

## Using SAS® Clinical Data Integration to Roundtrip a Complete Study Study Metadata (Define-XML) and Study Data (Dataset-XML)

Ken Ellis, SAS Institute Inc., Cary, NC, USA

### ABSTRACT

SAS Clinical Data Integration 2.6 now supports the complete CDISC Define-XML 2.0 specification for both import from, and creation of define.xml files. Support for value-level metadata and supplemental documents, as well as support for the new CDISC Dataset-XML specification have been added and give the user a powerful means by which to import and create complete study definitions. Topics covered include:

- import of CDISC Define-XML file metadata, including domains, value-level metadata, supplemental documents, computational algorithms and code lists
- import of accompanying domain data from CDISC Dataset-XML files
- creation of a complete CDISC Define-XML file including all imported metadata (roundtrip)
- creation of CDISC Dataset-XML files from the imported domain tables (roundtrip)

### INTRODUCTION

The CDISC Define-XML 2.0<sup>1</sup> specification (define.xml) provides a machine readable description of datasets used for submission to the FDA. The new CDISC Dataset-XML<sup>2</sup> specification (dataset.xml) provides a means for describing the underlying data associated with CDISC Define-XML 2.0 compliant dataset definitions.

SAS Clinical Data Integration 2.6<sup>3</sup>, in combination with SAS Clinical Standards Toolkit 1.7<sup>4</sup>, provides a rich set of tools for reading and writing both define.xml and dataset.xml files. This paper will focus on those tools and the processes that facilitate working with these files. A sample work flow will be developed that roundtrips a define.xml file and its associated dataset.xml files. It is assumed that the user has some familiarity with SAS Clinical Data Integration and the CDISC specifications referenced as in-depth discussions of each is beyond the scope of this paper. (See the References section for more information.)

## THE TOOLS

SAS Clinical Data Integration provides the following tools for reading and writing define.xml and dataset.xml files. We will look at them more in-depth as we encounter them in our workflow example:

1. **Study Level**
  - a. **Study from Define...**

Creates a basic study shell using the selected folder template. Values for the study name, description and protocol title are derived from the define.xml file. The user can optionally associate a data standard and library with the new study.
  - b. **Domain(s) from Define...**

Creates empty SAS table shells for each SDTM domain selected for import. All table level and column metadata are derived from the define.xml file.
  - c. **Analysis Dataset(s) from Define...**

Creates empty SAS table shells for each ADaM table selected for import. All table level and column metadata are derived from the define.xml file.
  - d. **Codelist Table from Define...**

Creates a single SAS table for all codelists selected for import. The generated table can be associated with a terminology package that is associated with the study.
2. **Table Level**
  - a. **Value Level Metadata from Define...**

Imports value level metadata found in the define.xml file into a pre-defined SAS data table that can be associated with the study via the study properties.
  - b. **Supplemental Documents from Define...**

Imports supplemental document data found in the define.xml file into a pre-defined SAS data table that can be associated with the study via the study properties.
3. **Clinical Transforms**
  - a. **CDISC-Define Creation**

Generates a new define.xml file from the selected inputs.
  - b. **CDISC-Define Validation**

Validates that a define.xml file adheres to a schema.
  - c. **CDISC-Dataset-XML Creation**

Creates new dataset-xml files for each selected table.
  - d. **CDISC-Dataset-XML Import**

Creates a SAS data table representation of a dataset-xml file.

## SETUP

There are several administrative tasks that must first be done before continuing. They are typically the responsibility of a Clinical Administrator and are beyond the scope of this paper. The basic necessities are as follows:

1. Import and activate a Data Standard
2. Create a suitable folder structure template for your study data
3. Create a Terminology Package for any codelist tables and associate it with the study
4. Create a Value Level Metadata table (empty) and associate it with the study
5. Create a Supplemental Documents table (empty) and associate it with the study.

## THE IMPORT PROCESS

We will develop a basic process that imports the data and metadata associated with a define.xml file and its associated dataset-xml files. We will examine the parts of each where applicable so that you gain some familiarity with not only the process, but what is being created. Note that the tools provided for importing a define.xml file are very granular and thus allow the user to pick and choose the parts of the define.xml that he or she might want to import. Finally, we will export (roundtrip) our study data back into a new define.xml file and associated dataset-xml files and will validate that the generated define.xml conforms to a schema. Let's get started...

### 1. Generate a new SAS Clinical Data Integration Study

Use the **New Study from Define...** context menu to launch the **New Study from Define** wizard (Figure 1). The wizard will first prompt for the location of a define.xml to use as an input source. Once selected, the wizard will validate that the file is indeed a define.xml file and then import the study name, description and protocol title values from the define.xml file's **GlobalVariables** element (Figure 2). Subsequent pages of the wizard allow the opportunity to change these default values, associate a folder template, and optionally associate a data standard (recommended), a library and a terminology package with the study. Once created, the new study will be displayed in the SAS Clinical Data Integration Folder tree.

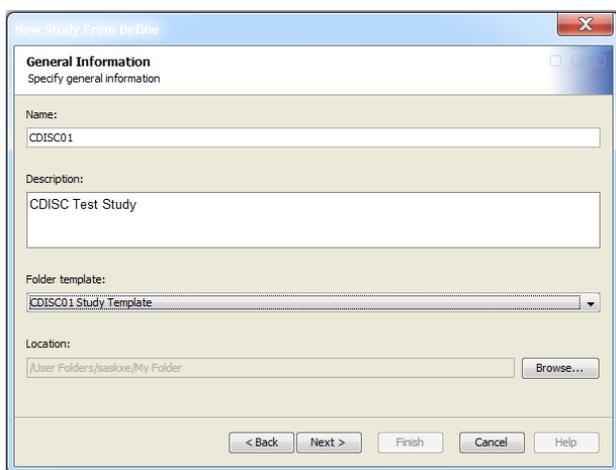


Figure 1. New Study from Define Wizard

```
<GlobalVariables>
  <StudyName>CDISC01</StudyName>
  <StudyDescription>
    CDISC Test Study
  </StudyDescription>
  <ProtocolName>CDISC01</ProtocolName>
</GlobalVariables>
```

Figure 2. **GlobalVariables** Element

### 2. Import the SDTM Domain definitions

Next, import the SDTM domain definitions and create empty SAS data table shells for each domain definition found in the define.xml file. All metadata associated at both the table and column level will be imported and stored in the SAS Clinical Data Integration metadata repository. Note that in order to reference permanent data tables on the file system that are associated with the metadata representations in SAS Clinical Data Integration, you will need to associate a Base SAS Library definition as part of the stored table metadata. It is also important to note that physical tables will not be created as part of the domain import process.

Use the **New Domain(s) from Define...** context menu wizard to launch the **New Domain(s) from Define** wizard (Figure 3). The wizard will prompt you for the location of a define.xml to use as an input source and a location to store the table metadata definition within SAS Clinical Data Integration. Next you will be required to select a data standard to associate with the table(s) you are about to import. Subsequent wizard pages allow you to select any (or all) domain definitions found and to optionally associate a SAS library definition for reference to permanent table storage. Upon completion of the wizard your domain table definitions will be created in the SAS Clinical Data Integration metadata repository (Figure 3, 4).

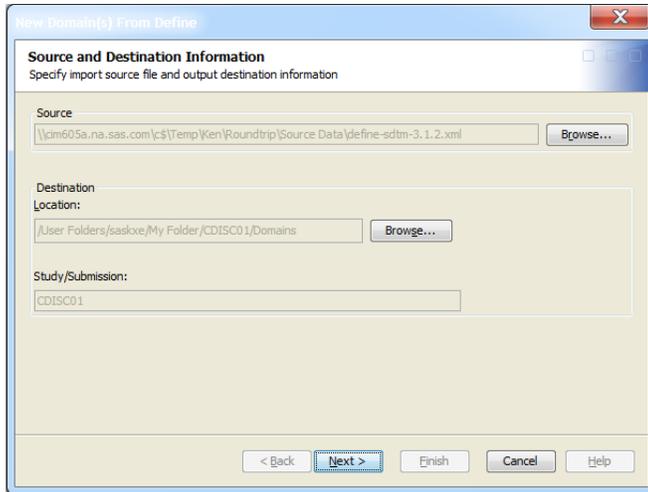


Figure 3. New Domain(s) from Define Wizard

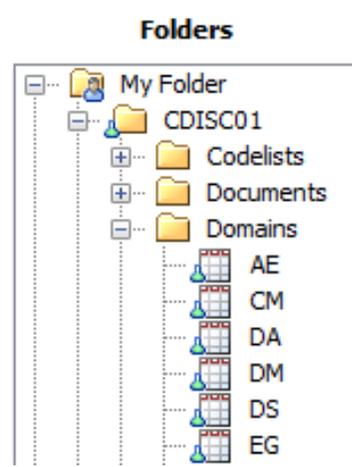


Figure 4. Partial Listing of Domains

A quick look at the AE domain's column properties (Figure 5) from within SAS Clinical Data Integration shows that the **ItemGroupDef** element's attributes were imported correctly (Figure 6).

