

Converting CDISC Controlled Terminology to SAS Formats

Sandeep Juneja, ASG Inc., Cary, NC
 Vivek Mohan, ASG Inc., Cary, NC

ABSTRACT:

CDISC Controlled Terminology (CT) is used to define and support the terminology needs of the CDISC domains. Applying CT to individual variables with different values across data sets in multiple studies can be a very cumbersome and pain-staking process. This paper describes an easy way of extracting unique values of the variable and unique values of the CT term and provides a GUI interface to carry out one-to-one mapping for CT terms and data values. This mapping allows generating customized SAS Formats which can be applied directly to raw data set variables when converting to CDISC compliant data sets. This utility not only reduces the time required to apply CT to applicable CDISC variables, but also eliminates any errors caused by manually typing in each unique value for a variable. The software/modules used in this utility are SAS, Microsoft Excel, VBA and SAS IOM.

INTRODUCTION:

CDISC CONTROLLED TERMINOLOGY (CT)

CT has been developed to standardize the unique values for certain variables in standard CDISC domains.

SAS INTEGRATED OBJECT MODULE (IOM)

SAS defines the Integrated Object Model as a set of “distributed object interfaces to SAS... IOM enables you to use industry-standard languages, programming tools, and communication protocols to develop client programs that access these services on IOM servers.”

By setting up references to the SAS IOM components in your VBA program, you can achieve most of the results that you can with SAS in batch.

THE SAS STORED PROCESS SERVER

SAS Stored Processes are slightly different than traditional SAS programs in that they must be executed on a SAS Integration Technologies server. They cannot be executed in a normal batch or interactive SAS session. The main difference between running a SAS job in a Workspace and running it as a Stored Process is that, with the former approach you can generate and submit code from the client or run code already existing on the server, while with the the Stored Process Server the code has to be accessible from the server.

	A	B	E	K
1	VARNAM	VARLBL	FMT	DATAMAP
11	AEDECOD	Dictionary-Derived Term	*	AESAES.AESPT_(\$150.)
15	AEBODSYS	Body System or Organ Class	(SOC)	AESAES.AESSOC(\$150.)
17	AESEV	Severity/Intensity	(AESEV)	AESAES.AESITSL(\$200.)
18	AESER	Serious Event	(NY)	AESAES.SAEPREF(\$NO_YES.)
19	AEACN	Action Taken with Study Treatment	(ACN)	AESAES.SDG_NAF(\$NO_YESA), AESAES.SDGNONF(\$NO_YESA), AESAES.SDGCHEG(\$NO_YESA), AESAES.SDGTDF(\$NO_YESA), AESAES.SDGPDI(\$NO_YESA)

Requirements:

The Mapping document is the most important document required for this utility (shown on left). Apart from the CDISC variable name and CDISC variable format, the mapping document should contain the following information at the CDISC variable level:

1. CT names must be specified, if applicable.
2. The variable name(s) and raw data set name(s) that maps to the CDISC variable must be specified in the following format <raw data set>.<variable name>(Format of Variable, if present) . It can either be one-one or one-many depending upon the data sets or variables used to derive the CDISC variable.

The mapping document in conjunction with the raw data sets provides the information required to produce the CT Mapping Excel Sheet.

THE PROCESS:

1. CREATING INDIVIDUAL CT DATA SETS

Download the available CT Excel workbook from CDISC website (see link in 'More Information' section at the end of this paper) and convert each CT as a unique SAS data set. Please make sure that each SAS data set has the same name as that of the controlled terminology it represents. For e.g.: acn.sas7bdat will correspond to ACN controlled terminology.

2. GENERATING THE CT MAPPING EXCEL SPREADSHEET

Using the mapping document and the sas data sets generated for each CT, a sas program can be written to generate the excel spreadsheet that contains the following information per CDISC domain (see below)

1. Summary Information: This lists the unique CTs and list of variables to which the CT is applied.

	A	B	C	D	E	F
1		Controlled Terminology		Variable List		
2		ACN		AEACN		
3		AESEV		AESEV		
4		NY		AESER, AESHOSP, AECONTRT		
5		OUT		AABOUT		
6		SOC		AEBODSYS		
7						
8						
9						
10						
11						
12	CDISC Format Name	CDISC Variable Name	Raw Variable Name (Label)	Raw Variable Values	Controlled Terminology Value(s)	Comment
50	OUT	ABOUT	AESOUTL(AE: Outcome - stat-DWL)		OUT	
51	OUT	ABOUT	AESOUTL(AE: Outcome - stat-DWL)	Ongoing	OUT	
52	OUT	ABOUT	AESOUTL(AE: Outcome - stat-DWL)	Resolved	OUT	
53	OUT	ABOUT	AESOUTL(AE: Outcome - stat-DWL)	Resolved with sequelae	OUT	

2. Variable Mapping Information: It contains the following information

- (i) CDISC Format Name
- (ii) CDISC Variable Name
- (iii) Raw variable (Raw variable label)
- (iv) Raw Variable distinct values
- (v) Name of the CT
- (vi) Section for any user comment.

Hint: Use ExcelXP tagset along with SAS to generate the above excel workbook with one sheet per CDISC domain.

3. SAS IOM/ VBA

Establish the connection between SAS IOM & VBA using appropriate References & code.

