Paper 202-2011

State of the Union: The Crossroads of CDISC Standards and SAS'® Supporting Role

Chris Decker, d-Wise Technologies, Raleigh, NC, USA

ABSTRACT

The Clinical Data Interchange Standards Consortium (CDISC) began in the late 90's with the goal of developing clinical research standards across the industry to optimize the drug development lifecycle and improve the regulatory review process . Over the last decade, CDISC has developed a myriad of models in the hope of reaching their goal. As with any new industry initiative, CDISC has gone through growing pains as the models have been met with mixed reviews by both industry and regulatory agencies. During this same timeframe, SAS has made a concerted effort to develop tools to support those same clinical research standards. This paper will discuss the current state of the CDISC standards describing the current use and adoption of the standards, an update on the individual CDISC models, and the challenges that lay ahead. In addition, the paper will describe the current SAS solutions that support CDISC standards and discuss the advantages and limitations of these solutions.

INTRODUCTION

The Clinical Data Interchange Standards Consortium (CDISC) began in the late 90's with the goal of developing clinical research standards across the industry. CDISC was incorporated in 2000 and began developing the first CDISC models, SDTM for data collection and submission, and ODM for the transfer of data between systems. Over the last decade the CDISC organization has developed a myriad of models to support the various needs of industry throughout the clinical data lifecycle from development of the protocol through submission. CDISC has gone through a maturity process as companies have implemented the CDISC standards. As with any new initiative, CDISC has gone through an up and down adoption curve during this timeframe and the models have been met with mixed reviews by both industry and the FDA. As part of this process, CDISC has worked with the FDA and industry to ensure the needs of both customers are met. This paper will provide a short CDISC history lesson, the current CDISC adoption by the FDA, what each CDISC team has delivered recently and is currently being worked on, and the challenges industry still faces in adopting standards.

During this same timeframe, SAS has made a concerted effort to develop tools to support the clinical research standards. It started with generic tools in the late 90's for clinical data warehousing and reporting that had limited success as they didn't quite meet the industry's needs and industry wasn't ready for point and click tools. In the early 2000's they began developing CDISC tools, but with no real aligned goals. Over the last three years, SAS has focused on delivering solutions with the goal of helping companies work specifically with their clinical data standards. The paper will describe the current SAS solutions that support CDISC standards and discuss the advantages and limitations of these solutions.

CDISC CROSSROADS

Over the last few years, CDISC has been at a crossroads with the regards to the adoption of standards and what the next steps should be moving forward. While a decade seems like a long time, developing and validating industry wide standards is an overwhelming challenge which means the organization is still in its infancy stages and is going through the typical growing pains of any standards organization. This section will provide a short history of CDISC and the current state of the standards.

CDISC HISTORY LESSON

Over the last 25 years, the adoption of data standards for the collection and transfer of clinical data has been a slow and challenging process. Within under industries, such as finance, the underlying data is somewhat static and the associated standards and tools are easier to implement for supporting the flow of the information. However, the clinical research arena provides a unique challenge because of the continuous change within both the medical and statistical science. A specific disease is not studied the same way it was five years ago, new endpoints are being created every day, and innovative statistical methodologies are being developed to support complex analyses. The ever changing process makes the development, implementation, and enforcement of data standards very complex and challenging.

In the beginning, the focus was on real time data collected in the health care environment such as hospitals and insurance companies. The standards used for this exercise are known as Health Level 7 (HL7), a group that was initiated over 25 years ago. However, HL7 was designed to meet the need to collect individual patient data points in the health care environment and could not easily be translated to clinical trials.

In the late 90's a group of innovative individuals got together to see if they could tackle the monumental challenge of defining a data standard across industry to support the clinical data lifecycle within clinical trials. From this collaboration, the Clinical Data Interchange Standards Consortium (CDISC) was formed with the mission "to develop and support global, platform-independent data standards that enable information system interoperability to improve medical research".

The initial CDISC focus was to support the submission of standard data to the FDA. With this goal in mind, the Study Data Tabulation Model (SDTM) and the Operational Data Model (ODM) were developed first. ODM was designed to provide a specification for the transfer of data between systems. While it was very well defined, and is probably still the only machine readable specification within CDISC, it was not initially implemented by many companies.

As the scope of CDISC grew over the years, they realized the need to support the flow of data through the entire lifecycle. Over the last decade a number of models have been developed within CDISC to support the needs of clinical trial data. Table 1 below contains a list of the most commonly used models CDISC has developed over the years, and the purpose the serve.

Model/Standard	**Purpose**
Operational Data Model (ODM)	XML specification supporting interchange of data, metadata or updates of both between clinical systems
Clinical Data Acquisition Standards Harmonization (CDASH)	Data model for a core set of global data collection fields (element name, definition, metadata)
Study Data Tabulation Model (SDTM)	Data model supporting the submission of data to the FDA including standard domains, variables, and rules
Analysis Dataset Models (ADaM)	Data model closely related to SDTM to support the statistical reviewer by providing data and metadata that is analysis ready
Define.xml	XML Specification to contain the metadata associated with a clinical study for submission
Standards for the Exchange of Non-clinical Data (SEND)	Data model extending SDTM to support the submission of animal toxicity studies
Protocol Representation Model (PRM)	Metadata model focused on the characteristics of a study and the definition and association of activities within the protocols, including "arms" and "epochs"
Terminology	Standard list of terms across all the CDISC data models

Table 1. Summary of Relevant CDISC Models

The standards described in this table have varying levels of maturity. The SDTM and define.xml are probably the most widely used CDISC components and have been referenced in various FDA documents. While CDISC has done an excellent job of laying the foundation for clinical data standards, changing the large and extremely lethargic clinical research industry is a daunting task. It has taken ten years to get organizations on board with this effort and moving in the same direction.

CDISC STATE OF THE UNION

Over the last 4 years CDISC has been through significant twists and turns that have led to some enlightening discoveries and what appears to be a very promising path forward.

In 2007, CDISC was still in its infancy stages which can be expected from a new standards organization trying to change a process that was decades old. They were struggling to gain adoption and show a true proof of value despite support from FDA and large companies within the pharmaceutical industry. The SDTM model was the most mature model and companies were attempting to adopt it within their organization. ODM was beginning to be used for data transfers between systems and the development of CDASH for data collection was underway.