#	Name	Description	Type
1	STUDYID	Study Identifier	Character
2	DOMAIN	Domain Abbreviation	Character
3	USUBJID	Unique Subject Identifier	Character
4	AESSEQ	Sequence Number	Numeric
5	AESPID	Sponsor-Defined Identifier	Character
6	AETERM	Reported Term for the Adverse Event	Character
7	AEMODIFY	Modified Reported Term	Character
8	AEDECOD	Dictionary-Derived Term	Character
9	AEBODSYS	Body System or Organ Class	Character
10	AESEV	Severity/Intensity	Character
11	AESER	Serious Event	Character
12	AEACN	Action Taken with Study Treatment	Character
13	AEREL	Causality	Character
14	AESTDTC	Start Date/Time of Adverse Event	Character
15	AEENDTC	End Date/Time of Adverse Event	Character
16	AESTDY	Study Day of Start of Adverse Event	Numeric
17	AEENDY	Study Day of End of Adverse Event	Numeric
18	AENRF	End Relative to Reference Period	Character

Figure 5. Column Properties for the AE Domain

```

<ItemGroupDef OID="IG.AE" Name="AE" Repeating="Yes" IsReferenceData="No" SASDatasetName="AE"
  Domain="AE" Purpose="Tabulation" def:Class="EVENTS" def:Structure="One record per adverse event per subject"
  def:ArchiveLocationID="LF.AE">
  <Description>
    <TranslatedText xml:lang="en">Adverse Events</TranslatedText>
  </Description>
  <ItemRef ItemOID="IT.AE.STUDYID" Mandatory="Yes" OrderNumber="1" KeySequence="1"/>
  <ItemRef ItemOID="IT.AE.DOMAIN" Mandatory="Yes" OrderNumber="2"/>
  <ItemRef ItemOID="IT.AE.USUBJID" Mandatory="Yes" OrderNumber="3" KeySequence="2" MethodOID="MT.AE.USUBJID"/>
  <ItemRef ItemOID="IT.AE.AESEQ" Mandatory="Yes" OrderNumber="4" MethodOID="MT.AE.AESEQ"/>
  <ItemRef ItemOID="IT.AE.AESPID" Mandatory="No" OrderNumber="5"/>
  <ItemRef ItemOID="IT.AE.AETERM" Mandatory="Yes" OrderNumber="6"/>
  <ItemRef ItemOID="IT.AE.AEMODIFY" Mandatory="No" OrderNumber="7"/>
  <ItemRef ItemOID="IT.AE.AEDECOD" Mandatory="Yes" OrderNumber="8" KeySequence="3"/>
  <ItemRef ItemOID="IT.AE.AEBODSYS" Mandatory="No" OrderNumber="9"/>
  <ItemRef ItemOID="IT.AE.AESEV" Mandatory="No" OrderNumber="10"/>
  <ItemRef ItemOID="IT.AE.AESER" Mandatory="No" OrderNumber="11"/>
  <ItemRef ItemOID="IT.AE.AEACN" Mandatory="No" OrderNumber="12"/>
  <ItemRef ItemOID="IT.AE.AEREL" Mandatory="No" OrderNumber="13"/>
  <ItemRef ItemOID="IT.AE.AESTDTC" Mandatory="No" OrderNumber="14" KeySequence="4"/>
  <ItemRef ItemOID="IT.AE.AEENDTC" Mandatory="No" OrderNumber="15"/>
  <ItemRef ItemOID="IT.AE.AESTDY" Mandatory="No" OrderNumber="16" MethodOID="MT.AE.AESTDY"/>
  <ItemRef ItemOID="IT.AE.AEENDY" Mandatory="No" OrderNumber="17" MethodOID="MT.AE.AEENDY"/>
  <ItemRef ItemOID="IT.AE.AEENRF" Mandatory="No" OrderNumber="18"/>
  <def:leaf ID="LF.AE" xlink:href="ae.xpt">
    <def:title>ae.xpt</def:title>
  </def:leaf>
</ItemGroupDef>

```

Figure 6. The **ItemGroupDef** Element for the AE Domain

### 3. Import the Codelists

SAS Clinical Data Integration can manage two distinct types of codelist data; those imported directly from SAS Clinical Standards Toolkit, and external codelist tables such as those imported from a define.xml or other source. The process of associating a codelist table with a study is:

- a. Create a Terminology Package
- b. Associate the Terminology Package with the study
- c. Associate the imported codelist table with the Terminology Package

These steps are not within the scope of this paper and it is assumed that the user is familiar with the processes involved. It is also important to note SAS Clinical Data Integration does not manage any metadata associated with an imported codelist table other than the SAS Base Library association to the physical data table itself.

Use the **New Codelist Table from Define...** context menu wizard to launch the **New Codelist Table from Define** wizard. The wizard will prompt you for the location of a define.xml to use as input and a location to store the codelist table's metadata definition within SAS Clinical Data Integration. The next wizard page will display all of the codelists that are available to import. Select any or all depending on your need (Figure 7). The final wizard page asks you to provide a codelist name, optional description and a required SAS Library used to create the permanent physical SAS data table. Upon completion of the wizard the new codelist table is created and registered with SAS Clinical Data Integration. Remember that you must also register the new codelist table with a SAS Terminology Package for it to be associated with a study.

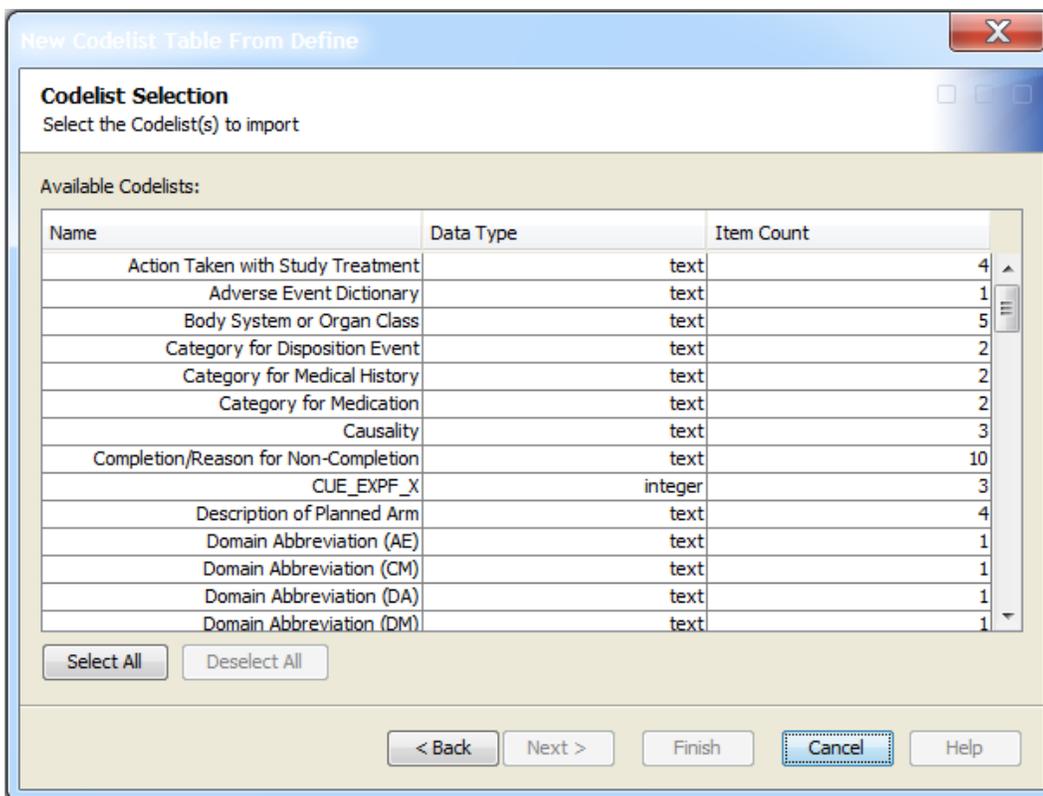


Figure 7. Codelist Selection Wizard Page

Finally, a quick look at a sample of the generated codelist table (Figure 8) shows that the **CodeList** element was imported correctly from the define.xml file (figure 9).

#	StudyVersion	CodeListName	CodeListNCI Code	CodedValueChar	CodedValueNCI Code
1	MDV.CDISCO1.SD...	Action Taken with Study Treatment ...	C66767	DOSE NOT CHANGED ...	C49504
2	MDV.CDISCO1.SD...	Action Taken with Study Treatment ...	C66767	DOSE REDUCED ...	C49505
3	MDV.CDISCO1.SD...	Action Taken with Study Treatment ...	C66767	DRUG INTERRUPTED ...	C49501
4	MDV.CDISCO1.SD...	Action Taken with Study Treatment ...	C66767	DRUG WITHDRAWN ...	C49502

Figure 8. Partial Listing of the Generated Codelist Table

```
<CodeList OID="CL.ACN" Name="Action Taken with Study Treatment" DataType="text" SASFormatName="$ACN">
  <EnumeratedItem CodedValue="DOSE NOT CHANGED" OrderNumber="1">
    <Alias Context="nci:ExtCodeID" Name="C49504"/>
  </EnumeratedItem>
  <EnumeratedItem CodedValue="DOSE REDUCED" OrderNumber="2">
    <Alias Context="nci:ExtCodeID" Name="C49505"/>
  </EnumeratedItem>
  <EnumeratedItem CodedValue="DRUG INTERRUPTED" OrderNumber="3">
    <Alias Context="nci:ExtCodeID" Name="C49501"/>
  <EnumeratedItem CodedValue="DRUG WITHDRAWN" OrderNumber="4">
    <Alias Context="nci:ExtCodeID" Name="C49502"/>
  </EnumeratedItem>
  <Alias Context="nci:ExtCodeID" Name="C66767"/>
</CodeList>
```

Figure 9. The **CodeList** Element for “Action Taken with Study Treatment”

#### 4. Import Value Level Metadata and Supplemental Documents

The processes for importing Value Level Metadata and Supplemental Documents from a define.xml file are identical other than the wizards used. The process steps are:

- a. Create an empty Value Level Metadata or Supplemental Documents table
- b. Associate the table with the study
- c. Import data from a define.xml into the table

Note that SAS Clinical Data Integration does not manage any metadata associated with the table or table columns other than the metadata link to the table itself. Users are free to edit the data within the tables as long as the table column integrity remains intact.

