Paper 072-2008

Using ExcelXP to Display SDTM Metadata And More

Christine Teng, Merck Research Labs, Merck & Co., Inc., Rahway, NJ
Shaoan Yu, Merck Research Labs, Merck & Co., Inc., Rahway, NJ

ABSTRACT

The Clinical Data Interchange Standards Consortium (CDISC) has established worldwide industrial standards that support platform-independent data standards that enable information system interoperability to improve medical research and related areas of healthcare. Many pharmaceutical companies have started implementation of CDISC data models such as Study Data Tabulation Model (SDTM) and Analysis Data Model (ADaM). As variable names and attributes are standardized, it is very easy to write programs to collect information about the data. The purpose of this paper is to demonstrate how standardization simplifies work processes. SAS®9 provides several approaches to create Excel output. There is an experimental tagset called ExcelXP that is available for download from the ODS Markup Resources site at http://support.sas.com/rnd/base/topics/odsmarkup/. The SAS®9 ExcelXP tagset generates XML output that conforms to the Microsoft XML Spreadsheet Specification ("XML Spreadsheet Reference", Microsoft Corp.). One can create XML output on a UNIX or Windows platform and the XML output can be read by EXCEL 2002 and later releases. ExcelXP provides simple options to create multiple worksheets. In this paper, the ExcelXP tagset is used in conjunction with the SAS Dictionary to create metadata documentation for a group of SDTM domains from a mock clinical trial project for demonstration.

SAS®9, Windows, Intermediate Level
Key Words: ExcelXP, Tagset, SAS Dictionary, PROC SQL

INTRODUCTION

The SAS®9 ExcelXP tagset generates XML output that conforms to the Microsoft XML Spreadsheet Specification ("XML Spreadsheet Reference", Microsoft Corp.). It provides the functionality to create multiple worksheets in a workbook as well as multiple tables within a single worksheet. These features are very useful for creating metadata documentation where each domain has its own worksheet with label. It enables quicker accessibility to locate the information for a group of domains. With SAS DICTIONARY and PROC SQL, the metadata documentation can be created without hard coding. The details of using PROC SQL and SAS DICTIONARY will not be covered in this paper. For more information regarding the SAS DICTIONARY and PROC SQL, please refer to the SAS manuals.

This paper is not a tutorial about the ExcelXP tagset. Rather, it demonstrates another application using the ExcelXP tagset. The detailed tutorials and references for the ExcelXP tagset can be found at the references section of this paper. In order to control the appearance of the output within Excel, PROC TEMPLATE can be used to create a style template. A template defines how to format output produced by a procedure or data step. For information about PROC TEMPLATE, please consult SAS®9 online documentation site at : http://support.sas.com/onlinedoc/913.

SAS provides many standard templates that allow for customization. To see a list of templates provided by SAS, (1) go to the Results windows, (2) right click on Results and select Template, (3) expand sashelp.Tmplmst (See Table-1 in Appendix). In the macro that builds the metadata documentation, we created a customized style template that uses certain fonts, colors and spacing inside my Excel workbook. This step is not required to use ExcelXP. However, style template makes the output more presentable.

DESIGN REQUIREMENTS

The following are the requirements for the metadata documentation:

A. Create a macro program with one parameter:
DATADIR is used to assign the input library name.

```
%ls_datastruc(datadir = datadir)
```

B. Create a metadata table inside a worksheet for each domain within the datadir library. The label of each domain should be listed first, followed by the attributes of the variables. (See Table-2 in Appendix)

C. If any variable in the domain has length of label > 40, length of variable name > 8 or length of a character variable > 200, a note will be shown. (See Table-3 in Appendix)

D. If a domain contains a variable name ending with testcd, create a second table after the metadata table in the same worksheet. (See Table-4 in Appendix)

E. After all worksheets of domains are created, create a global dictionary for all test codes defined in the project. (See Table-5 in Appendix)

IMPLEMENTATION

Since the ExcelXP tagset is still evolving, there are some limitations and hence its functionality may be changed in the future. It is recommended that the user always download the latest update to verify the changes and enhancements. To use the ExcelXP tagset, first download the latest ExcelXP tagset from the SAS ODS MARKUP page. This page also provides links to documentation for using and customizing tagsets. For this exercise, we use the ExcelXP Tagset version dated June 2007. Before using the ExcelXP tagset, check the codes or execute the following to see a list of options available in the ExcelXP tagset:

```
ODS tagsets.excelxp file = "test.xml"  options(doc="help");
```

Quick Reference for the TAGSETS.EXCELXP Tagset can also be found in http://support.sas.com/rnd/base/ods/odsmarkup/excelxp_help.html.

