ABSTRACT

Study Data Tabulation Model (SDTM) is an evolving global standard which is widely used for regulatory submissions. The automation of SDTM programming is essential to maximize the programming efficiency and improve the data quality. This paper intends to present some core programming logics to automatically create the Disease Response (RS) domain which is stated in CDISC SDTM IG v3.2 to represent the collected tumor data either quantitative measured or qualitative assessed in most of the clinical trials in oncology therapeutic area. The automation is realized by using SAS® macro facilities which include 1) the environment setting; 2) meta-data automation; 3) functionality oriented macros; 4) SUPPQUAL dataset automation; 5) structured end-to-end automation; 6) log check. The SAS® macros presented in this paper could also be applied to the automation of other SDTM oncology domains (i.e. TU, TR) or be extended to the domains in the SDTM Findings Observation Class.
SDTM domains are generated based on the multiple source datasets (raw data from INFORM) and SDTM specifications, which will involve complicated programming work.

To streamline the SDTM programming process, SAS macro facilities were developed to automatically map Raw Data to SDTM and help to write SAS codes, e.g.:
- read the attributes from specs
- write the attributes by itself
- convert data structure

**AUTOMATION’S GOAL**

- Readability
- Flexibility
- Feasibility
DISEASE RESPONSE (RS) - GENERAL DESCRIPTION

Represent the collected tumor data either quantitative measured or qualitative assessed, the response evaluation(s) determined from the data in Tumor Results (TR)

RS Example 1

<table>
<thead>
<tr>
<th>Row</th>
<th>STUDID</th>
<th>DOMAIN</th>
<th>SUBID</th>
<th>RSSEQ</th>
<th>RSLEV</th>
<th>RSTESTCD</th>
<th>RSTEST</th>
<th>RsCAT</th>
<th>RsORRS</th>
<th>RsSTRES</th>
<th>RsSTAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>1</td>
<td>TGESEP</td>
<td>Target Response</td>
<td>RCBST 1</td>
<td>PR</td>
<td>PR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>2</td>
<td>NIRTGST</td>
<td>Non-tumor Response</td>
<td>RCBST 1</td>
<td>SD</td>
<td>SD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>3</td>
<td>OVRRLST</td>
<td>Overall Response</td>
<td>RCBST 1</td>
<td>PR</td>
<td>PR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>4</td>
<td>TGESEP</td>
<td>Target Response</td>
<td>RCBST 1</td>
<td>NOT DONE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>5</td>
<td>NIRTGST</td>
<td>Non-tumor Response</td>
<td>RCBST 1</td>
<td>NOT DONE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>6</td>
<td>NRTNPRGF</td>
<td>Non-tumoral Progression</td>
<td>CLINICAL ASSESSMENT</td>
<td>Partial Effic</td>
<td>PD</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>ABC</td>
<td>RS</td>
<td>4444</td>
<td>7</td>
<td>OVRRLST</td>
<td>Overall Response</td>
<td>CLINICAL ASSESSMENT</td>
<td>PD</td>
<td>PD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Identifier Variables AND Timing Variables AND Topic and Qualifier Variables

**Vertical Structure EDS** OR **Horizontal structure EDS** OR **Complex structure EDS**

RS Domain
PRE-PROCESSING

Step 1: Environment & Domain Metadata Setting

/* Environment setting */
%setexecpath;

/* Domain metadata setting */
%let_sdtm=rs;
%sdtm_metadata (domain=&_sdtm);
%sdtm_shell (input_meta=&_sdtm._meta, domain=&_sdtm);

%setexecpath: automatically identify current working folder, initialize and set programming environment
%sdtm_metadata: read the attributes(order, length, type); parameter values and controlled terminology from sdtm specs, and SAS format generation
%sdtm_shell: convert the metadata attributes to a data step to create a shell dataset
Automation of SDTM Programming in Oncology Disease Response Domain
Yiwen Wang, Yu Cheng, Ju Chen
Eli Lilly and Company, China

PRE-PROCESSING
Step 2: Source Datasets Initialization
Map with Domain Shell

```plaintext
/*********************************************//*      Source Datasets Initialization       *//*********************************************/
%sdtm_varchg(inds=eds.rs1001,metads=meta_rs,outds=rs1001a);

/*********************************************/
/*        set RS with shell  *//*********************************************/
**data rs1001b;
  set rs_shell rs1001a;
run;*************/
```

`%sdtm_varchg`: change the input dataset attributes either by the spec or by user definition, considering multiple input datasets may contain different attributes for the same variable, this macro allows the user to be more flexible in dealing with attributes.
Step 3: Handling Identifier and Timing Variables