A Clinical Administrator would first create an empty table by selecting Study tab from within the **Study Properties** page, then selecting the appropriate **Add...** button for Value Level Metadata Table or Supplemental Documents Table (Figure 10). For this example we will trace the path of creating a new empty Value Level Metadata table.

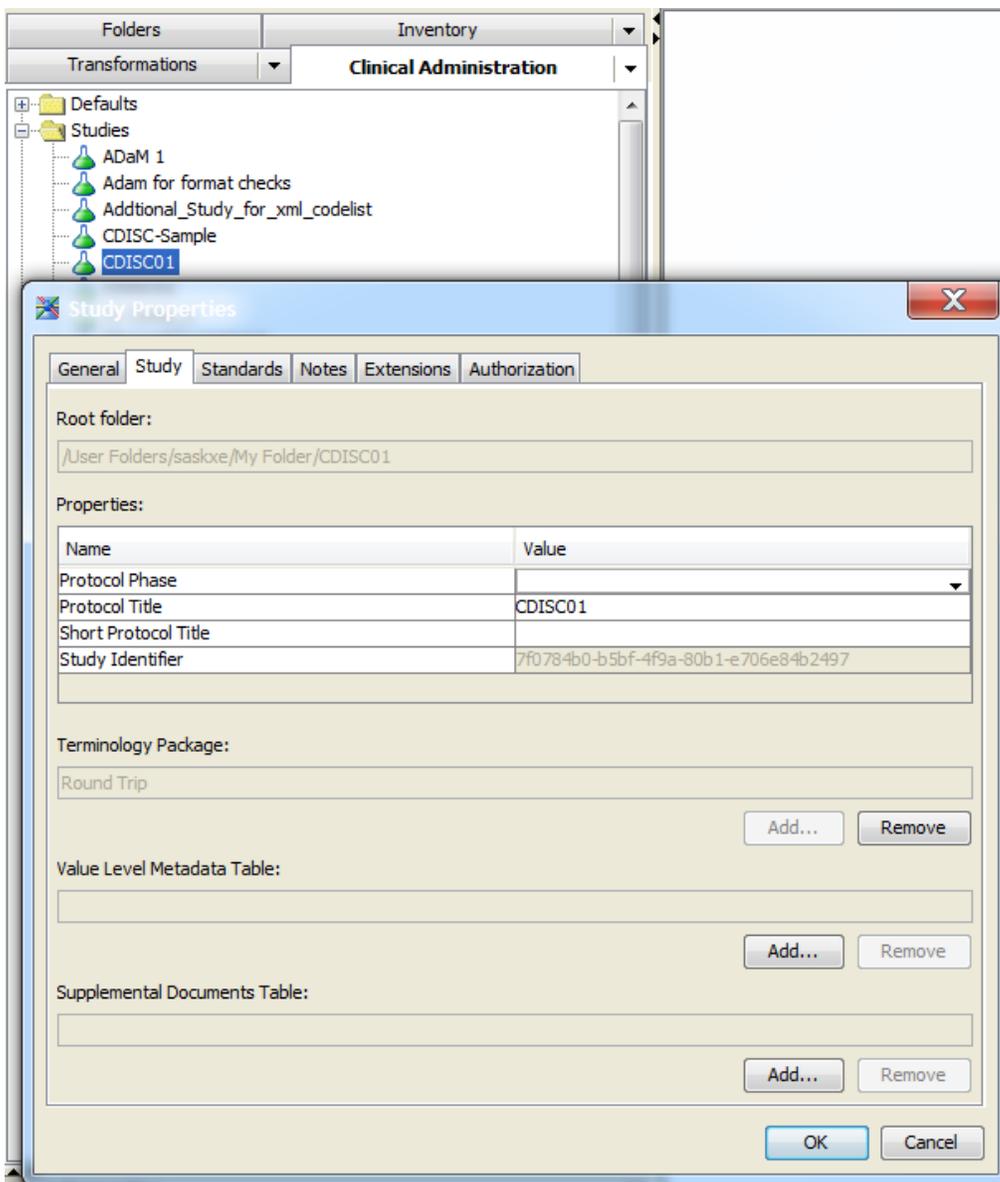


Figure 10: The Study Properties “Study” Tab

The **Available Value Level Metadata Tables** window is displayed and the user can select a preexisting table to associate with the study, or create a new one by selecting the **New...** button (Figure 11).

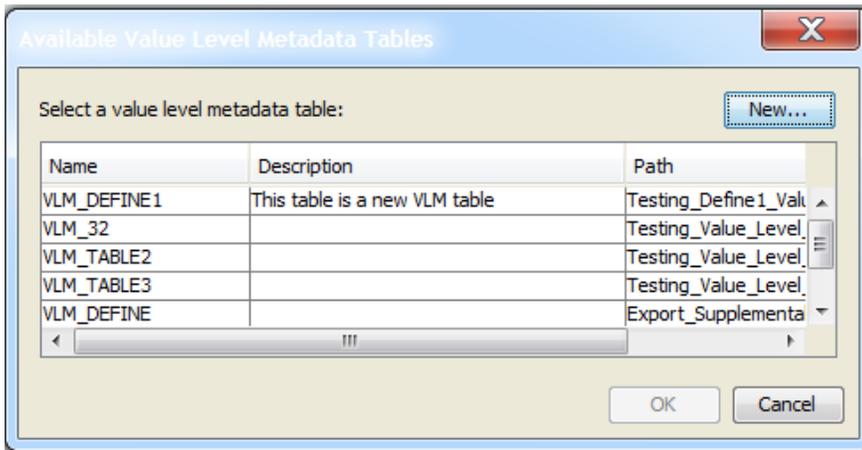


Figure 11. The Available Value Level Metadata Tables Window

If **New...** is selected, the **New Value Level Metadata Table** wizard is launched. The user must select a location within SAS Clinical Data Integration to store the metadata link to the physical table that will be created on the user's file system. Once chosen, the user is prompted for a table name, optional description and SAS Library used to create the permanent physical SAS data table (Figure 12).

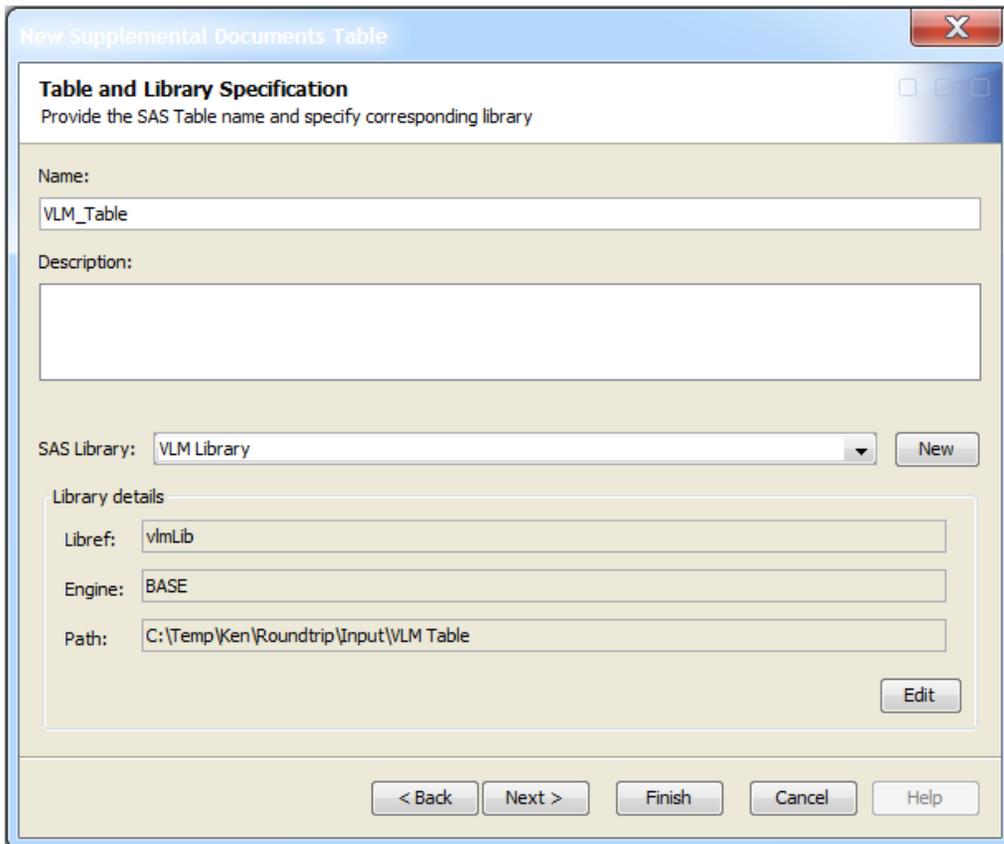


Figure 12. The Table and Library Specification Wizard Page

Upon completion of the wizard, the new empty table is created and pre-selected for association with the study. And, as was noted earlier, this same process is used to create and associate a Supplemental Documents table with a study, and after creating and adding both tables the Study Properties, Study tab would appear as in Figure 13.

