

## SDTM mapping becomes more EASY with SAS Clinical Data Integration!!

Saumilkumar Tripathi, Cytel, Pune, India

### ABSTRACT

CDISC SDTM (Study Data Tabulation Model) is recommended as the standard data structure to be used for submitting tabulation data to the US FDA for clinical trials. Approach for implementing such data standards should be quick and effective with less resource utilization. SAS® Clinical Data Integration is the solution from SAS® which helps user to organize, standardize and manage clinical research data and metadata in compliance with CDISC SDTM or company specific internal standards.

This paper will present the Graphical User Interface (GUI) approach compared to the conventional programming for SDTM mapping, performing CDISC Compliance checks and preparation of CRT-DDS define.xml which highlights the usage and unique features of SAS CDI. It also includes automation of such data integration tasks to gain speed and efficiency, different ways to check domain completion status, comparing metadata properties of domains across the studies and many more.

### INTRODUCTION

Many pharmaceutical companies and CROs consider each clinical data integration task as a unique effort which requires extensive manual coding and processes that are neither standard nor repeatable. SAS Clinical Data Integration solution brings the power of Graphical User Interface to integrate, extract, transform and load the clinical data and metadata across entire organization with the help of SAS® Clinical Standard toolkit. Using such ETL tool, it is quite easy to deploy clinical data standard models such as the CDISC SDTM or company specific standards within the organization which can be leveraged by all users without the need of developing additional metadata libraries or programs.

Development and Maintenance time of programs can be greatly reduced with help of any code generator tools compared to conventional programming platform. So to highlight such advantages, in this paper we will discuss SDTM implementation approach with SAS CDI considering it's unique features and advance options to create, validate and finally automate clinical datasets preparation and compliance check followed by the preparation of CRT-DDS Define.xml document. This paper will help all clinical programmers to understand the new way of implementing data standards with GUI technology to save time and manual programming efforts.

### DIFFERENT APPROACHES FOR SDTM MAPPING

There are different approaches for getting your data into shape from raw data. This paper will discuss the different methods which can be used for getting raw data into CDISC SDTM datasets. This process usually referred as Mapping. We will limit our focus to discuss conventional programming method and GUI method using SAS CDI.

#### 1) CONVENTIONAL PROGRAMMING APPROACH :

It is the most common method used by the industry over the years for preparing submission ready datasets. As a clinical programmer there are many options to go for. The objective is to accelerate the process in all way for accessing, managing, analyzing and finally reporting clinical data to reviewing agencies and getting the new drug faster in to the market. This section will discuss the example of creating demographic domain with BASE SAS® approach which involves SAS programming.

Following is the basic sample SAS code for creating DM SDTM dataset. This program involves calls of SAS Data step, Proc steps and Functions usage to create the desired final dataset.

```
/*Extracting the data from required Raw-Datasets*/  
  
proc sort data=rawdata.random out=random ;  
  by studyid ptno ;  
run;  
  
proc sort data=ocdata.invinfo out=invinfo ;  
  by studyid ptno ;  
run;
```



In above example three different raw datasets were used to create one combined dataset to get the information on randomization, Investigator and other demographic details. Random is the raw dataset available in SAS native from while other two raw datasets were extracted from Oracle database assuming that respective libraries were already defined.

This program involves Proc step to sort the data and output the sorted version of dataset in work library, Data step to combine all the raw dataset into one dataset and Final data step to output the SDTM DM dataset to permanent library with usage of various SAS Functions and Formats to transform the data. Also compliance can be checked through set of SAS commands and Define.xml file can also be created with the use of various macro utilities in BASE SAS Programming.

**2) GRAPHICAL USER INTERFACE APPROACH WITH SAS CLINICAL DATA INTEGRATION :**

As of now we have explored creation of SDTM datasets with BASE SAS Programming approach, it is now quite exciting to have a look on SAS Clinical Data Integration for SDTM data creation, compliance check and many more things. SAS CDI is the ETL tool built on top of the SAS® Data Integration product and it uses the component called Clinical Standard Toolkit which will be used by various processes in SDTM development.