While companies were making their best efforts to implement the standards, there was no focused or consistent message from the FDA. Most companies were using the models more at the end of the process instead of operationally, which in turn was adding significantly to their underling cost. This issue was compounded in 2007 when the FDA announced plans to move towards the HL7 standard, indicated the current incarnation of the CDISC standards might change drastically, and the SAS transport format delivery mechanism would be gone by 2013. This announcement had a strong ripple effect throughout the industry throwing both pharmaceutical companies and CDISC into a bit of chaos similar to yelling 'fire' in a crowded theater. Between this announcement and the continued absence of limited and consistent communication from the FDA, the industry was like a deer in headlights. Why should we spend enormous amount of money and resource implementing standards if it's going to change? What is HL7 and how am I going to begin to implement it? What does the FDA want?

Over the next two years, industry as well as regulatory agencies struggled to find their way. This came to a head at the CDISC Interchange in 2009. In front of a panel of regulatory leaders from the FDA, the audience begged for someone to give them a clear direction. What people didn't know was that these messages were now being heard loud and clear at the FDA and plans were already in place to address these issues.

In the summer of 2009, the FDA CDER division formed The Computational Science Center (CSC), a desperately needed infrastructure for CDER's scientific community with the goal of supporting a number of ongoing efforts in pre-market development, modernization of drug review, post-market safety, and drug quality. The stated mission of the newly formed group was to provide CDER reviewers a more aligned and automated method for completing reviews and more transparency in decision making and documentation and focused on key projects which included the adoption and enhancement of CDISC standards and expanding the use of electronic review tools. In addition, CBER, a much smaller division but one that was facing the same challenges, also began putting a structure in place to prepare and receive CDISC standard submissions. The clear message at this conference was that the announcement made in 2007 was on hold indefinitely and the training, implementation and enhancement of the CDISC standards was a priority. More importantly, both CDER and CBER were now to have dedicated functional resources to drive this effort.

The tone at the CDISC Interchange just one year later was a fundamental shift from hostility and confusion to progress and hope. While there are still challenges and the process will continue to move slowly, the general consensus was that progress is being made and for the first time, both industry and the FDA were moving in the same direction. Within that meeting, optimistic quotes were made by FDA leaders.

- Theresa Mullin, Director, Office of Planning and Informatics within CDER said: *"FDA is committed to using CDISC standards for the foreseeable future"*.
- Amy Malla, Project Manager responsible for implementing CDISC standards and tools within CBER said they are putting all their focus on accepting and using the CDISC standards and that sponsors must *"communicate with reviewers, follow SDTM, submit define.xml and focus on quality data".* She also announced that CBER was accepting SDTM as of May 15, 2010 and will begin accepting ADaM as of December 15, 2010 after CBER reviewers have had an opportunity to be trained.
- Chuck Cooper, lead within the CDER Computational Sciences Center, said CSC is *"moving forward with assessing current submissions for CDISC compliance, developing checklists for sponsors, and implementing training across all review divisions"*

In addition to these clear directives from FDA leaders, other tangible activities have been initiated to show continued support for the adoption of CDISC standards. In the fall of 2010, FDA issued and awarded a contract to convert over 120 legacy studies to the CDISC SDTM and ADAM standard to support cross trial analysis. FDA also issued a solicitation in December of 2010 to build a forward thinking data warehouse that can support the loading of CDISC data and make it accessible to reviewers within the FDA and researchers across many different organizations. FDA and CDISC have initiated a multiyear training plan to train all reviewer divisions within the agency on the use of CDISC standards, Finally, in 2010, the first of hopefully many Computational Science Center conference was held allowing industry and regulatory agencies to come together to collaborate on standards, share best practices, and discuss implementation challenges.

CDISC TECHNICAL UPDATE

With renewed vigor CDISC volunteers are tackling and delivering new and improved updates to the standards to help support the industry and FDA. Table 2 below describes a subset of the activities currently ongoing within each standard.

Model/Standard	**Update**
CDASH	• Version 1.1 released including new domains, implementation recommendations, and best practices • Implementation guide under development • Machine readable xml specification being developed in collaboration with the XML Technologies team • Device and therapeutic specific CRFs are under development

Model/Standard	Update
Terminology	• New terminology package release in January of 2011 with a focus on SEND data elements • Three new therapeutic area standards out for public review including Alzheimer's disease, cardiovascular, and Parkinson's disease
SDTM	• Metadata Submission Guidelines released • Collaborating with Terminology and CDASH to ensure new therapeutic specific data elements are aligned
ADaM	• Released ADaM Validation checks 1.0 and 1.1 • New ADaM structures ADAE and ADTTE out for public comment • Examples document, metadata guidelines, and new release of ADaM validation checks under development
Define.xml	• Released white paper regarding validation of the define.xml • Working on development of define.xml version 2.0 including support for value level metadata, ADaM analysis results metadata, and ?????

Table 2. Technical Update of the CDISC Models

In addition to the ongoing work of the teams, a number of important activities are ongoing within CDISC and their partner organizations. In 2009, CDISC initiated the CDISC SHARE project, the CDISC Shared Health and Clinical Research Electronic Library, whose goal is to provide a well managed and machine readable mechanism for managing data elements across the portfolio of CDISC standards. This project is under development with the hope of a beta release in 2011. This will provide a single source of the truth or standards and accelerate the development of standards.

Finally, the renewed focus from the FDA has led to significant increase in collaboration activities between CDISC and the FDA. In March of this year, three full day meetings were held to align the needs of the FDA with the strategic and technical goals of CDISC. These ongoing communications can only help to support and improve the future of the standards.

CDISC FUTURE CHALLENGES

Even though a number of optimistic activities have happened over the last two years with regards to CDISC, FDA and industry, challenges still exist with adopting the standards. As mentioned earlier, one of the biggest challenges in the adoption process is the strong reluctance to change. "If it isn't broken then don't fix it" is the comfortable quote people use and reluctance to change is a common behavior among most individuals.

Unfortunately, as companies begin to adopt data standards, they aren't defining it as an integral part of their operational process but after the fact as a necessary evil of submitting the data to the FDA. This issue began due to the widespread adoption of CDISC SDTM, a model defined for the raw data in a submission format. By defining this model first, CDISC started their standards development smack dab in the middle of the process, the creation of study data for submission in the model. At the time this made sense because the most important customer of the data was the FDA. Unfortunately, this creates challenges because it isn't the way data is collected nor is it the way data is analyzed within the drug development process. As companies try to adopt the SDTM standard, they are very reluctant to change their processes and internal operational data standard. Therefore, if the standards are not integrated into their process and initiated much further upstream during study design and data collection, SDTM ends up being a very expensive and time consuming exercise at the end of a clinical trial.

However, companies are slowly starting to modify their internal processes to better support the standard. With the increased use of the ODM model for data transfer as well as the introduction and swift implementation of CDASH for data collection, the adoption of the standards should increase rapidly over the next decade. The data standards will now be used at beginning of the clinical trial, the data collection step, and more easily move through the data transformation and analysis steps.