**Vertical Structure Transformation**

- **%sdtm_isodate**: covert date to ISO8601 date by using open SAS codes
- **%sdtm_relday**: extract Identifier variables and essential Timing variables from DM domain, and calculate study days of response assessment (RSDY)
- **%sdtm_tran**: transform the data structure from horizontal to vertical, matching with metadata from SDTM spec
  - **Key variables**: RSCAT RSTESTCD
  - **Corresponding variables**: RSEVAL RSGRPID RSORRES RSREASND RSTEST ...
- **%sdtm_visitnum**: map timing variables: visitnum, visit, epoch

```sas
data rs1;
  set rs1001b;
  if missing(rsdatdd) then rsdatdd=dd;
  if missing(rsdatmo) then rsdatmo=mo;
  if missing(rsdatyy) then rsdatyy=yy;
  %sdtm_isodate(yy=rsdatyy, mo=rsdatmo, dd=rsdatdd, hr='', mi='', dtc=RSDTC);
  ....
run;
/***********************************************************/
/* Merge with DM to get studyid, rfstdtc, usubjid etc */
/***********************************************************/
%sdtm_relday(inds=rs1, dmsubset=%nrstr(if first.%scan(&mergeby,-1,|), and last.%scan(&mergeby,-1,|) or input(scan(usubjid,-1,''),??best.))=input(subjid,??best.), mergeby=usubjid, addvar=studyid, rfstdtc=rfstdtc, outds=rs2_1);
/***********************************************************/
/* Vertical Structure transformation */
/***********************************************************/
%sdtm_tran(domain=&_sdtm, inds=rs2_1, outds=rs2_2, debug=Y);
%sdtm_visitnum(in=rs2_2, out=rs3_1, datec=rsdtc);
```
Automation of SDTM Programming in Oncology Disease Response Domain

Yiwen Wang, Yu Cheng, Ju Chen
Eli Lilly and Company, China

**POST-PROCESSING**

Step 4: Data Cut-Off if Needed

- Sort Observations & Variables Order
- Remove Format/Informat
- Add Domain Label

```plaintext
%sdtm_cutoff(in=rs3_1, out=rs3_2, datec=rsdtc); /*data cut-off based on needs*/
%sdtm_sortorder(domain=&_sdtm, inds=rs3_2, seqvar=&_sdtm.seq, outds=&_sdtm.1);
%sdtm_finalds(domain=&_sdtm, inds=&_sdtm.1, outds= sdtm.&_sdtm);
```

**%sdtm_cutoff:** data cut-off may be applied based on needs (e.g. safety review)

**%sdtm_sortorder:** add SEQID and sort the dataset either by the spec or by user definition (e.g. for debugging purpose)

**%sdtm_finalds:** final handling to generate final domain
- Sort Variables Order
- Remove Format / Informat
- Add Dataset Label
Step 5: SUPPQUAL (All Vertical Structure)

```plaintext
%sdm_metadata(domain=supp&_sdtm);
%sdtm_shell(input_meta=supp&_sdtm._meta, domain=supp&_sdtm);

proc sort data=supp&_sdtm; by usubjid &_sdtm.seq; run;

%sdtm_supp(domain=supp&_sdtm, inds=supp&_sdtm, meta_values=supp&_sdtm._meta_values, outds=supp&_sdtm.1);

%sdtm_sortorder(domain=supp&_sdtm,
               inds=supp&_sdtm.1(rename=(&_sdtm.seq=idvarvalx)),
               numpar=idvarvalx,
               outds=supp&_sdtm.2,
               debug=Y);

%sdtm_finalds(domain=supp&_sdtm, inds=supp&_sdtm.2, outds=sdtm.supp&_sdtm);

%thePostProc; => LOG CHECK
```
Enable data mapping automation: The macro-based tools (SDTM utility macros) allow data to be mapped automatically according to the required specifications. This automation tool works like a reader and automatically generates program codes as long as the specification is written in a machine-readable manner. The tools can apply to different SDTM domains including SUPPQUAL and RELREC.

Standardize programming style: Streamlined the SDTM programming logic and standardized the coding style. The standardized templates for the key safety domains and Oncology efficacy domains were created which can be re-used and as a result reduces the variability of quality among programmers.

Engage a broader range of applications: The developed macros could be used across different studies, therapeutic areas and for different delivery needs.
Acknowledgments
The authors would like to thank all of my colleagues who reviewed this paper and provided insight comments.

Contact Information
Your comments and questions are valued and encouraged. Contact the authors at:
Yiwen Wang, E-mail: wang_yi_wen_even@lilly.com
Yu Cheng, E-mail: cheng_yu_ella@lilly.com
Ju Chen, E-mail: chen_ju@lilly.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.

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