Let's explore the SAS Clinical Standard Toolkit in detail what have been mentioned in above paragraph. The SAS Clinical Standards Toolkit:

- Created by SAS developers using SAS Foundation code and the Java language
- This toolkit contains templates for standard CDISC SDTM domains, compliance checks, and controlled terminology tables with required variable level metadata as well as integrity constraints.
- It also contains SAS Foundation code utilities to support the import of standards and compliance checks to SAS Clinical Data Integration

SDTM dataset creation process with this method is divided into three parts as described in **Figure 1**.

- Extraction:** Extraction of input files from various platforms into SAS CDI Job will be carried out in this process. Best part here is input data can be easily imported from any kind of database like Oracle, Excel, Rave or SAS etc. supported by SAS platform with help of point and click interface which can also subset the input data as required.
- Transformation:** This part of the process includes necessary data conversion and various data management techniques to get the data into required shape. Advantage of SAS CDI is that all necessary transformations like Sorting, Look-up, SQL-Join etc. already available in tool, User just have to add them into the Job and need to define the parameters in transformation properties for the same through point and click wizard.
- Loading:** Finally, in this part target domain needs to be added which is the standard SDTM dataset template with all the metadata information pre-loaded with the table. Such domains will be available to add in to the job from Clinical Standard Toolkit which has been discussed in earlier part of the paper. Variables from raw datasets need to be mapped to the respective SDTM variable from earlier added target domain. Some variables will need use of SAS expression to derive the variable correctly.