While the current standards have limitations the industry must continue to work towards adopting the standards in their process even if it doesn't lead to immediate efficiencies in the short term. By jumping full throttle into the standards we can learn where the gaps are and work harder to close those gaps. This is easy to recommend in theory but leads to challenges as companies are under more pressure every day to get drugs submitted faster.

In the future, standards can be adopted more smoothly if the industry works harder at incorporating them earlier in the process. As CDASH matures we can work on collecting the data in a standard and thus make everything else downstream much easier as CDISC works towards tighter alignment of the standards. The standards can even go back further to the development of the protocol with the CDISC release of the Protocol Representation 1.0 Model which provides a standard for collecting metadata about a Protocol. By iteratively following this lifecycle of clinical data standards in the future (Figure 1) and improving the steps as we go along, standards will become an integral part of the process instead of a necessary evil.

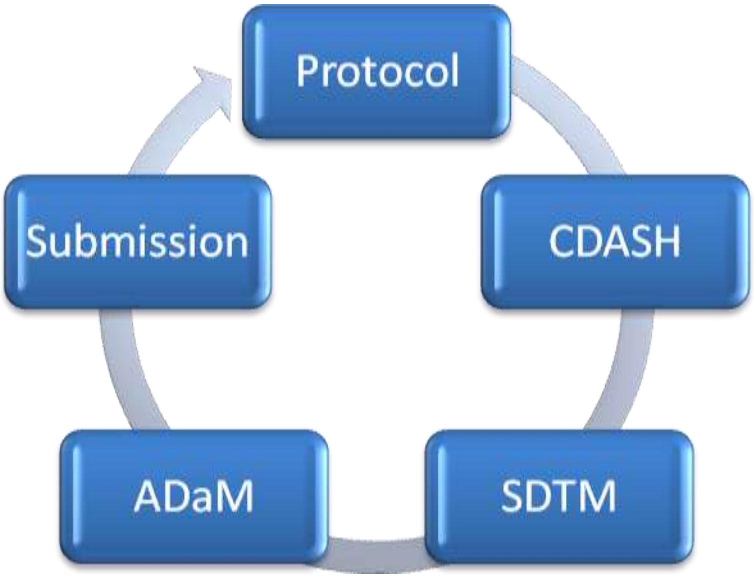


Figure 1. CDISC Standards Lifecycle

SAS AND CDISC PLAYING TOGETHER?

HISTORY LESSON

SAS has always been a core component to the data processing and analysis of clinical data. When the clinical research industry, primarily including non computer science users, began collecting clinical trial data electronically, people needed a programming language that was easy to use and could perform high end analytics. SAS was an obvious fit, and this began the long and dominant use of SAS for processing clinical data. In 1999, the FDA identified the SAS V5 transport file as the mechanism for delivering data to the FDA. The FDA selected the format because it was an open format which means the structure was in the public domain and could be consumed by other technologies.

In the mid 1990's, SAS began looking at building industry specific solutions and the pharmaceutical industry was an obvious target. SAS came out with two products to support data warehousing and clinical reporting. PH.DataWare was built on top of Warehouse Administrator and was SAS' first attempt at building an ETL specific tool for data transformations. PH.Clinical was built as a SAS report generation tool as well as a clinical tool for viewing and exploring clinical data. Both products had some success but were not widely adopted. The ETL solution was too rigid and did not provide enough flexibility for the uniqueness of clinical data, a challenge commonly seen in ETL solutions. The PH.Clinical solution took too much programming out of the hands of hard core SAS programmers and made them work with point and click interfaces. Both products were slowly phased out in the mid 2000's.

PROC CDISC was developed in the early 2000's to help support the new emerging CDISC standard. It attempted to support the ODM standard within CDISC and provided tools to move data back and forth between SAS and the ODM xml specification. While it provided some basic capabilities for clinical programmers it was not fully supported by SAS and never had a production release. In addition, because the use of ODM was somewhat limited early on, there was no real demand for this capability.

The biggest challenge SAS faced in implementing solutions to support clinical data standards was the Clinical SAS programmer. Programmers within this industry have a long history of using BASE SAS for creating very elaborate SAS frameworks to deal with clinical data standards. It was very difficult for SAS to develop solutions that met the flexible needs of clinical data and would win over the entrenched SAS programmers.

At the beginning of 2009 SAS put a new focus on developing solutions to support clinical data standards and transformations. After years of attempting to develop tools to support clinical data standards, SAS has developed what appears to be a robust framework to support the management of clinical data standards including the SAS Clinical Standards Toolkit and the SAS Clinical Data Integration Solution.

SAS CLINICAL STANDARDS TOOLKIT

The SAS Clinical Standards Toolkit (CST) is a framework of SAS macros, metadata, and configuration files including a representation of the SDTM metadata, a large set of validation checks, and the ability to create define.xml for submissions. This section will provide an introduction to the Toolkit including the installation components, running validation checks, creating define.xml, and some of the challenges and limitations we have encountered along the way.

What is in the Toolkit?

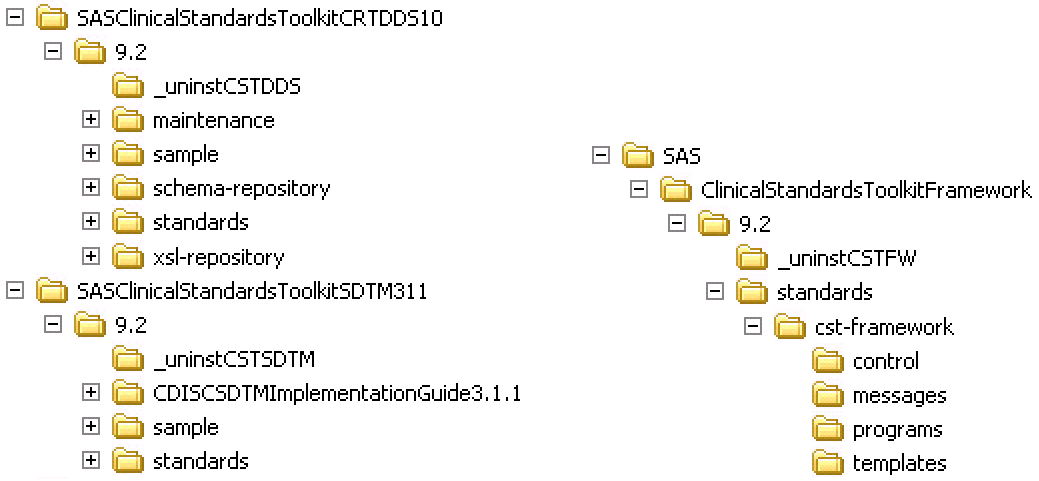
After CST is installed, the user is left with a multitude of files and folders scattered throughout their system. This is the first hint that a significant learning curve exists for understanding the varying components of CST. The installation puts parts of the framework into three main areas.

