PharmaSUG 2019 - Paper SS-337

Do-It-Yourself CDISC! A Case Study of Westat's Successful Implementation of CDISC Standards on a Fixed Budget

Rick Mitchell, Rachel Brown, Jennifer Fulton, Stephen Black, Marie Alexander, Westat

ABSTRACT

Do you find the prospect of implementing CDISC within the confines of a tight timeline and a fixed budget to be overwhelming? Westat, a premier research organization, offers the continuum of clinical trials support services, such as data management, data analysis, regulatory affairs, and site monitoring. Clinical research is dynamic, and it is imperative to comply with ever-changing federal guidances and regulations. This paper describes how Westat's statisticians, programmers, and data managers approached the implementation of CDISC standards on a fixed budget; discussing how we overcame challenges by creating tools and processes to improve efficiency, developing training programs, and reimagining how project teams collaborate to ensure data sets submitted to the FDA are compliant.

INTRODUCTION

Implementing CDISC processes across an organization is a complex undertaking that requires creative solutions and teamwork. A decade of working toward CDISC compliance for Westat probably mirrors the experience of many companies, but some of the issues encountered will differ depending on the organization. Because Westat is a full-service research organization that supports large-scale government surveys, exploratory and outreach projects, and large, multi-center clinical trials, it was vital to develop a CDISC implementation plan that builds staff skill to work across projects with CDISC requirements.

The variability of project work presented the first challenge to implementing CDISC. In the early days, the prospect of CDISC as a requirement for all IND submissions was looming, but FDA did not state a definitive date when the FDA would only accept CDISC compliant data sets. Staff members from the Westat Clinical Trials Practice attended CDISC-related conferences and training courses; but acquiring actual CDISC experience was another issue to overcome.

DIVING IN

Initially, Westat received a client request to develop CDISC Study Data Tabulation Model (SDTM) compliant data sets, for which analyses were already in progress. Our first challenge was that the data collection forms and database were not Clinical Data Acquisition Standards Harmonization (CDASH) compliant. The project started years earlier without the requirement of providing CDISC compliant deliverables. In addition, the study design was complex and the protocol required multiple custom domains. To ensure we could complete this assignment on time with credible data, we engaged a CDISC consultant to provide training, answer questions, and help develop domain specifications.

This first CDISC project was the foundation for subsequent processes. We developed SAS programming tools to create and check SDTM data sets and to support documentation and metadata such as define.xml. To respond to the increasing number of CDISC-based projects, it became apparent that substantial time and resources should be dedicated to share the new knowledge, experience, and tools with the Clinical Trials Practice data management and biostatistical staff.

INVESTING IN CDISC

A critical element of implementing a new regulated program such as CDISC is allocating the cost to individual projects. Westat made a business decision to invest in training the Westat Clinical Trials Practice staff supporting FDA regulated studies to deliver CDISC compliant data; while staff supporting non-regulated studies would continue to deliver data using existing standards.

The initial process of learning, training, and establishing standard operating procedures for implementing CDISC required staff time and an investment in external resources. This investment allowed project

teams to realize cost savings as the new processes streamlined work. Furthermore, project teams saw improved quality through process automations and built-in quality checks.

CREATING CUSTOM TOOLS

We generated a repository of custom tools that streamlined CDISC implementation processes and incorporated quality control checks. We tested tools across project teams with different users for their reliability. External programming consultants critiqued the tools and reviewed the data.

One such tool is a customized SDTM specifications template called the project's "SDTM specs" (Figure 1). This document uses the SDTM specifications downloaded directly from CDISC, which forms the foundation for any CDISC project. It includes the lists of domains and variables as provided in the specifications document from CDISC. We added worksheets for project teams to enter study-identifying information, the name and version of the CDISC model and Implementation Guide (IG), terminology versions, trial design information, trial summary parameters, and data sources for use in development of SDTM data sets, define.xml, and the aCRF. We update the document as the study progresses.

The document centralizes key study specifications in one location for project staff to access. Our SAS programs also pull information directly from the document. Changes to the specs can affect all SDTM data sets and supporting documents. The specs document drives the production of SDTM data sets and define.xml. All tools are based on the structure of this document and significant efficiencies are gained by using this standard tool.

Variable	Type	Label	Origin	CRFPage	Codelist Name	Code List	Role	Core	Description	Programmer Notes	CDISC Notes
STUDYID	Char	Study Identifier					Identifier	Req			Unique identifier fo
DOMAIN	Char	Domain Abbreviation			TA	C66734	Identifier	Req			Two-character ab
ARMCD	Char	Planned Arm Code					Topic	Req			ARMCD is limited t maximum length of kind of values that for a seven-period treatment and sep
ARM	Char	Description of Planned Arm					Synonym Qualifier	Req			Name given to an A
TAETORD	Num	Planned Order of Element within Arm					Timing	Req			Number that gives
ETCD	Char	Element Code					Record Qualifier	Req			ETCD (the compan character restricti not expected that
ELEMENT	Char	Description of Element					Synonym Qualifier	Perm			The name of the E
TABRANCH	Char	Branch					Rule	Exp			Condition subject r in this Arm (e.g., "
TATRANS	Char	Transition Rule					Rule	Exp			If the trial design a sequence, then th Element sequence
EPOCH	Char	Epoch			EPOCH	C99079	Timing	Req			Name of the Trial E
STUDYID	Char	Study Identifier					Identifier	Req			Unique identifier fo
DOMAIN	Char	Domain Abbreviation			TE	C66734	Identifier	Req			Two-character ab
ETCD	Char	Element Code					Topic	Req			ETCD (the compan