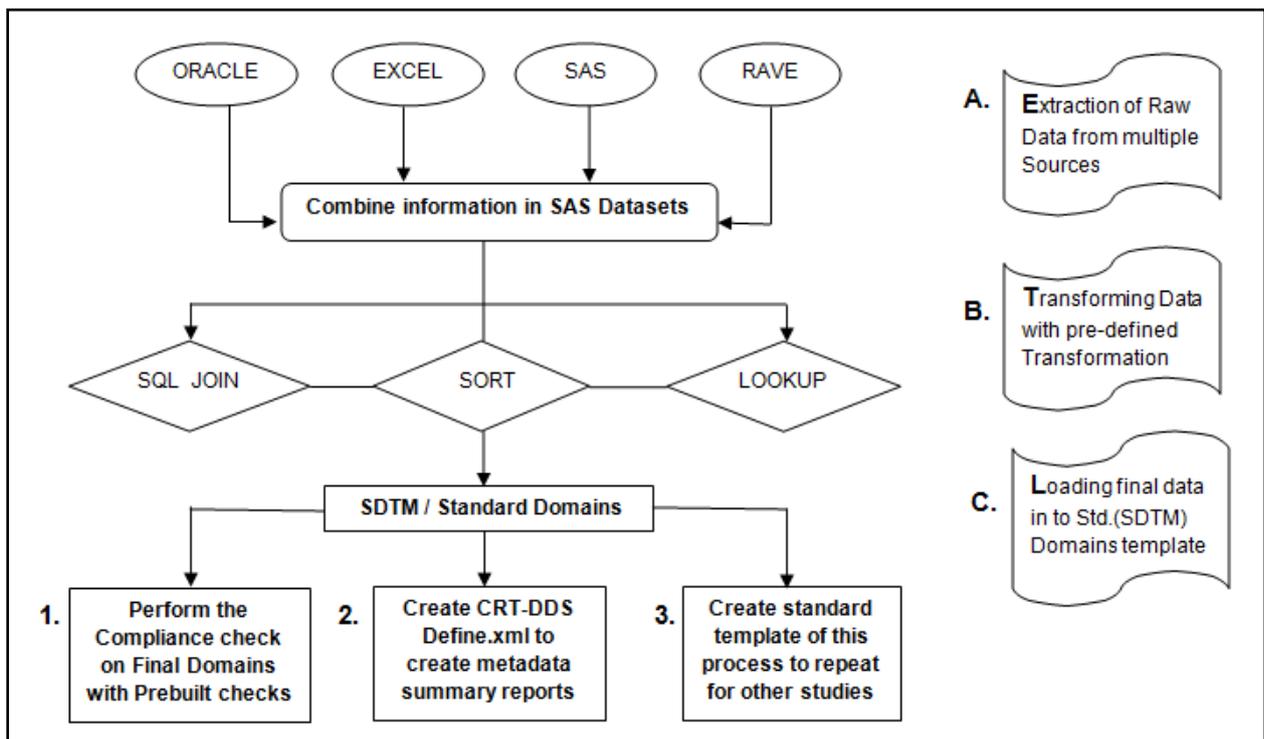


Figure 1: Flow diagram of ETL process in SAS® Clinical Data Integration

So, to get more clear idea, let's look at to the practical example of entire process. Following are the two different diagrams explaining building the Jobs in SAS CDI for creating SDTM AE & DM datasets in **Figure 2** and **Figure 3** respectively.