Framework

The folder in sasroot that corresponds to the installation of the CST is called “cstframework” and contains a subfolder sasmacro. This folder contains CST framework macros which become part of the SASAUTOS autocall library that is assigned at startup. The naming convention of these macros reflects their role in the CST and all begin with “cst”.

- “cst_” reflects macros used for general framework purposes
- “cstcheck_” reflects macros used for validation checks;
- “cstutil_” reflects utility macro

Within SAS 9.2, the sasroot directory usually sits within a directory called SASFoundation. After CST is installed, several folders will be created alongside SASFoundation whose names begin with “SASClinicalStandardsToolkit.” The remainder of the folder name includes the name of a standard as shown in Display 1. For the most part, a user will not touch this area as it includes standard framework macros that are used throughout other CST programs.



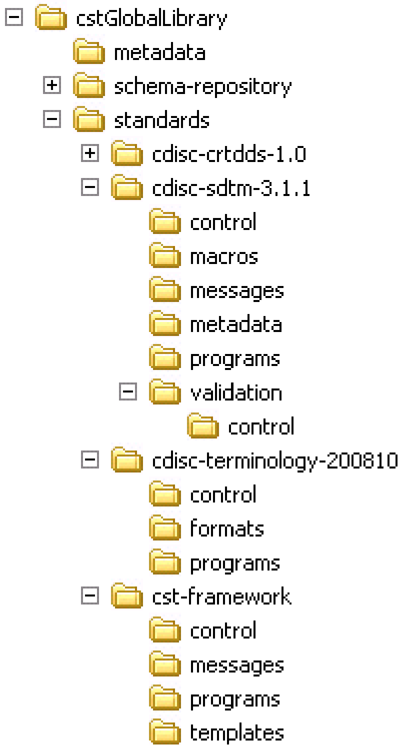
```
⊟ SASClinicalStandardsToolkitCRTDDS10
    ⊟ 9.2
          _uninstCSTDDS
        ⊞ maintenance
        ⊞ sample
        ⊞ schema-repository
        ⊞ standards
        ⊞ xsl-repository
⊟ SASClinicalStandardsToolkitSDTM311
    ⊟ 9.2
          _uninstCSTSDTM
        ⊞ CDISCSDTMImplementationGuide3.1.1
        ⊞ sample
        ⊞ standards

⊟ SAS
    ⊟ ClinicalStandardsToolkitFramework
        ⊟ 9.2
              _uninstCSTFW
            ⊟ standards
                ⊟ cst-framework
                      control
                      messages
                      programs
                      templates
```

Display 1. CST Framework Directory Structure

Global Library

The bulk of work done in a CST process will take place inside the CST root directory. By default SAS will install this on the C drive in the folder C:\cstGlobalLibrary. This folder structure contains each of the global standards that are ‘registered’ in the Toolkit. It will initially include the default standards from CDISC but in practice, will also contain a customer’s implementation of the standards. For example, a customer might have therapeutic, compound, or study specific standards that would be registered with the Toolkit. The cstGlobalLibrary folder contains a metadata sub-folder that contains overall metadata about each standard that is registered and a standards sub-folder that contains directories for each specific standard (e.g. SDTM 3.1.2) containing standard specific metadata, validation files, terminology, and other files and parameters specific to that standard. Display 2 below provides an example of this structure.



```
⊟ cstGlobalLibrary
    metadata
  ⊞ schema-repository
  ⊟ standards
    ⊞ cdisc-crtdds-1.0
    ⊟ cdisc-sdtm-3.1.1
        control
        macros
        messages
        metadata
        programs
      ⊟ validation
          control
    ⊟ cdisc-terminology-200810
        control
        formats
        programs
    ⊟ cst-framework
        control
        messages
        programs
        templates
```

Display 2. CST Global Library Directory Structure

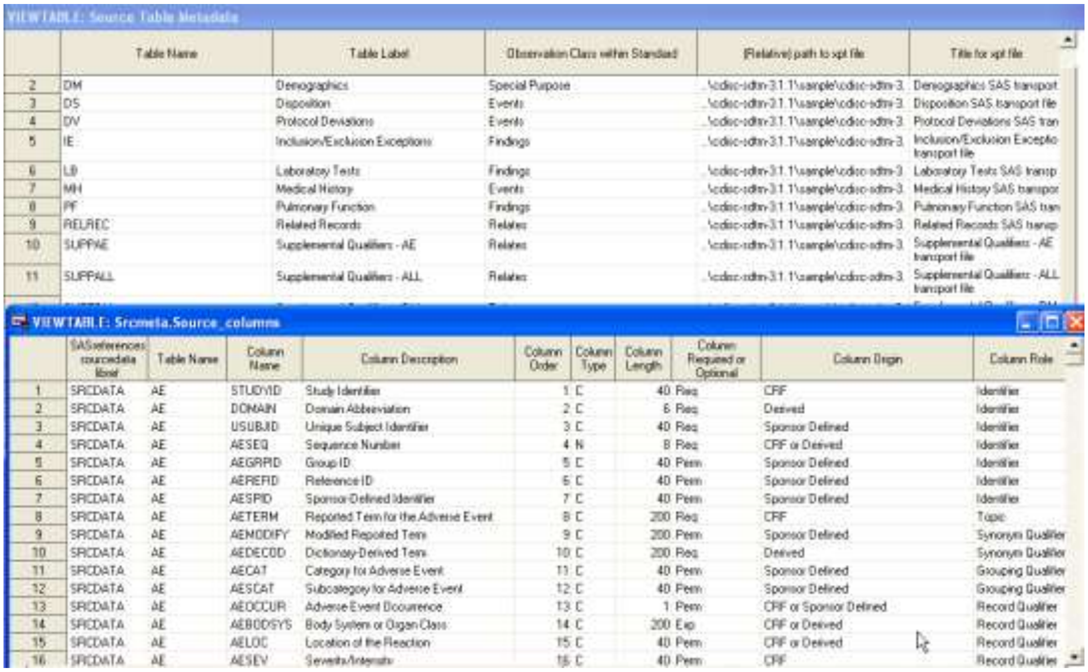
Study Library

The final component is one that is not initially available after installation but which needs to be created to work with an individual clinical trial. While the global library contains the default standards, a user needs to create specific folders and files that contain information for the specific study. This information can be copied from the existing global standards and modified to meet the needs of the specific study. While there are boatloads of configuration data sets for developing either a global standard or implementing an individual study, let's review a few of the more important data sets (Table 3).