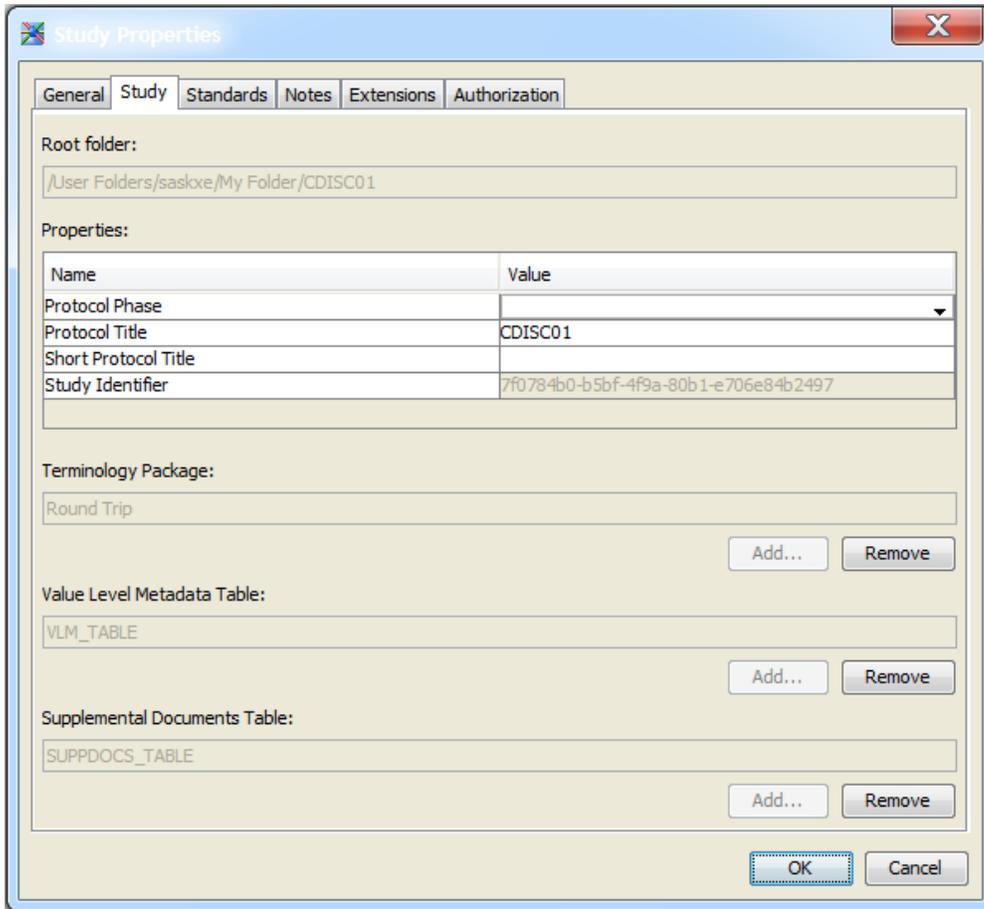


Figure 13. The Study Properties Page

A quick look at the study's folders shows that the tables have been created (Figure 14).

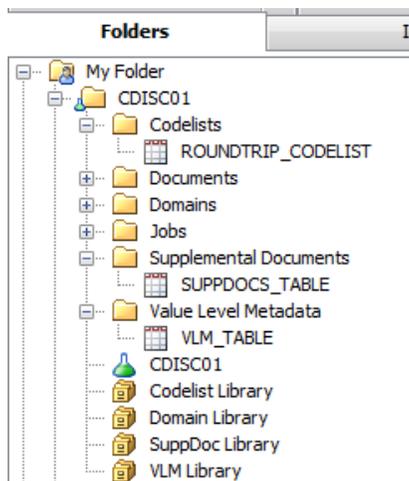


Figure 14. The Empty Tables have been created

The last step of the process involves importing data into the empty SAS data tables created in the previous steps. To do this, launch the **Value Level Metadata from Define...** or **Supplemental Documents from Define...** context menu by right-clicking on the table icon. The appropriate wizard will be launched and once again prompt you for the path to the define.xml file from which you wish to import the data. Upon completion of the wizard, the data will be imported into the appropriate table (Figure 15).

#	TABLE	COLUMN	VALUE	WHERECLAUSE	LABEL
1	DA	DAORRES	...	DATESTCD EQ "DISPAMT"	Dispensed Amount
2	DA	DAORRES	...	DATESTCD EQ "RETAMT"	Returned Amount
3	EG	EGORRES	...	EGTESTCD EQ "INTP"	Interpretation: Original Results
4	EG	EGORRES	...	EGTESTCD EQ "PRMEAN"	Summary (Mean) PR Duration (Orig U)
5	EG	EGORRES	...	EGTESTCD EQ "QRS DUR"	Summary (Mean) QRS Duration (Orig U)
6	EG	EGORRES	...	EGTESTCD EQ "QTMEAN"	Summary (Mean) QT Duration (Orig U)
7	EG	EGORRES	...	EGTESTCD EQ "VRMEAN"	Summary (Mean) Ventricular Rate (Orig U)
8	EG	EGSTRESC	...	EGTESTCD EQ "QTCB"	QTCB - Bazett's Correction Formula
9	EG	EGSTRESC	...	EGTESTCD EQ "QTCF"	QTCF - Fridericia's Correction Formula
10	IE	IEORRES	...	IETESTCD EQ "EXCL01"	Is pregnant, nursing, or planning to beco...
11	IE	IEORRES	...	IETESTCD EQ "EXCL02"	Is unable or unwilling to undergo multiple ...
12	IE	IEORRES	...	IETESTCD EQ "EXCL03"	Is known to have had a substance abuse...
13	IE	IEORRES	...	IETESTCD EQ "INCL01"	Is age 18 - 85
14	IE	IEORRES	...	IETESTCD EQ "INCL02"	Has Xyz disease of at least 10 weeks dur...
15	IE	IEORRES	...	IETESTCD EQ "INCL03"	Did not respond to a standard course of ...

Figure 15. Partial Listing of imported Value Level Metadata

### 5. Import Define-XML Data into the Domain Table Shells

SAS Clinical Data Integration provides a pre-defined Clinical transform that allows you to import CDISC Dataset-XML files. The transform relies on functionality provided by SAS Clinical Standards Toolkit to first create the domain tables (SAS data sets) on the file system and to then load data into them. The domain tables are "linked" to their counterparts within SAS Clinical Data Integration via the SAS Library that was initially associated with the metadata representation of the domain when it was imported from the define.xml file. This same define.xml file must be used as an input to the transformation when creating and loading the dataset-xml files as metadata definitions within it are used to validate the incoming data.

First, create a new **Document** within SAS Clinical Data Integration that references the define.xml by selecting the **New->Document...** context menu from within the study and pointing the document to the define.xml file that was used to import the domain metadata definitions (Figures 16, 17).

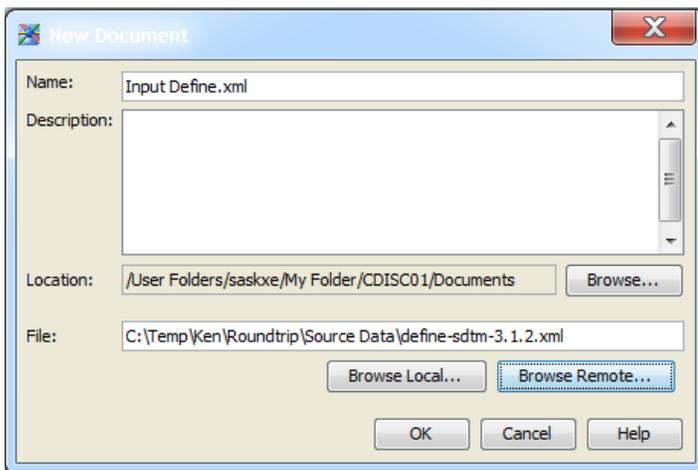


Figure 16. Document Definition for the define.xml File

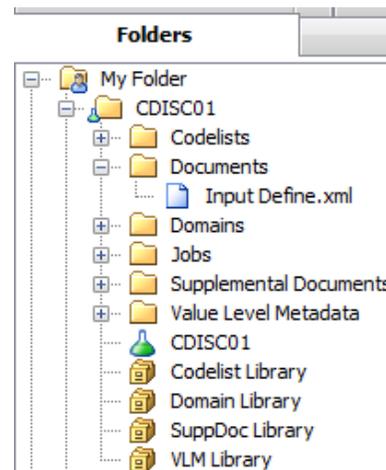


Figure 17. Document Folder

Next, create a new **Job** and drop the **CDISC-Dataset-XML Import** transform onto the Job palette. The transform takes one input; the Document (created above) that references the define.xml file that was initially used to import the domain metadata definitions (Figure 18).