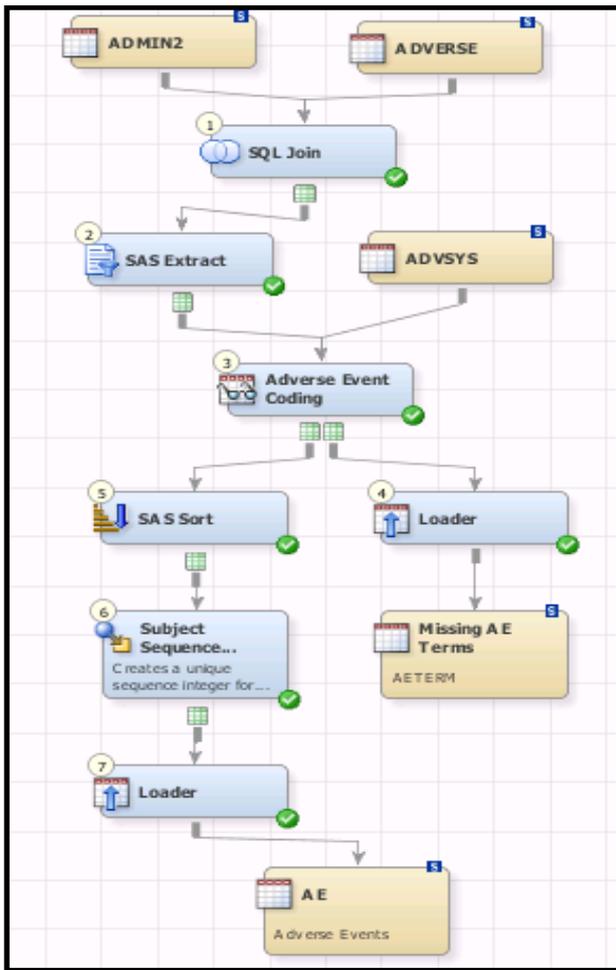


Figure 2: Job diagram to Load the AE SDTM Dataset

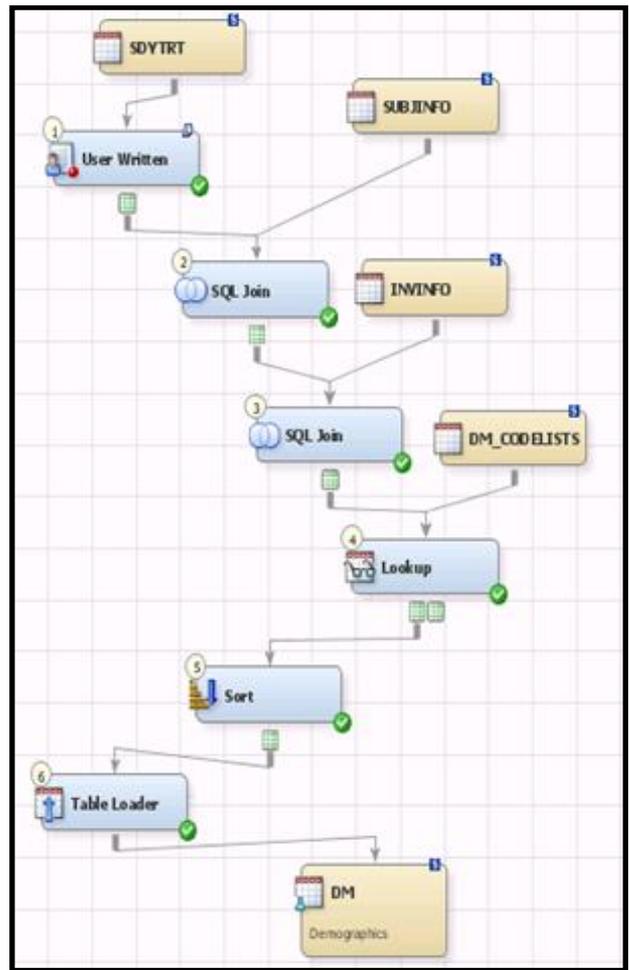


Figure 3: Job diagram to Load the DM SDTM Dataset

This Job loads the respective SDTM dataset with the required data. The process is carried out in three steps Extraction, Transformation and Loading which are discussed in brief earlier. In both Jobs input data is extracted from respective raw data sources and it is loaded into pre-defined target domains through variable mapping with required data transformation. Clinical standard toolkit will act as central metadata repository which is having all Standard SDTM dataset templates with Metadata properties only and no actual data. So these templates will be used as Target Domains, while respective raw datasets from various platforms will act as Source Tables and SAS CDI will support different Transformations like Sort, Lookup, SQL-Join etc. highlighted with sky blue colour in above job flow for required data transformation activities.

Data can be extracted through import wizard from any database supported by SAS Engine. In above example all pale yellow coloured boxes are referred to input SAS datasets except last one (AE and DM) which are SDTM target domain with pre-defined metadata properties like column name, label, length, format etc. SAS CDI has a capability of connecting to Oracle, Rave, Teradata, Db2, Microsoft Excel and various databases based on available licenses of SAS/Connect®. Figure 2 contains ADMIN2, ADVERSE and ADVSYS as source tables (Input rawdata) for Loading AE domain job while Figure 3 contains SDYTRT, SUBJINFO, INVINFO as source table and DM\_CODELISTS as lookup table for Loading DM domain job.

SQL Join, lookup, Sort are the commonly used SAS CDI transformations which are also used in above examples for performing various data manipulation techniques. After adding this transformation in to CDI Job from the given tab, programmer just needs to define the various parameters under the properties window. Data which is extracted from source tables are going to be stored in temporary dataset created by individual transformation which is used to transfer data to the next transformation or to the table loader for loading the final data in to SDTM target domain imported from Clinical standard toolkit. Sort transformation is used for sorting the data and lookup is used for looking up for the code list in Lookup Dataset (AE Terms, Questionnaires code list etc.). Below **Figure 4** is containing an example of inner diagram of SQL join transformation containing navigate panel at top right corner which can be used to construct the SQL query in a way user wants. Navigate panel can be used for selecting columns to be included in select clause, table to be used in from clause and type of join (Left / Right / Inner Join) to be performed in SQL query. Also Clauses tab can be used for adding or removing where, group by, having or order by clauses. Complex join can also be performed with help of subquery and inline views.