Under the pre-configuration part of the requirement **A** below, only specifications are described since coding for this part is not the focus of this paper. The sections, where the worksheets are built, provide more detailed coding information.

REQUIREMENT A – Create a macro program with one parameter.

```
%MACRO ls_datastruc(datadir= );
```

*Pre-configuration before building the worksheets;

NULLTBL – A table used to build a header in the global worksheets for the requirement E.

TESTTBL – A table that contains all of the TESTCD and the associated TEST description. The TESTCD values are collected from each domain that has a variable ending with TESTCD. This is used in the requirement E.

*Set up the style template;

```
proc template;
     define style styles.XLStatistical;
     parent = styles.Statistical;
           :
           :
 ods listing close;
```

*Set up the workbook;

Include the ExcelXP tagset code

```
%let _ODSDEST=tagsets.ExcelXP;
ods &_ODSDEST   path = "c:\temp\excelXP"
                file = "test.xml"
               style = XLStatistical;
```

*Build the worksheets (see requirements below);

```
ods &_ODSDEST close;
```

```
%MEND;
```

REQUIREMENT B – Create a metadata table inside a worksheet for each domain defined in the macro parameter datadir.

```
proc sql noprint;
      %*dsetname contains all domains in the libname datadir;
       select memname into :dsetname separated by '+'
       from dictionary.tables
       where libname="&datadir" and memtype="DATA" ;

      %*examlst contains all domains that have a variable name ending with TESTCD;
       select memname into :examlst separated by ' '
       from   dictionary.columns
       where  libname="&datadir" and memtype="DATA" and name like '%TESTCD' ;
quit;

%let num=1;
%let list = %upcase(%scan(&dsetname, &num, +));

%*Use Do-While loop to create individual worksheet;
%do %while (&list. ne  );
      %*Create worksheet with defined options;
        ods &_ODSDEST options(absolute_column_width = "6, 16, 10, 45, 25"
                                    sheet_interval = "none"
                                        sheet_name = "&list");

      %*Print domain name and label at the beginning of the sheet;
        proc sql;
             select ' ', substr(memname,1) as Data_Set, ' ',  substr(memlabel,1) as
                    Data_Set_Label, ' '
             from   dictionary.tables
             where  libname = "&datadir" and memtype = "DATA" and memname = "&list";
        quit;

      %*Print domain columns and attributes information;
        proc sql;
             select int(varnum) as Pos, upcase(name) as VarName,
                    propcase(catx('',type,put(length, best5.))) as TypeLen,
                    substr(label,1) as Label,  ' ' as Deriviation_Comments
             from   dictionary.columns
             where  libname = "&datadir" and memtype = "DATA" and memname = "&list"
             order by varnum;
        quit;
```

REQUIREMENT C – If any variable in the domain has length of label > 40, length of variable name > 8 or length of character > 200, a note will be shown.

```
         proc sql;
              select  int(varnum) as Pos, upcase(name) as VarName,
                      propcase(catx('',type,put(length, best4.))) as TypeLen,
                      substr(label,1) as Label,
                      case
                      when length(label) > 40 then 'length of label > 40'
                      when length(name) > 8 then 'length of variable name > 8'
                      when length > 200  then 'length of character value > 200'
                      else ' '
                      end as Check_Length
              from   dictionary.columns
              where  libname="&datadir" and memtype="DATA" and memname="&list"
              order by varnum;
         quit;
```

REQUIREMENT D – If a domain contains a variable ending with testcd, create a second table after the metadata table in the same worksheet.

```
    %if %index(&examlst., &list.) %then %do;
       proc sql;
            select distinct "&list" label='Domain', &list.testcd label='Test Code',
                  ' ', &list.test Label='Test Description'
            from &datadir..&list.;
       quit;
    %end;

    %*Ready to build the next worksheet;
    %let num = %eval(&num + 1);
    %let list = %upcase(%scan(&dsetname, &num, '+'));
%end;
```