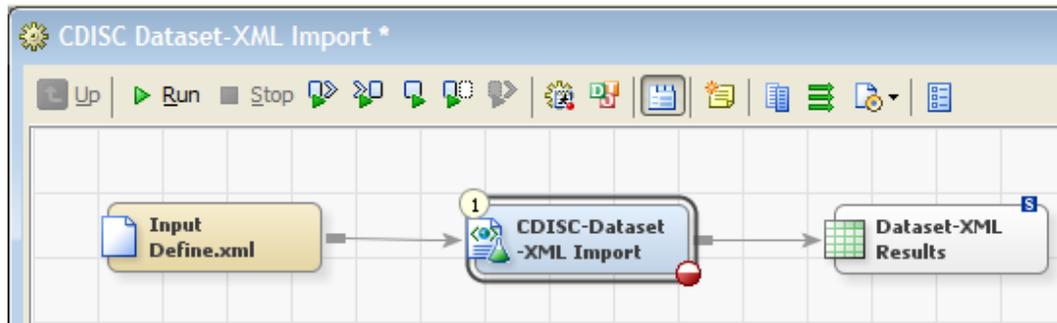


Figure 18. The CDISC-Dataset-XML Import Job Palette

Next, set the input source (where the dataset-xml files reside that you will be importing) and output destination (folder where the physical SAS data tables will be created). To do so, double-click on the transform to open its properties window and then select the **Generation** tab. Browse to, and select the folder where the dataset-xml files reside on your file system that you wish to import. Next select the output library (destination for the SAS data tables that will be created). Note that you should use the same library definition that was used to import the domain metadata definitions in **step 2**. By doing so, you will establish the “link” between the metadata definitions and the physical tables (Figure 19).

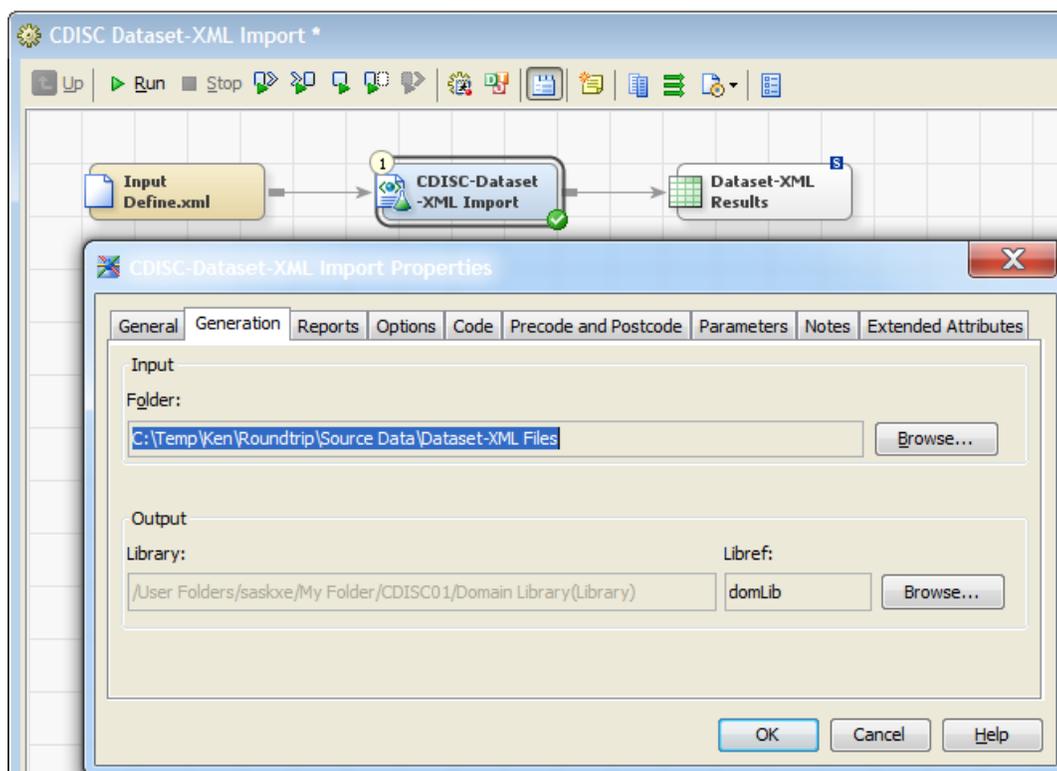


Figure 19. The Completed Transform's Generation Tab

Finally, select **OK** button on the transform’s properties window and then run the transform. When the Job completes, there will be a SAS data table created for every dataset-xml file that was found in the input folder (Figure 20).

Name	Date modified	Type	Size
ae	4/1/2015 4:00 PM	SAS Data Set	128 KB
cm	4/1/2015 4:00 PM	SAS Data Set	128 KB
da	4/1/2015 4:00 PM	SAS Data Set	128 KB
dm	4/1/2015 4:00 PM	SAS Data Set	128 KB
ds	4/1/2015 4:00 PM	SAS Data Set	128 KB
eg	4/1/2015 4:00 PM	SAS Data Set	128 KB
ex	4/1/2015 4:00 PM	SAS Data Set	128 KB
ie	4/1/2015 4:00 PM	SAS Data Set	128 KB
lb	4/1/2015 4:00 PM	SAS Data Set	128 KB
mh	4/1/2015 4:00 PM	SAS Data Set	128 KB
pe	4/1/2015 4:00 PM	SAS Data Set	128 KB

Figure 20. Partial Listing of Generated Domain Data Tables

To verify the link between the domain metadata definitions created in **step 2**, and the physical tables created via the **CDISC-Dataset-XML Import** transform, simply right click on any of the domain definitions inside of SAS Clinical Data Integration that have a corresponding physical table and select the Open context menu item. (Figures 21, 22).

#	STUDYID	DOMAIN	USUBJID	AESEQ	AESPID	AETERM
1	CDISC01	AE	CDISC01.100...	1 1		AGITATED ...
2	CDISC01	AE	CDISC01.100...	2 2		ANXIETY ...
3	CDISC01	AE	CDISC01.100...	3 3		DECREASED ...
4	CDISC01	AE	CDISC01.100...	1 1		DIARRHEA ...
5	CDISC01	AE	CDISC01.100...	2 2		HEMORRHOI...
6	CDISC01	AE	CDISC01.100...	3 3		HEADACHE ...
7	CDISC01	AE	CDISC01.100...	4 4		VOMIT ...
8	CDISC01	AE	CDISC01.100...	5 5		VOMIT ...
9	CDISC01	AE	CDISC01.200...	1 1		ANXIETY ...
10	CDISC01	AE	CDISC01.200...	2 5		LEFT KNEE PA...
11	CDISC01	AE	CDISC01.200...	3 3		CONSTIPATI...
12	CDISC01	AE	CDISC01.200...	4 4		TIREDDNESS ...
13	CDISC01	AE	CDISC01.200...	5 2		NAUSEA INTE...
14	CDISC01	AE	CDISC01.200...	1 3		LIGHTHEADE...
15	CDISC01	AE	CDISC01.200...	2 1		MUSCLE SPAS...
16	CDISC01	AE	CDISC01.200...	3 2		PALPITATION...

Figure 21. Partial Listing of AE Domain

```

<ItemGroupData ItemGroupOID="IG.AE" data:ItemGroupDataSeq="1"><ItemData ItemOID="IT.AE.STUDYID"
  Value="CDISC01"/>
  <ItemData ItemOID="IT.AE.DOMAIN" Value="AE"/>
  <ItemData ItemOID="IT.AE.USUBJID" Value="CDISC01.100008"/>
  <ItemData ItemOID="IT.AE.AESEQ" Value="1"/>
  <ItemData ItemOID="IT.AE.AESPID" Value="1"/>
  <ItemData ItemOID="IT.AE.AETERM" Value="AGITATED"/>
  <ItemData ItemOID="IT.AE.AEMODIFY" Value="AGITATION"/>
  <ItemData ItemOID="IT.AE.AEDECOD" Value="Agitation"/>
  <ItemData ItemOID="IT.AE.AEBODSYS" Value="Psychiatric disorders"/>
  <ItemData ItemOID="IT.AE.AESEV" Value="MILD"/>
  <ItemData ItemOID="IT.AE.AESER" Value="N"/>
  <ItemData ItemOID="IT.AE.AEACN" Value="DOSE NOT CHANGED"/>
  <ItemData ItemOID="IT.AE.AEREL" Value="POSSIBLY RELATED"/>
  <ItemData ItemOID="IT.AE.AESTDTC" Value="2003-05"/>
  <ItemData ItemOID="IT.AE.AESTDY" Value="3"/>
  <ItemData ItemOID="IT.AE.AEENRF" Value="AFTER"/>
</ItemGroupData>
    
```

Figure 22. The **ItemGroupData** Element for the AE Domain Corresponding to the First Row

## THE EXPORT PROCESS

SAS Clinical Data Integration provides a set of pre-defined Clinical Transforms that can be used to create define.xml and dataset-xml files, and a transform that can be used to validate a define.xml file against a standard schema. The transforms rely on functionality provided by SAS Clinical Standards Toolkit. We will first create a define.xml file that incorporates the domain, codelist, value level metadata and supplemental documents tables that we created in **steps 2-4** of the **Import Process**. Next we will validate the newly created define.xml file against the standard schema for CDISC-Define-XML Version 2.0. Finally, we will export the physical domain table data into corresponding dataset-xml files, thus completing the roundtrip.