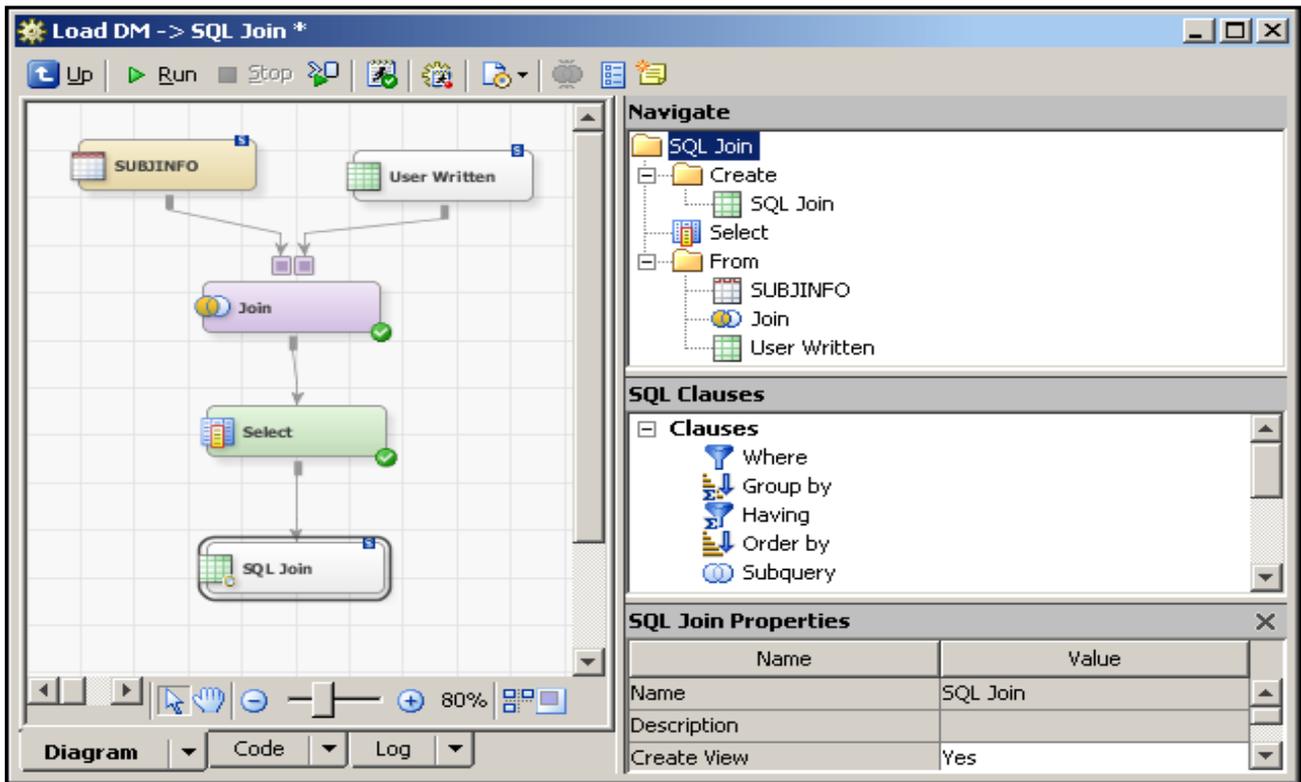


Figure 4: Inner diagram of SQL Join transformation

In GUI environment programmer can always use inbuilt transformation for different task and also have an advantage of writing their own SAS Foundation code using User Written transformation for performing complex programming logic including macro creation and calls. As metadata for the tables are pre-defined in the available templates of target domain, programmer just need to map the respective column with help of drawing an arrow from source to target table as highlighted in below **Figure 5**. In addition to mapping, certain columns may require SAS Expression to convert the variable type or to make use of various SAS functions. SAS Expressions are usually contains the constant text for character variables or use of SAS functions like upcase, lowcase, scan, substr, input, put etc. for different data transformation activities. Most of the transformation available in SAS CDI has mapping tab under the properties window which can be used to add or modify the variable mapping. SAS CDI has a feature of automatic propagation of all columns from source tables to temporary target tables which means copying and mapping data automatically from source to target. This can be controlled through turning this option on or off from job editor window. In Manual propagation user can select data from which source column need to be added to which target column or how data should be mapped from source to target with or without use of appropriate SAS Expression. Common use of expression during mapping is to convert variable type or to remove leading and trailing spaces from the data or to perform the conditional logic based on given specification for SDTM variable derivation.