Data Set	Location	Purpose
SASREFERENCES	*study*/control	Contains all the libname and filename parameters required for the Toolkit programs to run including options such as the location of the data, source and reference metadata, and the location and order of the controlled terminology.
VALIDATION_CONTROL	*study*/control	Contains the full list of validation checks for a specific global standard or a subset of those checks at a study level
TABLES	*study*/metadata	Contains table level metadata for the global standard or a specific study. This includes the metadata required by the standard and is used for both validation and creation of define.xml
COLUMNS	*study*/columns	Contains column level metadata for the global standard or a specific study. This includes the metadata required by the standard and is used for both validation and creation of define.xml

Table 3. Summary of Important Data Sets within CST

Display 3 below provides a sample of the Toolkit TABLES and COLUMNS data sets containing both standard SAS metadata as well as the specific metadata required within SDTM.



VIEWTABLE: Source Table Metadata

	Table Name	Table Label	Observation Class within Standard	(Relative) path to xpt file	Title for xpt file
2	DM	Demographics	Special Purpose	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Demographics SAS transport
3	DS	Disposition	Events	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Disposition SAS transport file
4	DV	Protocol Deviations	Events	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Protocol Deviations SAS tran
5	IE	Inclusion/Exclusion Exceptions	Findings	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Inclusion/Exclusion Exceptio transport file
6	LB	Laboratory Tests	Findings	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Laboratory Tests SAS transp
7	MH	Medical History	Events	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Medical History SAS transpor
8	PF	Pulmonary Function	Findings	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Pulmonary Function SAS tran
9	RELREC	Related Records	Relates	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Related Records SAS transp
10	SUPPAE	Supplemental Qualifiers - AE	Relates	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Supplemental Qualifiers - AE transport file
11	SUPPALL	Supplemental Qualifiers - ALL	Relates	..\cdisc-sdtm-3.1.1\sample\cdisc-sdtm-3.	Supplemental Qualifiers - ALL transport file

VIEWTABLE: Srcmeta.Source_columns

	SASreferences sourcedata libref	Table Name	Column Name	Column Description	Column Order	Column Type	Column Length	Column Required or Optional	Column Origin	Column Role
1	SRCDATA	AE	STUDYID	Study Identifier	1	C	40	Req	CRF	Identifier
2	SRCDATA	AE	DOMAIN	Domain Abbreviation	2	C	6	Req	Derived	Identifier
3	SRCDATA	AE	USUBJID	Unique Subject Identifier	3	C	40	Req	Sponsor Defined	Identifier
4	SRCDATA	AE	AESEQ	Sequence Number	4	N	8	Req	CRF or Derived	Identifier
5	SRCDATA	AE	AEGRPID	Group ID	5	C	40	Perm	Sponsor Defined	Identifier
6	SRCDATA	AE	AEREFID	Reference ID	6	C	40	Perm	Sponsor Defined	Identifier
7	SRCDATA	AE	AESPID	Sponsor-Defined Identifier	7	C	40	Perm	Sponsor Defined	Identifier
8	SRCDATA	AE	AETERM	Reported Term for the Adverse Event	8	C	200	Req	CRF	Topic
9	SRCDATA	AE	AEMODIFY	Modified Reported Term	9	C	200	Perm	Sponsor Defined	Synonym Qualifier
10	SRCDATA	AE	AEDECOD	Dictionary-Derived Term	10	C	200	Req	Derived	Synonym Qualifier
11	SRCDATA	AE	AECAT	Category for Adverse Event	11	C	40	Perm	Sponsor Defined	Grouping Qualifier
12	SRCDATA	AE	AESCAT	Subcategory for Adverse Event	12	C	40	Perm	Sponsor Defined	Grouping Qualifier
13	SRCDATA	AE	AEOCCUR	Adverse Event Occurrence	13	C	1	Perm	CRF or Sponsor Defined	Record Qualifier
14	SRCDATA	AE	AEBODSYS	Body System or Organ Class	14	C	200	Exp	CRF or Derived	Record Qualifier
15	SRCDATA	AE	AELOC	Location of the Reaction	15	C	40	Perm	CRF or Derived	Record Qualifier
16	SRCDATA	AE	AESEV	Severity/Intensity	16	C	40	Perm	CRF	Record Qualifier

Display 3. TABLES and COLUMNS Metadata within the CST Framework

Running Validation Checks

To illustrate the connectivity the files described above, we'll start with the end of the validation process and work our way backward. The actual validation process is initiated with the execution of one macro: %sdtm_validate. Though this macro requires several pieces of information (e.g. the location of the SDTM data), it is defined without macro parameters. A thorough understanding of how CST works must begin with the knowledge of where and how it gets the information it needs.

In the last section, the SASREFERENCES data set was identified as a critical source of information for the names and locations of all the important files for the process. Once all the librefs and filerefs documented in SASREFERENCES have been allocated, the user has access to all the necessary files. Table 4 below contains a list of minimum parameters that must be present and documented in SASREFERENCES for an SDTM validation process to run successfully.

TYPE/SUBTYPE	Purpose
AUTOCALL	At minimum, one such observation should exist that documents the location of the SDTM macros provided by SAS. This can be in the default directory structure or in a copy of it. Other such observations can point to other chosen macro libraries. The ORDER variable determines the position of a library in the autocall path.
FMTSEARCH	Certain validation checks verify that the values of variables conform to a specified controlled term list. Such lists are to be contained in format catalogs. For the program to find such catalogs, these observations must be found in SASREFERENCES. The ORDER variable determines the position of a catalog in the format search path.
MESSAGES	At least two such observations should be found – one that points to a framework messages data set, and another that points to an SDTM-specific validation messages data set. Custom messages data sets can also be added.
CONTROL/REFERENCE	This points to the SASREFERENCES file

TYPE/SUBTYPE	Purpose
CONTROL/VALIDATION	The name and location of the control data set that contains the checks that %sdtm_validate will execute.
REFERENCECONTROL/ VALIDATION	The name and location of the master validation check data set.
REFERENCEMETADATA/ TABLE	The name and location of the data set that contains table information about the standard. This is used as a basis for comparison for certain SDTM validation checks.
REFERENCEMETADATA/ COLUMN	The name and location of the data set that contains variable information about the standard. This is used as a basis for comparison for certain SDTM validation checks.
RESULTS/ VALIDATIONMETRICS	The name and location of the data set that contains statistics about the execution of %sdtm_validate.
RESULTS/ VALIDATIONRESULTS	The name and location of the data set that contains results of the SDTM checks.
SOURCEDATA	The location of the SDTM study data.
SOURCEMETADATA/ COLUMN	The name and location of the data set that contains variable information about the current study.
SOURCEMETADATA/ TABLE	The name and location of the data set that contains table information about the current study.

