAS Programmer	After database lock and analysis completion	Prior to QA review
24.	**QA Review**	QA Reviewer	Once all prior	Prior to final

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
	Send all documents to be included in the CDISC delivery to QA for review according to the Guide for QA Review of define.xml and Supporting Documents. This review should be scheduled several weeks ahead of time.		items are finalized and delivery package is considered final	delivery to the sponsor
25.	**Set up project folders to accommodate final CDISC deliverables** Ensure the correct folder structure is set up for the submission such that final deliverables will be organized according to Metadata Submission Guidelines, page 5.	SAS Programmer	After QA Review	Prior to final delivery to the sponsor

At this juncture the checklist items confirm that the files in the CDISC package are in the proper format, and ready for review by our QA staff. Some checklist reminders include: ensure are documents in pdf format with bookmarking if they are more than 5 pages; files must meet the criteria for size as explained in the FDA Study Data Technical Conformance Guidance; datasets greater than 5 GB in size are split into smaller datasets no larger than 5 GB. Sponsors should submit these smaller datasets, in addition to the larger non-split datasets. The split datasets are placed in a separate sub-directory labeled “split”. A clear explanation regarding how these datasets were split is presented within the relevant data reviewer’s guide.

All documents included in the CDISC delivery are then sent to QA for review. There is also a separate work instruction to guide the QA reviewers regarding what to look for, and what cannot be changed.

The checklist reminds the user to ensure the correct folder structure is set up for the submission such that final deliverables are organized according to Metadata Submission Guidelines.

CHECKLIST ITEM #26: CRITICAL QC STEPS PRIOR TO DELIVERY

	CDISC Task	Responsible Party/Parties	Target Start	Target Completion
26.	**Critical QC Steps Prior to Delivery** An independent reviewer (a staff member with minimal involvement in the steps 1 – 25) will review the CDISC Package according to Work Instruction for Final CDISC Package Delivery. Save the completed checklist in the project folders for documentation.	Independent Reviewer (e.g. Biostatistician, SAS Programmer, or Project Manager)	Prior to final delivery to the sponsor	Upon delivery to the sponsor

When the developer of the CDISC Package has determined that it is complete and ready for delivery, an independent reviewer (staff with minimal involvement with the development steps) will review the CDISC Package using a series of questions that indicate the critical QC steps that must be addressed prior to delivery. If the reviewer has any comments or concerns, they should work with the developer to address them until all critical QC steps can be signed off. The completed set of questions with sign off is saved in the project folders for documentation. These questions guide the independent reviewer through their assessment of the CDISC package. Some examples of good questions for the reviewer include:

- Are all SDTM datasets required for the study saved as V5 Transport (XPT) files in the SDTM datasets folder?
- Are any SDTM XPT files > 5 GB in size divided into sub-datasets and stored in a subfolder named “Split”?
- Is the define.xml for SDTM data, reviewed by QA per the Guide for QA Review of define.xml and Supporting Documents, included in the SDTM datasets folder?

- Is an XSL stylesheet included in the SDTM datasets folder?
- Are the primary SAS programs used to generate SDTM datasets, saved as ASCII files, included in the SDTM programs folder?
- Is the annotated CRF for SDTM data (acrf.pdf), reviewed by QA per the Guide for QA Review of define.xml and Supporting Documents and the Work Instruction for Annotation Case Report Forms (CRFs) for SDTM Data, included in the SDTM datasets folder?
- Is a Complex Algorithms document for SDTM, reviewed by QA per the Guide for QA Review of define.xml and Supporting Documents, included in the SDTM datasets folder? Note this may not be needed.
- Is a SDRG, reviewed by QA per the Guide for QA Review of define.xml and Supporting Documents, included in the SDTM datasets folder?
- Are all findings, if any, from the final Pinnacle 21 reports for SDTM datasets and define.xml, noted and explained in the SDRG?
- Are all documents referenced in define.xml for SDTM datasets included in the SDTM datasets folder?

Add similar questions for the ADaM data.

If the reviewer finds any problems such as missing files, hyperlinks that don't work, incorrect file size, etc., the project lead is notified and together the issues are resolved and the CDISC package is deemed ready for submission.

CONCLUSION

Organizing the demanding mission of preparing a CDISC-compliant data package into manageable discrete steps skillfully assesses the scope of the project, manages costs and timelines, assures no steps are missed, and facilitates efficiency and consistency within and across projects. Starting with a high-level checklist and maximizing each step with tools, trainings, work instructions, and other references is a practical approach to accomplish this goal. A comprehensive checklist systematically guides the user to follow the steps. It is uncomplicated and directs the user to specific information as needed, it delineates each key task using minimal steps to avoid user fatigue, and it is clearly understood by users for consistent execution. Consistency leads to accuracy. Consistency leads to efficiency. Consistency leads to approvals!

ACKNOWLEDGEMENTS

Thank you to our supervisors, editorial staff, and Westat senior leadership for your assistance and support in the writing of this paper.

DISCLAIMER: The contents of this paper are the work of the authors and do not necessarily represent the opinions, recommendations, or practices of Westat.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. You can email the author for a copy of the complete checklist described in this paper. Contact the author at:

Jennifer Fulton
JenniferFulton@westat.com

[1] Westat's SAS programming repository is proprietary and not available in any public forum. Contact any of the authors for information about how we can use our CDISC tools and experience to assist you.