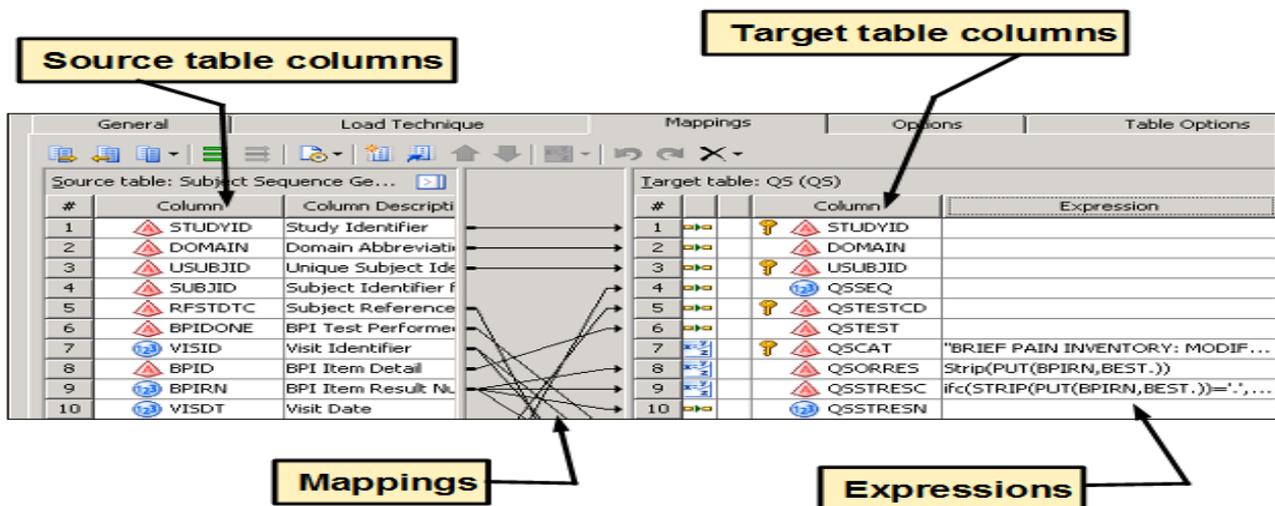
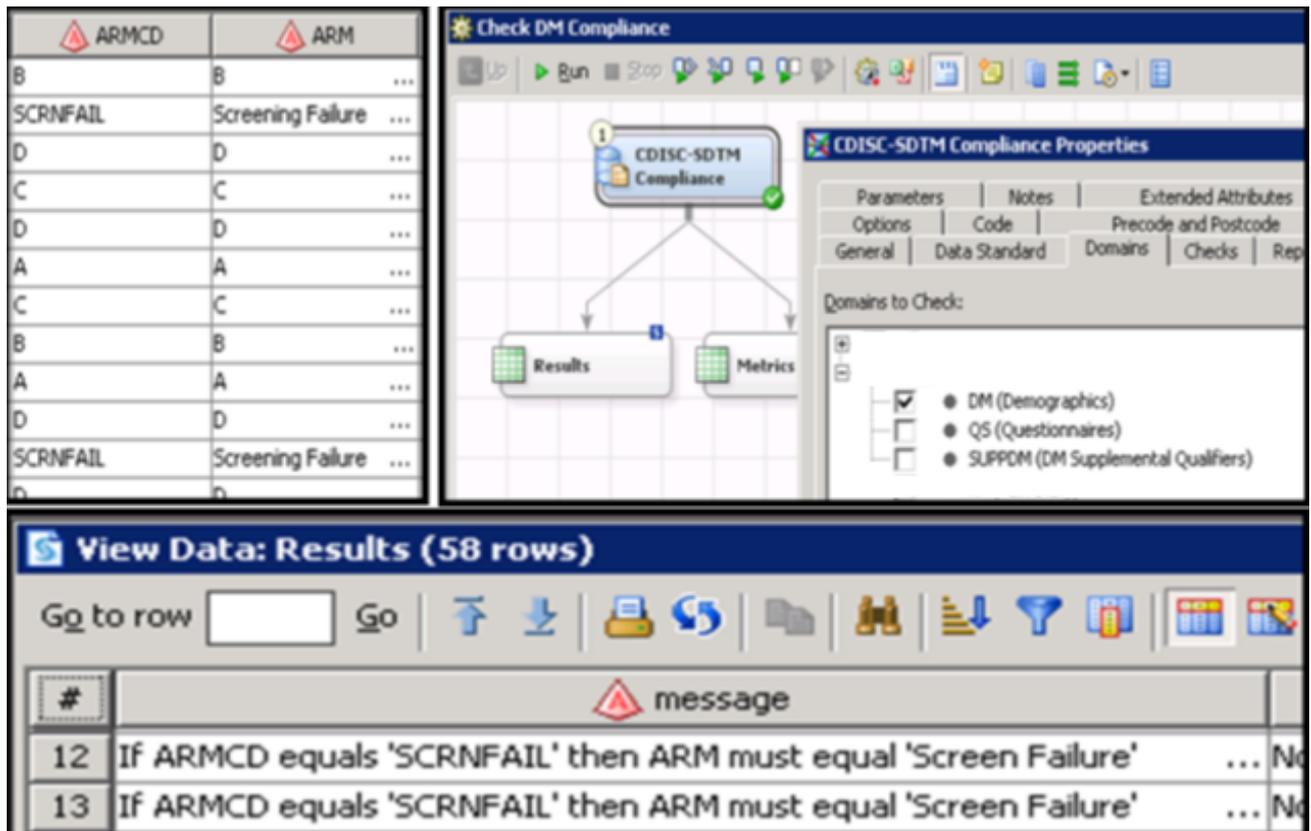