References:

In order to use IOM from a VB project, it is necessary to add in references to the SAS link libraries. To add these, go to *Project>References* on the main VB menu. Scroll down the list and click to add the following references:

```
Dim obWS As SAS.Workspace
Dim obWSMgr As New
SASWorkspaceManager.WorkspaceManager
Dim obDS As SAS.DataService

' This Function is used to Start SAS and initialize
local Workspace Manager

Public Function Start_SASjob()

    Dim XmlInfo As String

    'Create a local SAS Workspace
    Set obWS =
obWSMgr.Workspaces.CreateWorkspaceByServer _
    ("", VisibilityProcess, Nothing, "",
    "", XmlInfo)

End Function
```

- SAS: Integrated Object Model (SAS System 9.1)
- SASWorkspaceManager 1.1 Type Library

This will allow VB to resolve references to the SAS objects used to connect to the server.

A brief overview of the steps used in our utility to develop the GUI interface is provided below

- i). Establish Connection with SAS IOM - Below is the standard code used to establish the connection bridge between SAS & VBA.

```

Set obLibRef = obDS.AssignLibref("CT", "", ctdata,
"")

Call Submit_SAS("proc sort data=ct." & rkdst & "
out=tds(rename= (CDISC_Preferred_Term= ctname) keep=
CDISC_Definition CDISC_Preferred_Term); by
CDISC_Preferred_Term; run;")

' Create the connection to SAS via ADO
obConnection.Open "provider=sas.iomprovider.1;
SAS Workspace ID=" + obWS.UniqueIdentifier

' Open work.tds dataset
obRecordSet.Open "work.tds", obConnection,
adOpenStatic, adLockReadOnly, adCmdTableDirect

' Create object for Recordset.fields
Set fld = obRecordSet.Fields

' Initialize the Variable and Record Counter
Dim VarCnt As Integer
Dim RecCnt As Integer

' Get the Variable and Record Counter
VarCnt = obRecordSet.Fields.Count
RecCnt = obRecordSet.RecordCount
If RecCnt = 0 Then GoTo Fun_Exit

' Re-Initialize the Vars array to the size of
dataset
ReDim var1(RecCnt, VarCnt) As Variant
ReDim var2(RecCnt, VarCnt) As Variant

< ----- Insert Processing code ----->

' De-Assign LibRef and release the Data Service
obDS.DeassignLibref (obLibRef.Name)

' Close the ADO connection
obConnection.Close
Set obConnection = Nothing

Call End_SASJob

```

Also, similar to the code above, snippets of code to close SAS, Submit the SAS job etc. are available on the internet.

ii). Establish the Library connection for the CT data and raw data set.

iii). Establish the connection to SAS via ADO to read in the SAS data set.

iv). Read in the unique value for CT data and for each raw data set variable and display the results in the GUI screen.

v). Select the appropriate Raw variable value and the CDISC value and implement one-one mapping.

4. GENERATING THE FORMAT.SAS PROGRAM

Once the mapping is completed, the stored SAS process can be called to create the format.sas program.

```
Call Start_SASjob
Call Submit_SAS("%include '" & lpath & "\Create_Fmt.sas'")
Call Submit_SAS("%Create_Fmt(xlfile=" & nwbk & ",sheetnm=" &
sname & ");")
Call End_SASJob
```

```
%macro Create_Fmt(xlfile=, sheetnm=);
  %let filen=%scan(&xlfile,-1,"\");
  %let fpath=%substr(&xlfile,1,%index(&xlfile,&filen)-1);

  proc import datafile="&xlfile" out=final dbms=EXCEL replace;
  getnames=No;
  sheet="Sheet1";
  run;

  data final;
  length dsnm $15.;
  set final;
  dsnm="&sheetnm";
  run;

  proc sort data=final out=final1 nodupkey;
  by dsnm f1 f4 f5;
  where f1~=' ';
  run;

  options noquotelenmax;
  filename sasfile "&fpath\&sheetnm._format.sas";

  ←----- Processing code to generate formats ----→
%mend Create_Fmt;
```

Snapshot of Create_Fmt.sas program.

```
*****START: FORMAT CREATION *****;

proc format;
  value $ACN
    '1'='DOSE INCREASED'
    '2'='DOSE REDUCED'
    '3'='DRUG INTERRUPTED'
    '4'='DRUG WITHDRAWN'
    '5'='DRUG NOT CHANGED'

  ;
  value $AESEV
    ' '=' '
    'Mild'='MILD '
    'Moderate'='MODERATE '
    'Severe'='SEVERE '

  ;
  value $NY
    'N'='No '
    'Y'='Yes '

  ;
quit;
***** END: FORMAT CREATION *****;
```

Snapshot of Format.sas program

CONCLUSION:

This utility not only removes any typing errors but it also drastically reduces the time required to map the variables to the proper CT values. It helps in standardizing the CDISC data sets very efficiently and quickly.

REFERENCES

Greg Silva "Using IOM and Visual Basic in SAS(r) Program Development" *Proceedings of the Twenty eight Annual SAS® Users Group International Conference*.
<<http://www2.sas.com/proceedings/sugi28/032-28.pdf>>

ACKNOWLEDGEMENTS

We would like to thank Balaji Ayyappan and Nivethitha Manickavasagar for providing us excellent ideas and helping develop code to implement this utility.

RECOMMENDED READING

CDISC CONTROLLED TERMINOLOGY

<http://www.cdisc.org/standards/terminology/index.html>

SAS IOM

<http://support.sas.com/documentation/cdl/en/itechwcdg/61500/PDF/default/itechwcdg.pdf>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. You can contact us at:

Sandeep Juneja
ASG Inc.
2000 Regency Parkway, Suite # 675
Cary, NC 27518
Phone: (919) 653 3982
Email: sandeep.juneja@asg-inc.com

Vivek Mohan
ASG Inc.
2000 Regency Parkway, Suite # 675
Cary, NC 27518
Phone: (919) 653 3997
Email: vivek.mohan@asg-inc.com

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