REQUIREMENT E – Create a global dictionary for all test codes defined in the project.

```
ods &_ODSDEST options(absolute_column_width="10, 15, 55, 20"
                          sheet_interval="none"
                              sheet_name="TestCode");

proc sql;
     select ' ' label='Purpose: ',' ' label='List of Tests'  from NULLTBL;
     select distinct domain label='Source', paramcd Label='Parameter Name',
            param  label='Parameter Description',
            case
            when paramcd eq '' then 'TESTCD value is missing'
            when param eq '' then 'TEST value is missing'
            else ' '
            end as Check_Missing
     from testtbl
     order by domain;
quit;
```

As shown above, with the use of PROC SQL, SAS DICTIONARY tables, standardized SDTM structure, and simple ExcelXP options, we are able to quickly build up the workbook with multiple worksheets that contain the metadata information for a list of domains. This information is very useful to help learn and verify a project database design.

SUMMARY

ExcelXP is one of the many tools in SAS to create Excel output. It allows simple configurations to generate Excel output. With SAS Dictionary tables, we found it very useful and easy to create documentation for quality assurance purposes. Please visit the SAS support website at http://support.sas.com/rnd/base/ods/odsmarkup/index.html for additional ExcelXP tagset information and examples.

REFFERENCES

DelGobbo, V. 2006. "Creating AND Importing Multi-Sheet Excel Workbooks the Easy Way with SAS®". Proceedings of the Thirty-First Annual SAS Users Group International Conference, 31. CD-ROM. Paper 115.

Gebhart, E. 2005. " ODS Markup: The SAS® Reports You've Always Dreamed Of ". Proceedings of the Thirtieth Annual SAS Users Group International Conference, 30. CD-ROM. Paper 85.

Zender, C. 2005. "The Power of Table Templates and DATA _NULL_". Proceedings of the Thirtieth Annual SAS Users Group International Conference, 30. CD-ROM. Paper 88.

PharmaSUG 2006 Paper: "Simple Ways to Use PROC SQL and DICTIONARY TABLES to Verify Data Structure of the Electronic Submission Data Sets" By Christine S. Teng and Wenjie Wang.

SAS Macro Language: Reference

SAS SQL Procedure User's Guide

ACKNOWLEGEMENTS

The author would like to thank the management team for their encouragement and review of this paper.

TRADEMARKS

CONTACT INFORMATION

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

Christine Teng Merck & Co., Inc. Rahway, NJ 07065 christine_teng@merck.com	
Shaoan Yu Merck & Co., Inc. Rahway, NJ 07065 Shaoan_yu@merck.com	

APPENDIX

Table – 1 (Available Tagsets in SAS®9)