Figure 5: Column mapping diagram from source to target table with use of expression for few column derivation

## PhUSE 2014

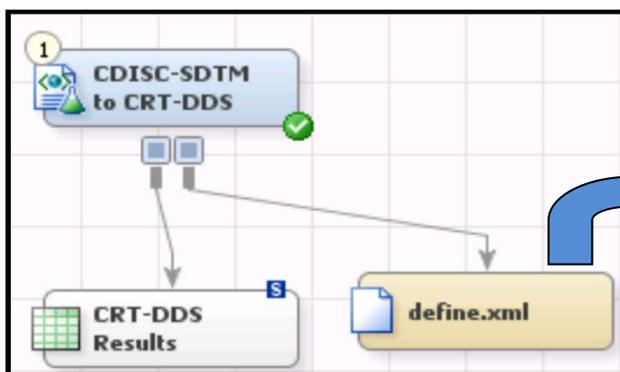
Table Loader is an important transformation to load the data from the temporary tables generated by the various transformations in to Target SDTM domains with final variable mappings. In above mentioned Loading AE and DM jobs, data is getting loaded into AE and DM SDTM dataset from the various source tables with the use of necessary transformation.

Once the SDTM domain is ready, it can be validated for CDISC SDTM compliance through previously imported compliance checks from Clinical standard toolkit in SAS CDI. There is a pre-built transformation in SAS CDI called as CDISC-SDTM Compliance where user just need to edit the properties of the transformation to select required checks for specific SDTM domain. **Figure 6** illustrates the usage of such compliance check transformation for identifying the correct values of ARM variable. Here some records contain the value of ARM as “Screening Failure” instead of required value of “Screen Failure”. Result metrics of executed checks can be created as RTF, PDF or HTML file for further review and reference.



**Figure 6: SDTM Compliance check job and resulting dataset highlighting one of the reasons for failed check**

After getting acceptable report from compliance checks run, programmer can create Define.xml file with similar way of process using “CDISC-SDTM to CRT-DDS” transformation available in SAS CDI. This file can be further customized based on client’s or submission requirement. Below **Figure 7** is the job flow highlighting the use of CDISC-SDTM to CRT-DDS transformation to create define.xml and **Figure 8** is the snapshot of created define.xml file.



**Figure 7: Job diagram for creating CRT-DDS Define.xml**

Dataset	Description	Structure	Purpose	Keys
DM	<a href="#">Demographics</a>	Special Purpose Domains - One record per subject	Tabulation	STUDYID USUBJID
QS	<a href="#">Questionnaires</a>	Findings - One record per questionnaire per question per time point per visit per subject	Tabulation	STUDYID USUBJID QSTESTCD QSCAT VISITNUM
SUPPDM	<a href="#">DM Supplemental Qualifiers</a>	Special Purpose Datasets - One record per IDVAR, IDVARVAL, and QNAM value per subject	Tabulation	STUDYID RDOMAIN USUBJID QNAM

**Figure 8: Created Define.xml File**

## PhUSE 2014

Moreover, SAS CDI supports creation of custom study specific domains and there is feature called Promoting Custom domain as standard domain which will help programmer to add that custom domain template in Standard template list so that it can be later used directly in similar other study with pre-defined metadata. Similarly custom compliance checks can also be created for the validation of custom domains. As a Clinical administrator one can monitor the domain development or validation status through Monitor Domain Status wizard and can also import the additional Controlled Terminology datasets based on requirement. Analyzing Domain Usage Across the Studies wizard will help in identifying the usage of particular domain across the study or submission. So, here from creation of SDTM dataset till generating Define.xml file all process will be carried out with the power of GUI interface and minimal coding efforts. Thus, automation and repeatability could greatly be increased using such tools.