### 1. Create Source and Destination Documents

The clinical transforms used in the following steps require Document objects that reference physical files on your file system. The output document (**Output Define.xml**) will serve as an output for the **CDISC-Define-XML Creation** transform. This document should reference the file that you want to use as the output destination for the define.xml file that will be created in **step 2** below. A separate input document (**Input Define.xml**) will serve as an input to the **CDISC-Dataset-XML Creation** transformation and should reference the define.xml file that was used during the import process. To create a SAS Clinical Data Integration Document, select the **New->Document...** context menu from within the study. (Figures 23, 24).

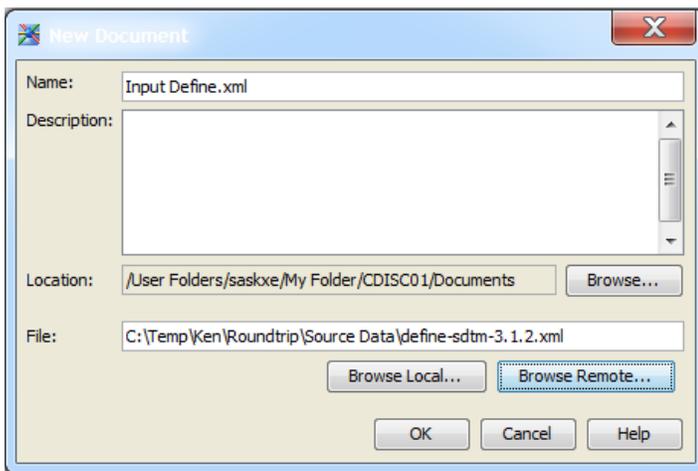


Figure 23. The Input define.xml file Document Definition

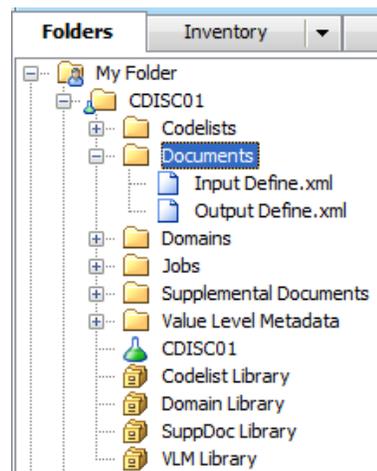


Figure 24. Document Folder

## 2. Export the Study Domains to a Define.xml File

To create a define.xml file you must first create a new **Job** and drop the **CDISC-Define-Creation** transform onto the Job's palette. The transform takes the **Output Define.xml** document created in **step 1** above (Figure 25).

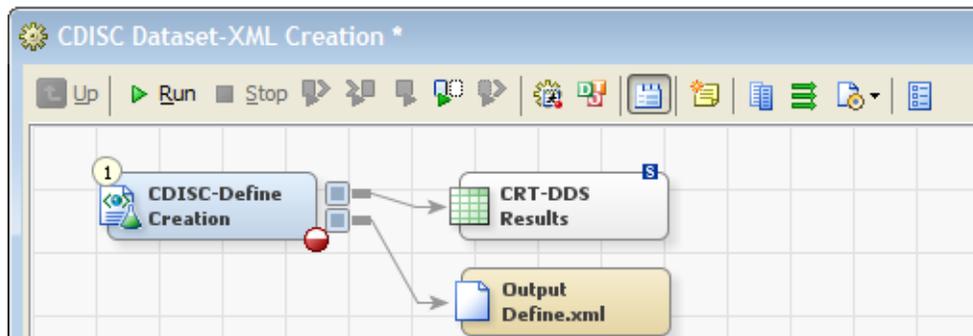


Figure 25. The CDISC-Define Creation Job Palette

Double-click on the transform to open its properties window and then select the **Tables** tab. Next, in the **Data Standard Type** pull-down menu, select the data standard that you initially associated with your study and the domain table shells that you created earlier. The **Study/Submission** pull-down menu will populate with all studies associated with the selected data standard. Select the study created in **step 1** and the **Tables** list will be populated with all domain definitions that are defined within the study for this data standard. Select the domains that should be included in the new define.xml that will be created (Figure 26).

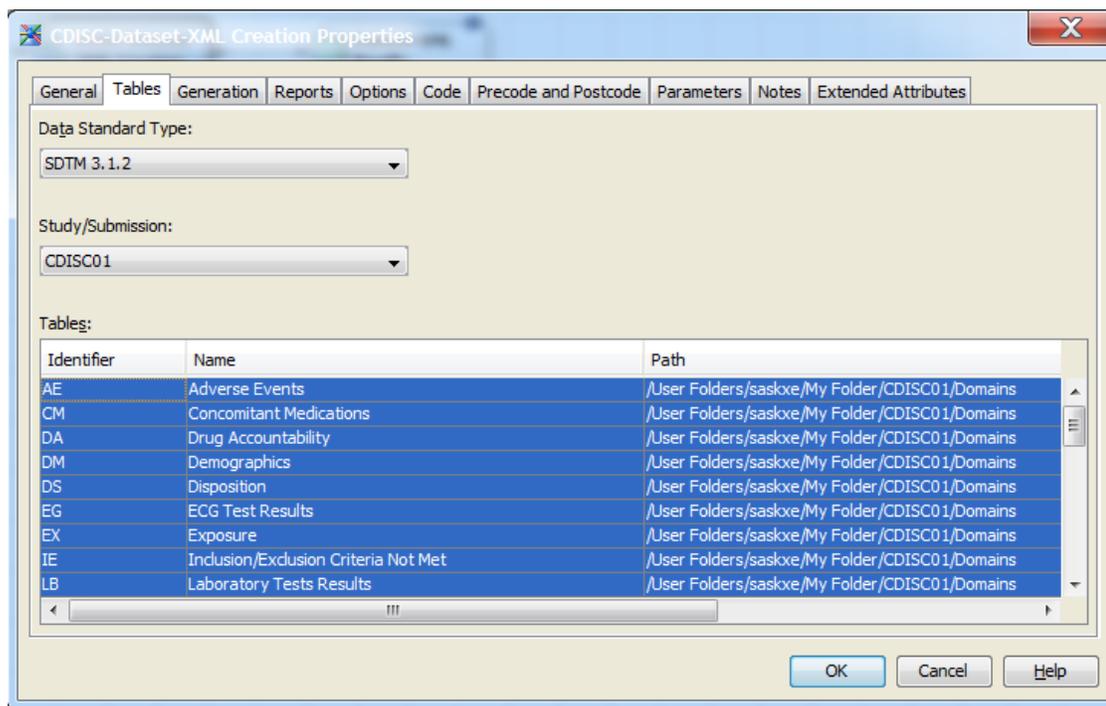


Figure 26. The Tables Tab with all Domains Selected for a Study

Next select the **Generation** tab for the transform. This page allows you to set various options for the new define.xml file you are about to create. SAS Clinical Data Integration supports creation of both define versions 1 and 2 (note that the default is version 1), as well as creation of a Define.pdf (version 1 only) and support for default or custom stylesheets. Select **2.0** in the **Define version** pull-down, and select the **Use default output stylesheet** radio button (Figure 27).