Table 4. Required Parameters for Executing CST Validation

Creating Define.xml

In addition to the SDTM standards supplied by SAS, there is an additional standard whose name is CDISC-CRTDDS, which includes the files necessary for creating define.xml. This directory structure contains files and folders similar to the SDTM standard, but also includes some additional folders. Within validation/control is another master validation check data set that now serves as the master list of checks to validate the Define.xml data sets. There is a new folder called stylesheet that includes a default XSLT stylesheet to be used to help view the Define file.

Creating define.xml from the study metadata is a two step process. The first step is to run the macro supplied by SAS called %crtdds_sdtm311todefine10. This uses the SDTM tables and columns files to create a set of relational SAS data sets that contains the minimum set of components for define.xml. The user should note that this only includes the minimal set. The overall data model within CST includes 39 data sets that map to define.xml. However, there is nothing that allows the user to populate the other sets and thus it becomes a manual process. Running this macro takes the default SDTM metadata files and creates the set of CRT-DDS files.

After the set of CRT-DDS data sets are created, another CST macro called %crtdds_write creates the define.xml file along with a data set that summarizes the results of the execution of the process. Also during this process, a user can validate the relational data sets that are created to ensure they will be compliant with define.xml.

Just like the SDTM validation, this process requires a SASREFERENCES data set that must contain the parameters listed in Table 5 below.

TYPE/SUBTYPE	Purpose
AUTOCALL	One such observation should exist that documents the location of the CRT-DDS macros provided by SAS.
FMTSEARCH	Certain validation checks verify that the values of variables conform to a specified controlled term list.
MESSAGES	At least two such observations should be found – one that points to a framework messages data set, and another that points to an CRT-DDS specific validation messages data set.
CONTROL/ REFERENCE	This points to the SASREFERENCES file itself
SOURCEDATA	This points to the directory that stores the data sets used to create Define.xml.
SOURCEMETADATA	Two observations pointing to the column and table level metadata

TYPE/SUBTYPE	Purpose
EXTERNALXML	Defines location for the define.xml itself
RESULTS/ VALIDATION	This points to the data set that provides results of the validation of the data sets used to create Define.
RESULTS/ RESULTS	This points to the data set that provides results of the Define.xml generation process, including XML schema validation.
REFERENCEXML/ STYLESHEET	This points to the location of the XSLT stylesheet

Table 5. Required Parameters for Executing CST Define.xml Creation

After creating Define.xml, the user can view the Define file and the schema validation results by looking in the directory defined within SASREFERENCES. During our implementation of the CST define.xml process, we identified a number of issues with the content and implementation of the define.xml file. These have been documented and communicated to the SAS development team.

Limitations

We have implemented CST for customers and have identified a number of challenges and limitations during that process. As you can tell from the summary above, there is a significant learning curve regarding all the pieces and parts that are included in the Toolkit and more importantly how they fit together. While SAS does a fairly decent job documenting the functional bits, there is really no documentation on how to implement the tools within a company's process.

While the Toolkit has some robust tools for storing the metadata, running validation checks, and creating define.xml, it still requires a lot of manual work to get the metadata into the Toolkit. All the metadata is stored in SAS data sets and processes must be developed to feed those data sets.

The Toolkit does provide a robust set of validation checks including webSDM, JANUS, and SAS defined checks. However, there is no detailed documentation that describes the checks and how a company might implement them. If you just load metadata and let the programs rip, your validation process could run for days and you might not understand what you receive at the end.

Finally, during the implementation of define.xml, we identified a number of issues with how the define.xml is rendered and the various tags are populated with content. While the xml file passed a schema validation, the content is not reall in the right place in some examples.

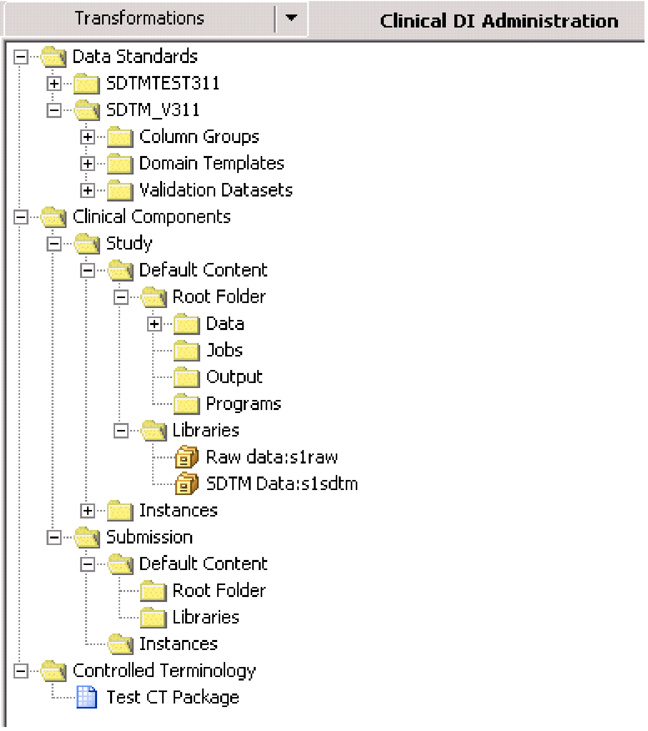
SAS CLINICAL DATA INTEGRATION

Based on their traditional data integration tools, SAS has developed Clinical Data Integration (CDI), a solution to support the management of standards as well as the transformation of clinical data to a standard. CDI consists of a set of plug-ins sitting on top of SAS Data Integration Studio which is a traditional ETL solution.

These added capabilities contain specific functionality relevant to the clinical transformation process including the management of CDISC standards, creating study specific components, facilitating customer specific standards, building SDTM custom domains, and reporting on the use of the standards across an organization. A key component of CDI is the previously described Toolkit which is the plumbing for CDI that executes the validation checks and creates define.xml. This section will provide a summary of CDI from a process perspective.

CDI Administration

One of the first things you notice after installing the CDI components is the addition of a number of customized tabs. The Clinical DI Administration tab within Display 4 is a CDI add on that helps manages the clinical components within the system. A clinical 'component' is a specific type of object in DI that that has customized metadata attached to it as well as the system understanding what to do with that object. This tab is where administrators would register new CDISC standards, register their own company specific standards, create a default study or submission folder structure, customize the metadata associated with a study or submission, and manage the controlled terminology.