### WHICH IS BETTER?? PROGRAMMING v/s GUI APPROACH

It has always been difficult to migrate from one approach to another for any process, as this will require new tools, experienced team to work on that tool or inter personnel training etc. Although BASE SAS Programming has much flexibility to write the code and to get the desired reports or datasets, moving to GUI approach will greatly reduce the production and maintenance time of CDISC SDTM dataset development and validation.

#### 1) BENEFITS WITH BASE SAS PROGRAMMING APPROACH :

- No restrictions in terms of coding, as sometimes you can write complex logic through programming easily rather than getting that done in GUI.
- Full power of BASE SAS can be utilized through use of SAS Procs, Macro, SQL, Functions and Call routines etc. in programming to extract, transform and load the data.
- SAS Macros can be used to develop SDTM datasets to maintain the consistency across the study or Project.
- SAS Programming approach is cost effective compared to GUI tool which requires need of purchasing multiple licenses for using the product

#### 2) BENEFITS WITH GUI APPROACH(SAS CDI) :

- No need of manual coding which minimizes development time of any Clinical datasets preparation and validation.
- Drag and drop data transformation wizards to manage the clinical data.
- Automating data integration jobs to make use of resources in a better way(SDTM mapping & Compliance check).
- Graphical User Interface(GUI) gives you more flexible environment to perform various actions.
- Centralized Metadata management to check Domain Status and Metadata properties across Studies.
- Pre-defined SDTM dataset template, Compliance checks and Controlled Terminology tables imported from SAS Clinical Standard Toolkit.

### CONCLUSION

In this paper we compared BASE SAS Programming approach and GUI approach with use of SAS CDI in the task of converting clinical trials data into the standard CDISC SDTM datasets. One is the method which is in use since so long in the industry and another one is quite new, exciting, less time consuming, more result oriented and customized version of old process. In BASE SAS we have little tool support and more programming platform while in SAS CDI we have context specific drag and drop wizards to perform various actions through point and click. However, both methods have its own advantages and disadvantages. Overall purpose of this paper is to illustrate advance technology in area of SDTM development and validation which is the most common and important task now days in Clinical industries as this is the recommended data standards to be used for submitting clinical trials data to US FDA.

### REFERENCES

- 1) Confessions of a Clinical Programmer: Dragging and Dropping Means Never Having to Say You're Sorry When Creating SDTM Domains : <http://support.sas.com/resources/papers/proceedings11/205-2011.pdf>
- 2) [http://www.sas.com/en\\_us/industry/life-sciences/clinical-data-integration.html](http://www.sas.com/en_us/industry/life-sciences/clinical-data-integration.html)
- 3) <http://www.sas.com/resources/factsheet/sas-clinical-data-integration-fact-sheet.pdf>

### ACKNOWLEDGMENTS

I would like to thank all my colleagues at Cytel who shared their experiences, read my drafts and gave me the valuable and constructive feedback.

## PhUSE 2014

### CONTACT INFORMATION

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

Saumilkumar Tripathi  
Cytel Statistical Software & Services Pvt. Ltd.  
6<sup>th</sup> Floor, Lohia-Jain IT Park – A Wing,  
Survey #150, Paud Rd, Bhusari Colony,  
Kothrud, Pune, Maharashtra 411038, India  
Work Phone: + 91-20 6709 0226  
E-mail: [saumilkumar.tripathi@cytel.com](mailto:saumilkumar.tripathi@cytel.com)

SAS and all other SAS Institute Inc. product or service names are registered trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks of their respective companies.