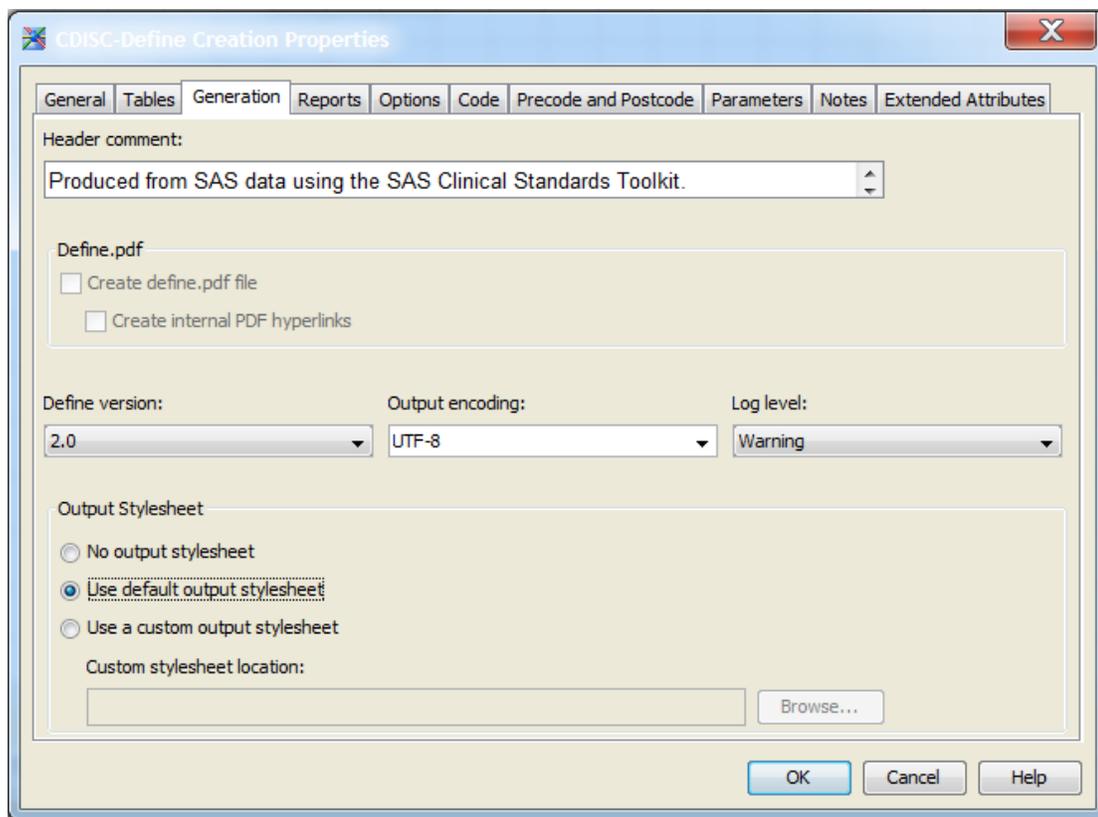


Figure 27. The Generation Tab

Finally, select the **OK** button on the transform’s properties window and then run the transform. When the Job completes, the new define.xml file will be created using the physical file referenced by the output document.

### 3. Validate the Define.xml File

SAS Clinical Data Integration can also be used to validate any CDISC compliant define.xml file against its standard schema. To validate a define.xml file, first create new **Job** and drop the **CDISC-Define-Validation** transform onto the Job’s palette. The transform takes only one external input, a document object that points to physical define.xml file to be validated. In our case, this would be the **Output Define.xml** document created in **step 1** above as it references the define file created in **step 2** above. (Figure 28).

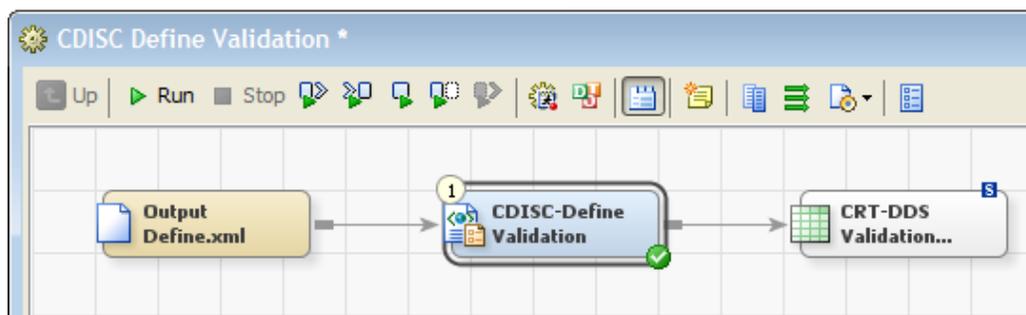


Figure 28. The CDISC-Define-Validation Job Palette

Double-click on the transform to open its properties window and then select the **Validation** tab. Note that the transformation can validate either define versions 1 or 2. In the **Define version** pull-down box select **2.0**, then close the properties window by selecting the **OK** button. Run the transform. When finished, open the CRT-DDS Validation Results (Work Table) and verify that no errors were generated (Figure 29).

#	checkid	resultid	resultseq	seqno	srcdata	message	resultseverity	resultflag
1		CST0108	1	1	CST_SETPR...	The properties were processed...	Info	0
2		CST0108	1	1	CST_SETPR...	The properties were processed...	Info	0
3		CST0200	1	1	CST_CREAT...	The SAS libref csttmplt was allo...	Info	0
4		CST0102	1	2	CST_CREAT...	work.sasreferences was create...	Info	0
5		CST0200	1	1	CST_INSERT...	SASReferences data set was s...	Info	0
6		CST0200	1	2	CSTUTIL_AL...	SASReferences data set was s...	Info	0
7		CST0108	1	1	CST_SETPR...	The properties were processed...	Info	0
8		CST0108	1	1	CST_SETPR...	The properties were processed...	Info	0
9		CST0200	1	1	CSTUTILXML...	PROCESS STANDARD: CDISC-...	Info	0
10		CST0200	1	2	CSTUTILXML...	PROCESS STANDARDVERSION:...	Info	0
11		CST0200	1	3	CSTUTILXML...	PROCESS DRIVER: None or uns...	Info	0
12		CST0200	1	4	CSTUTILXML...	PROCESS DATE: 2015-04-06T1...	Info	0
13		CST0200	1	5	CSTUTILXML...	PROCESS TYPE: XMLVALIDATE ...	Info	0
14		CST0200	1	6	CSTUTILXML...	PROCESS SASREFERENCES: C:...	Info	0
15		CST0200	1	7	CSTUTILXML...	PROCESS STUDYROOTPATH: <...	Info	0
16		CST0200	1	8	CSTUTILXML...	PROCESS GLOBALLIBRARY: c:\...	Info	0
17		CST0200	1	9	CSTUTILXML...	PROCESS CSTVERSION: 1.7 ...	Info	0
18		CST0200	1	1	JAVA CHECK...	No Java issues	Info	0
19		CST0200	1	1	CSTUTILPRO...	XML Log file C:\Users\saskxe\A...	Info	0
20		CST0100	1	2	CSTUTILXML...	No errors detected in the XML f...	Info	0
21		CST0102	1	1	CSTUTIL_SA...	_cdiwrk.VALIDATION_RESULTS...	Info	0
22		CST0102	1	1	CSTUTIL_SA...	_cdiwrk.VALIDATION_RESULTS...	Info	0

Figure 29. The Validation Results Table

#### 4. Export the Domains to Dataset-XML files

To export dataset-xml files you must first create a new **Job** and drop the **CDISC-Dataset-XML Creation** transform onto the Job palette. The transform takes one input; a Document such as the one created in **step 1** that references a define.xml. Note that the metadata from the referenced define.xml file *must* match that of the domain table(s) that are to be written out as dataset-xml files. For the purposes of this paper, we could use either the input *or* output define.xml documents since we did not alter any of the domain metadata. For completeness, we will use the **Output Define.xml** document containing the current metadata definitions. (Figure 30).

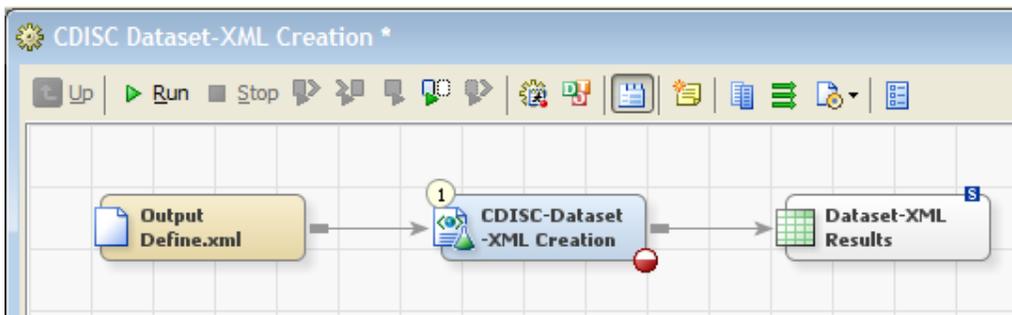


Figure 30. The CDISC-Dataset-XML Creation Job Palette

Double-click on the transform to open its properties window and then select the **Tables** tab. In the **Data Standard Type** pull-down menu, select the data standard that you initially associated with your study and the domain table shells that you created earlier. The **Study/Submission** pull-down menu will populate with all studies associated with the selected data standard. Select the study whose table data you wish to output as dataset-xml files and the **Tables** list will populate. Finally, select the tables for which dataset-xml files will be created. (Figure 31).