Figure 1. SDTM Specifications Template – A sample of the "Variables" list from the generic SDTM Specs document. Project staff fill in study-specific information, such as Origin and Description.

SAS TOOLS TO GENERATE DATA

Next we developed SAS macros based on our SDTM specifications document for use with any CDISC project. The simplest of these tools is a program that creates the Trial Summary (TS) domain (Figure 2) using study-specific Trial Summary parameter information entered into the SDTM specs. The project management staff provide information needed for TS. Stipulating the information in the SDTM specs document, rather than in a SAS program, makes it easier to gather input on these parameters from project team members. An additional benefit to this approach is that our SAS programmers don't need to search for the information while developing the SDTM programs.

TSPARMCD	TSPARM	TSSEQ	TSVCDREF	TSVCDVER	TSVAL	TSVALCD	Core	Source
ACTSUB	Actual Number of Subjects	1					Required	
ADAPT	Adaptive Design	1	CDISC				Required	
ADDON	Added on to Existing Treatments	1	CDISC				Required	
AGEMAX	Planned Maximum Age of Subjects	1	ISO 8601				Required	
AGEMIN	Planned Minimum Age of Subjects	1	ISO 8601				Required	
BIOSPRET	Biospecimen Retention Contains DNA	1					If Applicable	
BLNDSTAT	Blinded Status SDTM Dataset Generation; Blinded Status at SDTM Dataset Generation	1					If Applicable	
CITNSTDY	Citation Used in Study	1					If Applicable	
CMSPSTAT	Commercial Sponsor Status	1					If Applicable	
COMPTRT	Comparative Treatment Name	1					If Applicable	
CRMDUR	Confirmed Response Minimum Duration	1	ISO 8601				If Applicable	

Figure 2. Trial Summary – A sample of the "TS" section from the SDTM specs document; Study-specific information such as TSVAL and Source would be filled in by project staff.

We developed another tool that uses the Trial Design section (Figure 3) of the SDTM specs document to establish Trial Arms (TA), Trial Elements (TE), and Trial Visits (TV) domains. Identifying this information before the programming phase allows project managers to provide input on ELEMENT and EPOCH definitions for a clinical trial. In the past, SAS programmers had to define important trial parameters, many of which required input from other team members.

ArmCode	Arm	Element Number	ElementCode	Element Name	RuleStartElement	RuleEndElement	Element Duration	Epoch Number	EpochName
ARM1	Arm 1 Name	1	SCREEN	Screening	Informed consent	4 weeks after start of element	P28D	1	SCREENING
ARM1	Arm 1 Name	2	TREAT	Treatment	First dose of Drug XYZ	2 weeks after start of element	P2W	2	TREATMENT
ARM1	Arm 1 Name	3	FOLLOWUP	Follow-Up	28 days after last dose of Drug XYZ	6 months after start of element	P6M	3	FOLLOW-UP
ARM2	Arm 2 Name	1	SCREEN	Screening	Informed consent	4 weeks after start of element	P28D	1	SCREENING
ARM2	Arm 2 Name	2	TREAT	Treatment	First dose of Drug XYZ	2 weeks after start of element	P2W	2	TREATMENT
ARM2	Arm 2 Name	3	FOLLOWUP	Follow-Up	28 days after last dose of Drug XYZ	6 months after start of element	P6M	3	FOLLOW-UP
ARM3	Arm 3 Name	1	SCREEN	Screening	Informed consent	4 weeks after start of element	P28D	1	SCREENING
ARM3	Arm 3 Name	2	TREAT	Treatment	First dose of Drug XYZ	2 weeks after start of element	P2W	2	TREATMENT
ARM3	Arm 3 Name	3	FOLLOWUP	Follow-Up	28 days after last dose of Drug XYZ	6 months after start of element	P6M	3	FOLLOW-UP

Figure 3. Design– A sample of the "Design" section from the SDTM Specs document; all text would be updated to describe a specific study.

SAS TOOLS TO CHECK AND PROCESS THE DATA

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

Another macro which is the longest of our CDISC SAS macros, creates variables common to all domains, such as STUDYID and USUBJID, reducing repetitive code across programs. The macro 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 a number of CDISC 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.

Publicly available software, Pinnacle 21, is an industry standard to assess the compliance to model specifications of CDISC data. Pinnacle 21 replicates many of Westat's macro checks, but it is helpful to address these issues during the initial development of each SAS program.