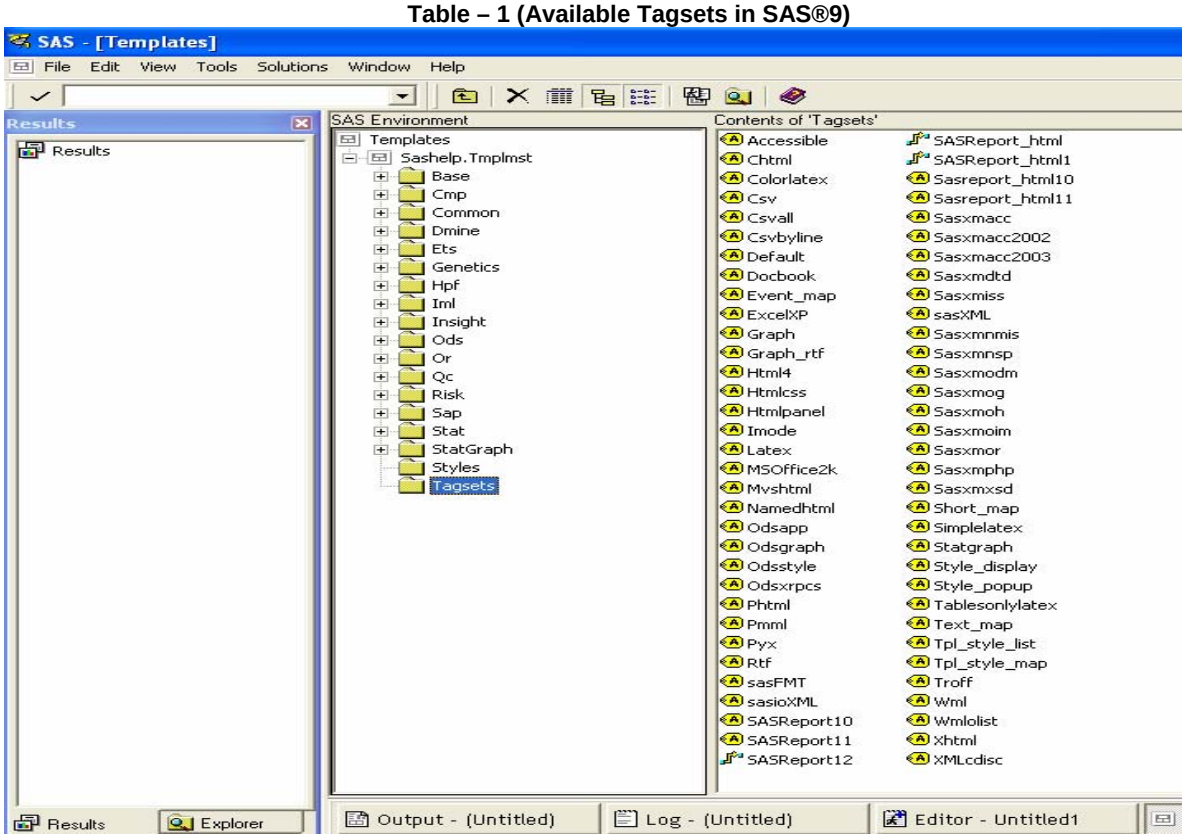

Table – 2 (Requirement B)

	A	B	C	D	E
1		**Data_Set**		**Data_Set_Label**	
2		AE		Adverse Events Data Set	
3					
4	**Pos**	**VarName**	**TypeLen**	**Label**	**Check_Length**
5	1	STUDYID	Char200	Study Identifier	
6	2	DOMAIN	Char2	Domain Abbreviation	
7	3	USUBJID	Char200	Unique Subject Identifier	
8	4	AESEQ	Num8	Sequence Number	
9	5	AEGRPID	Char200	Group ID	
10	6	AEREFID	Char200	Reference ID	
11	7	AESPID	Char200	Sponsor-Defined Identifier	
12	8	AETERM	Char200	Reported Term for the Adverse Event	
13	9	AEMODIFY	Char200	Modified Reported Term	
14	10	AEDECOD	Char200	Dictionary-Derived Term	
15	11	AECAT	Char200	Category for Adverse Event	
16	12	AESCAT	Char200	Subcategory for Adverse Event	
17	13	AEOCCUR	Char2	Adverse Event Occurrence	
18	14	AEBODSYS	Char80	Body System or Organ Class	
19	15	AELOC	Char200	Location of the Reaction	
20	16	AESEV	Char8	Severity/Intensity	
21	17	AESER	Char2	Serious Event	
22	18	AEACN	Char17	Action Taken with Study Treatment	
23	19	AEACNOTH	Char17	Other Action Taken	
24	20	AEREL	Char200	Causality	
25	21	AERELNST	Char200	Relationship to Non-Study Treatment	
26	22	AEPATT	Char200	Pattern of Adverse Event	
27	23	AEOUT	Char33	Outcome of Adverse Event	
28	24	AESCAN	Char2	Involves Cancer	
29	25	AESCONG	Char2	Congenital Anomaly or Birth Defect	

AE CM CO DM DS EG EX IE LB MH ML PC PE PR QS SC SE SU SV TA TE

Table – 3 (Requirement C)