- Data Standards
  - SDTMTEST311
  - SDTM_V311
    - Column Groups
    - Domain Templates
    - Validation Datasets
- Clinical Components
  - Study
    - Default Content
      - Root Folder
        - Data
        - Jobs
        - Output
        - Programs
      - Libraries
        - Raw data:s1raw
        - SDTM Data:s1sdtm
    - Instances
  - Submission
    - Default Content
      - Root Folder
      - Libraries
    - Instances
- Controlled Terminology
  - Test CT Package

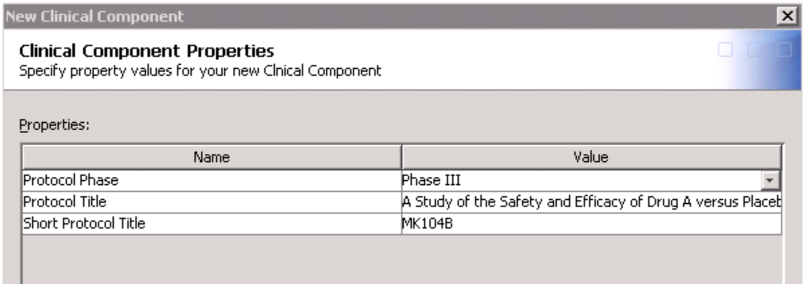
Display 4. Data Standards, Clinical Component, and Terminology Templates

The first step to use the 'clinical' capabilities of the solution is to register standards within the environment. This is the first glimpse into the key concept that CDI uses the Toolkit components described earlier as a significant part of its plumbing. In order for standards to be registered they must already exist within the Toolkit. In display 4, a user would select the Data Standards folder and choose the import option. This would connect to the underlying Toolkit framework and show the user the standards that have been registered. They can then import those standards and they would be available to use within CDI.

Within the clinical components users can create either a compound or study object. This is basically a template users can use when implementing a specific study. You can have one study template for your entire company or have multiple study templates, say by therapeutic area or compound. Each template contains a default directory structure as well as customized metadata. These specific clinical components are unique because they mean something to the other functionality within CDI. How you define this information is very dependent on your business process. This tab allows you configure your CDI environment and should be managed by small subset of Standards Administrators across your organization.

Initiating a Study

Once the standards have been imported and default submission and study templates have been created, a user can initiate a study. From the folders tab the user can select to create a new study which will guide them through a wizard. Within this wizard, they select the standard (e.g. Customer SDTM 3.1.2) associated with their study, complete customized study metadata, identify the associated libraries, and select the controlled terminology used with the study. Display 5 shows one screen within this wizard which lets the user define the metadata for a study.



Clinical Component Properties

Specify property values for your new Clnical Component

Properties:

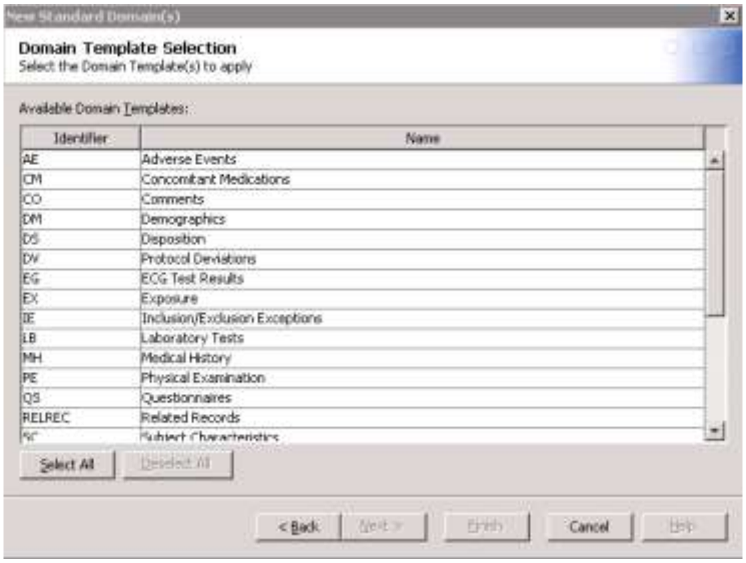
Name	Value
Protocol Phase	Phase III
Protocol Title	A Study of the Safety and Efficacy of Drug A versus Placeb
Short Protocol Title	MK104B

Display 5. Customized Metadata for the Study

When a user selects a standard to associate with the study they are selecting standards that have been registered in the Toolkit. The Toolkit comes with the default SDTM 3.1.1, however, no one implements either default SDTM model or all the associated domains. Instead they will customize this standard on numerous levels such as a company global standard, therapeutic standards, and compound/study standards. Each of these standards would need to be registered as a new standard within the Toolkit and imported into CDI. The standard can then be selected when creating a study. Creating this link tightly integrates the Toolkit framework functionality with the CDI capabilities.

Using Standards

Once a study has been created users can then begin working with the study transforming their raw data to a standard. They will first want to import the domains they want to use and modify the metadata that is specific to their study. Display 6 is part of a wizard for how a user would select the domains they need from the standard domains registered to that study. For example, they might not collect all the SDTM domains or specific variables within those domains.



Domain Template Selection

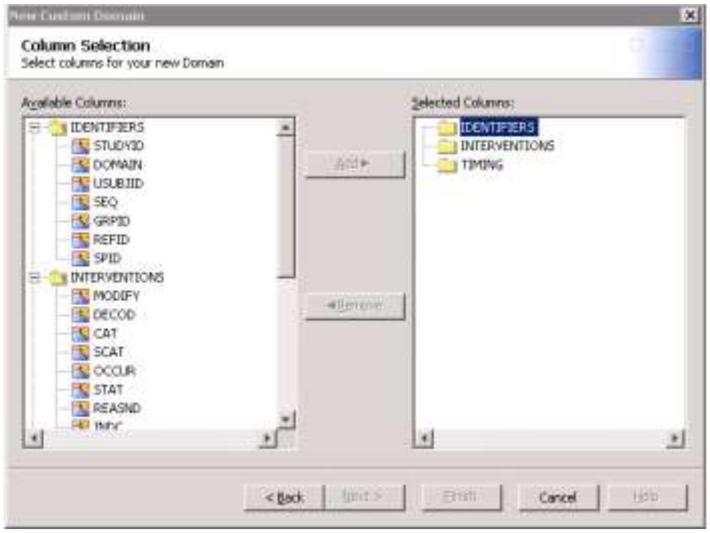
Select the Domain Template(s) to apply

Available Domain Templates:

Identifier	Name
AE	Adverse Events
CM	Concomitant Medications
CO	Comments
DM	Demographics
DS	Disposition
DV	Protocol Deviations
EG	ECG Test Results
EX	Exposure
IE	Inclusion/Exclusion Exceptions
LB	Laboratory Tests
MH	Medical History
PE	Physical Examination
QS	Questionnaires
RELREC	Related Records
SC	Subject Characteristics

Display 6. List of Domain Templates when Creating a Study

After selecting the specific domains of interest they can customize domains removing variables, modifying metadata, or defining additional rules. In addition to using the standard domains, the user can also create custom domains based on the SDTM standard. The user will be guided through a wizard that will help them make selections on the type of SDTM structure and variables required. Display 7 shows one screen within the wizard that guides the user through defining the standard variable fragments from the SDTM structure types.