Westat runs a SAS macro after all SDTM domains are generated. This 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. It prints problems found in response to all these checks to the listing (.lst) file of the data program that called it.

SAS TOOLS TO DEVELOP CDISC SUPPORTING DOCUMENTATION AND METADATA

Westat programmers wrote a SAS macro to create the study's define.xml. Any time new data are available or specs are updated, we run the macro to update define.xml. This allows us to create define.xml early in the project, include it in Pinnacle 21 checks of the domains, and to conduct a more thorough check of SDTM data and define.xml early in the process and throughout data collection and programming. This macro frees the SAS programming teams from the burden of maintaining XML code. When revisions are required in define.xml, the SDTM Specs document is updated rather than in XML code.

Another macro pulls Case Report Form (CRF) location information from the specs to annotate the SDTM CRF according to aCRF guidelines. Corresponding 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. Westat's Quality Assurance department performs a QC check of the aCRFs as well.

There was significant time invested to generate this collection of CDISC tools. These tools evolved, project by project, to streamline processes and improve quality control. They assist new CDISC programmers by facilitating input from other project team members with information for the SDTM data set package and alerting them to potential issues throughout the development process. The tools reveal potential issues early in the programing process, for quick resolution. Over time, these added efficiencies result in substantial cost savings.

GETTING EVERYONE ON BOARD

A critical next step in the process of CDISC implementation was sharing our experience with the Westat Clinical Practice project staff. Expanding the CDISC expertise within our organization enabled us to respond to the increased volume of CDISC implementation requests from other Westat projects with regulated studies. We developed a CDISC training program on many 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 Westat staff experience, online resources such as the CDISC webinars, and consultant input.

We generated detailed work instructions, which give project teams an organized set of references when initiating a CDISC project. 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 Westat or online.

STRENGTHENING THE CDISC SAS PROGRAMMING TEAM

The complexity of the clinical trial determines the number of CDISC programmers assigned to a project. Some clinical trials may require only two SAS programmers; one to lead the programming effort and another to independently validate programs. A study needing an analysis report will require both SDTM and ADaM (Analysis Data Model) data. It is not always feasible to have separate programming teams

where one group specializes in ADaM and the other group specializes in SDTM, especially for a clinical trial that has a relatively small number of patients or scope of data to be collected. Instead, Westat programmers are involved in their projects from start to finish. ADaM data builds on SDTM data, and both require a comprehensive understanding of the study protocol and Statistical Analysis Plan. A programmer who developed SDTM data for a study has the knowledge to create ADaM data for that same study. Projects benefit from consistent, dedicated project staff throughout the trial.

CREATING NEW CROSS-DEPARTMENT COLLABORATIONS

Prior to CDISC, the Westat SAS programmers often had limited input into the CRF design or database structure. Data management staff developed the database prior to and independent of the SAS programming team's input related to production of the final data sets and reports. The SAS programmers received CDISC training first since they were ultimately responsible for the final deliverables. On the first CDISC projects the databases were already built, hence our SAS programmers created CDISC data sets from legacy data. The more compliant processes from protocol development, to CRF design and database build, the more efficiently the SAS programmers can produce the end deliverables (data sets, metadata, and supporting documentation that make up the CDISC package).

Project managers need a high-level understanding of the CDISC project flow, to provide input in the trial design domains. They also need to understand how CDISC impacts the timeline, scope, and costs of a project. The way these functional areas approach the next project, both independently and collaboratively, had to change with the new era of CDISC.

Westat created CDISC working groups within project management, data management, biostatistics, and SAS programming functional areas. These groups established work instructions, training programs, outreach plans, and tools related to the CDISC processes. A "CDISC Summit" team meets frequently to ensure processes effectively facilitate CDISC work products.

CONCLUSION

Westat's implementation of CDISC has evolved over the last decade. Establishing the cross-functional area collaborations to develop tools and streamline processes resulted in quality improvements for each phase of CDISC implementation. Deliverables receive more vigilant quality control to ensure CDISC compliance.

Westat's training materials, instructions, and tools are reviewed continually to stay abreast of new versions of models released by CDISC. Westat provides services such as data management and biostatistics across a variety of types of clinical trials and indications, which requires the ability to track and incorporate CDISC specifications for any therapeutic area. There are fewer changes today than in the past, but continuous quality improvement will remain a function of Westat's CDISC world.

Westat's successful implementation of CDISC has been both challenging and rewarding. Team members across the functional areas embraced the challenge of implementing CDISC. Westat's CDISC capabilities strengthened, so we can better meet our client needs for delivering CDISC compliant data sets. For organizations beginning to learn the CDISC process, there is light at the end of the tunnel because you too can Do-It-Yourself!

ACKNOWLEDGEMENTS

Thank you to our supervisors, editorial staff, and Westat management 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. Contact the authors at:

Rick Mitchell
RickMitchell@westat.com

Rachel Brown
RachelBrown@westat.com

Jennifer Fulton
JenniferFulton@westat.com

Stephen Black
StephenBlack@westat.com

Marie Alexander
MarieAlexander@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.