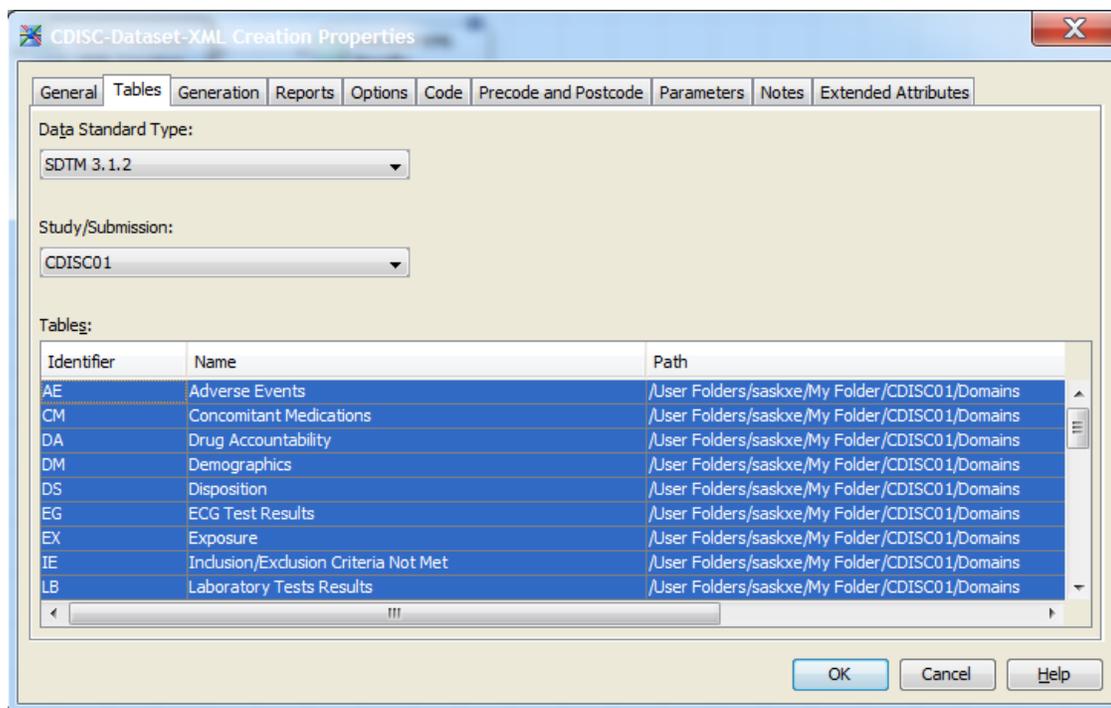


Figure 31. The Tables Tab with all Domains Selected for a Study

Next, select the **Generation** tab in the transform’s properties window. Select a physical **Output folder** location for writing the dataset-xml files and then select the **OK** button to close the properties window. Run the transform. When the transform has completed, the output folder will be populated with individual dataset-xml files for each table selected (Figure 32). Finally, a quick visual comparison of the output dataset-xml file’s contents with the content of the corresponding original dataset-xml file that was imported will show that the correct data was written to the dataset-xml file (Figures 33, 24)

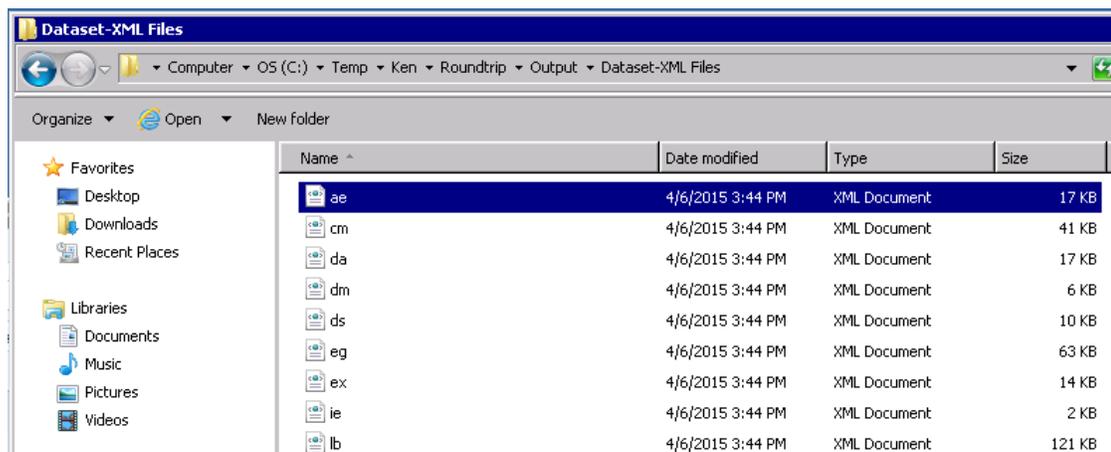


Figure 32: Partial listing of the dataset-xml Files Created

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Produced from SAS data using SAS Clinical Standards Toolkit -->
<ODM
  xmlns="http://www.cdisc.org/ns/odm/v1.3"
  xmlns:data="http://www.cdisc.org/ns/Dataset-XML/v1.0"
  ODMVersion="1.3.2"
  FileType="Snapshot"
  FileOID="STUDY1.AE"
  PriorFileOID="DEFINE"
  CreationDateTime="2014-07-15T17:08:26"
  data:DatasetXMLVersion="1.0.0">
  <ClinicalData StudyOID="STUDY1" MetadataVersionOID="MDV.CDISC01.SDTMIG.3.1.2.SDTM.1.2">
    <ItemGroupData ItemGroupOID="IG.AE" data:ItemGroupDataSeq="1">
      <ItemData ItemOID="IT.AE.STUDYID" Value="CDISC01"/>
      <ItemData ItemOID="IT.AE.DOMAIN" Value="AE"/>
      <ItemData ItemOID="IT.AE.USUBJID" Value="CDISC01.100008"/>
      <ItemData ItemOID="IT.AE.AESEQ" Value="1"/>
      <ItemData ItemOID="IT.AE.AESPID" Value="1"/>
      <ItemData ItemOID="IT.AE.AETERM" Value="AGITATED"/>
      <ItemData ItemOID="IT.AE.AEMODIFY" Value="AGITATION"/>
      <ItemData ItemOID="IT.AE.AEDECOD" Value="Agitation"/>
      <ItemData ItemOID="IT.AE.AEBODSYS" Value="Psychiatric disorders"/>
      <ItemData ItemOID="IT.AE.AESEV" Value="MILD"/>
      <ItemData ItemOID="IT.AE.AESER" Value="N"/>
      <ItemData ItemOID="IT.AE.AEACN" Value="DOSE NOT CHANGED"/>
      <ItemData ItemOID="IT.AE.AEREL" Value="POSSIBLY RELATED"/>
      <ItemData ItemOID="IT.AE.AESTDTC" Value="2003-05"/>
      <ItemData ItemOID="IT.AE.AESTDY" Value="3"/>
      <ItemData ItemOID="IT.AE.AEENRF" Value="AFTER"/>
    </ItemGroupData>
  </ClinicalData>
</ODM>

```

Figure 33: First Row of Input Dataset-xml File Data for the AE Domain

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Produced from SAS data using the SAS Clinical Standards Toolkit. -->
<ODM
  xmlns="http://www.cdisc.org/ns/odm/v1.3"
  xmlns:data="http://www.cdisc.org/ns/Dataset-XML/v1.0"
  ODMVersion="1.3.2"
  FileType="Snapshot"
  FileOID="STUDY1.AE"
  PriorFileOID="DEFINE"
  CreationDateTime="2015-04-06T15:44:13"
  data:DatasetXMLVersion="1.0.0">
  <ClinicalData StudyOID="STUDY1" MetadataVersionOID="MDV.CDISC01">
    <ItemGroupData ItemGroupOID="IG.AE" data:ItemGroupDataSeq="1">
      <ItemData ItemOID="IT.AE.STUDYID" Value="CDISC01"/>
      <ItemData ItemOID="IT.AE.DOMAIN" Value="AE"/>
      <ItemData ItemOID="IT.AE.USUBJID" Value="CDISC01.100008"/>
      <ItemData ItemOID="IT.AE.AESEQ" Value="1"/>
      <ItemData ItemOID="IT.AE.AESPID" Value="1"/>
      <ItemData ItemOID="IT.AE.AETERM" Value="AGITATED"/>
      <ItemData ItemOID="IT.AE.AEMODIFY" Value="AGITATION"/>
      <ItemData ItemOID="IT.AE.AEDECOD" Value="Agitation"/>
      <ItemData ItemOID="IT.AE.AEBODSYS" Value="Psychiatric disorders"/>
      <ItemData ItemOID="IT.AE.AESEV" Value="MILD"/>
      <ItemData ItemOID="IT.AE.AESER" Value="N"/>
      <ItemData ItemOID="IT.AE.AEACN" Value="DOSE NOT CHANGED"/>
      <ItemData ItemOID="IT.AE.AEREL" Value="POSSIBLY RELATED"/>
      <ItemData ItemOID="IT.AE.AESTDTC" Value="2003-05"/>
      <ItemData ItemOID="IT.AE.AESTDY" Value="3"/>
      <ItemData ItemOID="IT.AE.AEENRF" Value="AFTER"/>
    </ItemGroupData>
  </ClinicalData>
</ODM>

```

Figure 33: First Row of Output Dataset-xml File Data for the AE Domain

## CONCLUSION

SAS Clinical Data Integration Version 2.6, coupled with the underlying functionality of SAS Clinical Standards Toolkit Version 1.7 provides a powerful set of tools for managing both the import and export of clinical trials metadata and data. These tools, in combination with others provided for metadata management within SAS Clinical Data Integration, facilitate the easy exchange of information between a CRO and a study sponsor, and can be used to generate key pieces of a submission to the FDA.

## REFERENCES

1. CDISC Define-XML Specification, Version 2.0, March 5, 2013  
(<http://www.cdisc.org/define-xml>)
2. CDISC Dataset-xml, Version 1.0  
(<http://www.cdisc.org/dataset-xml>)
3. SAS Clinical Data Integration 2.6: User's Guide  
(<http://support.sas.com/documentation>)
4. SAS Clinical Standards Toolkit 1.7: User's Guide  
(<http://support.sas.com/documentation>)

## CONTACT INFORMATION

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

Ken Ellis  
SAS Institute Inc.  
SAS Campus Drive  
Cary, NC 27519, USA

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.