Column Selection

Select columns for your new Domain

Available Columns:

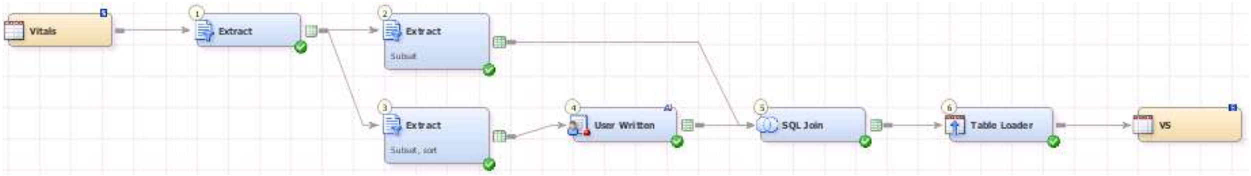
- IDENTIFIERS
  - STUDYID
  - DOMAIN
  - USUBJID
  - SEQ
  - GRPID
  - REFID
  - SPID
- INTERVENTIONS
  - MODIFY
  - DECOD
  - CAT
  - SCAT
  - OCCUR
  - STAT
  - REASND
  - INDC

Selected Columns:

- IDENTIFIERS
- INTERVENTIONS
- TIMING

Display 7. List of Standard SDTM Fragments for Creating New Domain

After the user has completed the customization of their domains for a study, they can begin building ETL processes known as jobs. Display 8 contains a example of a job that creates the SDTM VS domain. Building jobs is standard functionality of an ETL tool and won't be covered within this paper. While we won't dwell on default ETL functionality, this is probably the most challenging part of the process for customers who are more familiar with base SAS programming. A discourse on using ETL solutions is found below.



Display 8. CDI Job that Creates an SDTM VS Domain

Once the user has developed jobs there are additional tools within CDI that can support the compliance of metadata within and across studies as well as the creation of the define.xml. Customized clinical transformations supplied with CDI will allow the user to run the standard SDTM validation checks, explore reports across studies that communicate where specific domains and variables are used, and generate define.xml. However, once again, CDI uses the Toolkit as the plumbing so any issues identified in previous sections will also be carried over to CDI.

Using an ETL Solution

In general, ETL products use a rigorous process that separates inputs from the transformation code and the target tables. In some industries this very rigid process works well because the source structure, and more importantly, the target data are very standard and robust. However, within clinical data, both the underlying medical science and the analytical science are always changing, so defining rigid targets and repeatable processes can be challenging. The famous saying for a clinical programmer is 'This study is unique'. In addition, the derivations that sometimes occur in the transformation process can be quite complex.

Over the last three decades, the process for building clinical data sets has to been to write and run SAS code. An ETL process as defined above has been used many times by scores of clinical programmers – even if they do not want to put a formal name to it. For instance, data is extracted from Oracle Clinical into SAS data sets. Programmers then write SAS code to clean, analyze, and transform into alternative forms (e.g. SDTM domains). In fact, a normal SAS programmer will say that they have been practicing ETL for many years!

However, a true ETL process will bring more to the table than just the code that is used to move data from point A to point B. The majority of ETL tools available today allow you to use metadata to perform variety of automated tasks such as documenting standard processes to allow for impact analysis. These functions bring value added to the table for the development process.

In general, clinical programmers will try to write code from the top down. This contradicts the traditional methodology of ETL and software development programming in which you design and modularize your code as much as possible. In building large scale systems, architects usually spend a majority of their time designing the components before they ever write a line in a code and the same methodology should be used within clinical programming. In addition, clinical programmers will sometimes try to write as much code within a single task as possible. Breaking down the tasks within your program into smaller encapsulated pieces makes the code more reusable and easier to review. While writing smaller more straightforward pieces might make the program lengthier and seem tedious it provides a more scalable solution and hopefully a significant amount of reusable code.

Another challenge revolves around the individual programmer's need to get the work done. They just want to write SAS code and this new 'tool' only makes their work more tedious and slows down their production. What they don't realize is that by supporting a 'write some SAS code' approach to transforming data, they create a process that is fractured and not repeatable. More importantly, it does not support the management and reuse of metadata which is critical to developing and maintaining standards.

The overarching question is whether existing ETL tools provide the flexibility needed within clinical programming while maintaining the structure and rigors of an ETL process.

CONCLUSION

CDISC has had an interesting evolutionary process over the last decade and has experienced its own trials and tribulations in the development and marketing of standards. However, in recent years, with the help of FDA and industry, everyone involved in the development, adoption, and delivering of clinical data standards seems to be on the same path and headed in the right direction. As a whole, the industry has gone from 'if' they should adopt CDISC standards to 'when' and 'how'. The adoption of CDISC standards will continue to mature and over the next decade they will become an integral part of the business process.

SAS has tried to keep up with both the CDISC standards as well as the needs of users who work with clinical data standards. While a number of challenges and limitations have been identified within this paper, the overall direction and dedication that SAS seems to be making is very promising. With the SAS Clinical Standards Toolkit and SAS Data Integration solutions, SAS appears to be headed in the right direction with supporting the needs within the industry. They still face the challenge of the traditional SAS programmer who just wants to write code, but the gap is closing as efficiencies become more apparent with the use of these tools.

ACKNOWLEDGMENTS

I would like to acknowledge Mike Molter, a Senior Life Sciences Consultant at d-Wise who has done most of the heavy lifting for the team and provided significant input during the development of the paper.

CONTACT INFORMATION

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

Name: Chris Decker
Enterprise: d-Wise Technologies
Address: 4020 Westchase Blvd Suite 527
City, State ZIP: Raleigh, NC 27607
Work Phone: 919-600-6234
E-mail: cdecker@d-wise.com
Web: www.d-wise.com