	A	B	C	D	E
1		**Data_Set**		**Data_Set_Label**	
2		LB		Laboratory Findings Data Set	
3					
4	**Pos**	**VarName**	**TypeLen**	**Label**	**Check_Length**
5	1	STUDYID	Char200	Study Identifier	
6	2	DOMAIN	Char2	Domain Abbreviation	
7	3	USUBJID	Char200	Unique Subject Identifier	
8	4	LBSEQ	Num8	Sequence Number	
9	5	LBGRPID	Char200	Group ID	
10	6	LBREFID	Char200	Specimen ID	
11	7	LBSPID	Char200	Sponsor-Defined Identifier	
12	8	LBTESTCD	Char200	LAB Test or Examination Short Name	
13	9	LBTEST	Char200	LAB Test or Examination Name	
14	10	LBCAT	Char255	Category for Lab Test	length of character value > 200
15	11	LBSCAT	Char200	Subcategory for Lab Test	
16	12	LBORRES	Char200	Result or Finding in Original Units	
17	13	LBORRESU	Char200	Original Units	
18	14	LBORNRLO	Char200	Reference Range Lower Limit in Original Unit	length of label > 40
19	15	LBORNRHI	Char200	Reference Range Upper Limit in Original Unit	length of label > 40
20	16	LBSTRESC	Char200	Character Result/Finding in Std Format	
21	17	LBSTNRC	Char200	Reference Range for Char Rslt-Std Units	
22	18	LBSTRESN	Num8	Numeric Result/Finding in Standard Units	
23	19	LBSTRESU	Char200	Standard Units	
24	20	LBSTNRLO	Num8	Reference Range Lower Limit-Std Units	
25	21	LBSTNRHI	Num8	Reference Range Upper Limit-Std Units	
26	22	LBNRIND	Char200	Reference Range Indicator	
27	23	LBSTAT	Char8	Lab Status	
28	24	LBREASND	Char200	Reason Test Not Done	
29	25	LBNAM	Char200	Vendor Name	

AE CM CO DM DS EG EX IE **LB** MH ML PC PE PR QS SC SE SU SV TA TE

Table – 4 (Requirement D)

	A	B	C	D	E
1		**Data_Set**		**Data_Set_Label**	
2		LB		Laboratory Findings Data Set	
79					
80	**Domain**	**Test Code**		**Test Description**	
81	LB	ALB		Albumin	
82	LB	ALB1		ALB	
83	LB	ALP		Alkaline Phosphatase	
84	LB	ALT		Alanine Aminotransferase	
85	LB	APTT		Activated Partial Thromboplastin Time	
86	LB	AST		Aspartate Aminotransferase	
87	LB	BASO		Basophils	
88	LB	BILDIR		Direct Bilirubin	
89	LB	BILI		Bilirubin	
90	LB	BLD		Urine Blood	
91	LB	BUN		Blood Urea Nitrogen	
92	LB	CA		Calcium	
93	LB	CAST		Casts	
94	LB	CAST1		UCAST	
95	LB	CASTFAT		Fatty Casts	
96	LB	CASTGRA		Granular Casts	
97	LB	CASTLEUK		Leukocyte Casts	
98	LB	CHOL		Cholesterol	
99	LB	CHOL1		CHOL	
100	LB	CL		Chloride	
101	LB	CRAMMUR		Ammonium Urate Crystals	
102	LB	CREAT		Creatinine	
103	LB	EOS		Eosinophils	
104	LB	EPIC		Epithelial Cells	
105	LB	GGT		Gamma Glutamyl Transferase	

AE CM CO DM DS EG EX IE **LB** MH ML PC PE PR QS SC SE SU SV TA TE

Table – 5 (Requirement E)

	A	B	C	D
1	**Purpose:**	**List of Tests**		
2				
3				
4	**Domain**	**TESTCD**	**TEST**	**Check_Missing**
5	EG	ARATE	Atrial Rate	
6	EG	INTP	Interpretation	
7	EG	PR	PR Interval	
8	EG	QRS	QRS Interval	
9	EG	QRSA	QRS Axis	
10	EG	QT	QT Interval	
11	EG	QTC	QTc Interval	
12	EG	QTCB	QTc Interval Bazett	
13	EG	QTCF	QTc Interval Fridericia	
14	EG	RR	RR Interval	
15	EG	VRATE	Ventricular Rate	
16	IE	EX1		TEST value is missing
17	IE	EX1	Smokes 4 packs a week	
18	IE	IN1		TEST value is missing
19	IE	IN1	Age Between 18 and 64?	
20	LB	ALB	Albumin	
21	LB	ALB1	ALB	
22	LB	ALP	Alkaline Phosphatase	
23	LB	ALT	Alanine Aminotransferase	
24	LB	APTT	Activated Partial Thromboplastin Time	
25	LB	AST	Aspartate Aminotransferase	
26	LB	BASO	Basophils	
27	LB	BILDIR	Direct Bilirubin	
28	LB	BILI	Bilirubin	
29	LB	BLD	Urine Blood	

EG EX IE LB MH ML PC PE PR QS SC SE SU SV TA TE TV VS **TestCode**