Paper FC01

Cost Effective Ways to Generate DEFINE.PDF & DEFINE.XML

Sy Truong, Meta-Xceed, Inc, Milpitas, CA

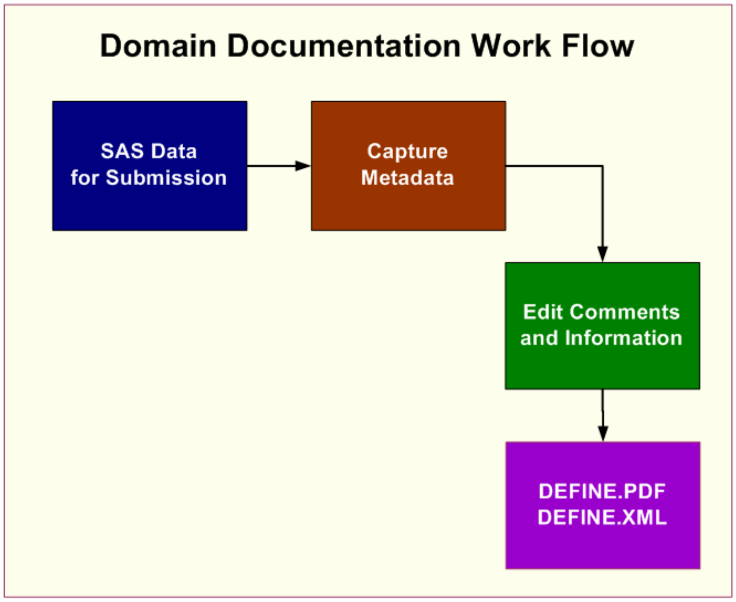
ABSTRACT

You can have greater understanding and management of your data if it is well documented with data definition documentation in the format of DEFINE.PDF and DEFINE.XML. As the number of datasets and variables increase, this can be very resource intensive. The time consuming documentation task is compounded by the fact that there are constant changes to the data so the documentation has to keep up with the changes in order for it to be useful and accurate. This paper will suggest methods and tools that would enable you to document your data definition document without purchasing a complex expensive system.

INTRODUCTION

When you plan for a road trip, you need a map. This is analogous to understanding the data that is going to be part of an electronic submission. The reviewer requires a road map in order to understand what all the variables are and how they are derived. It is within the interest of all team members involved to have the most accurate and concise documentation pertaining to the data. This can help your team work internally while also speeding up the review process which can really make or break an electronic submission to the FDA. Some organizations perform this task at the end of the process but they really lose out on the benefits which the document provides for internal use. It is therefore recommended that you initiate this process early and therefore gain the benefit of having a road map of your data.

The process that is involved in managing and creating the data definition documentation is as follows:



The process is an iterative one since the SAS datasets are updated. The constant need to update the documentation is therefore one of the challenges which this paper will address.

LEVELS OF METADATA

There are several steps towards documenting the data definition. Most of what is being done is documenting metadata which is information about the data that is to be included. There are several layers to the metadata. These include:

1. **General Information** – This pertains to information that affects the entire set of datasets that are to be included. It could be things such as the name of the study, the company name, or location of the data.
2. **Data Table** – This information is at the SAS dataset level. This includes things such as the dataset name and label.

3. **Variable** – This information pertains to attributes of the variables within a dataset. This includes such information as variable name, label and lengths.

The order in which the metadata is captured should follow the same order as the layers that are described.

STEP 1: Capture the general information pertaining to the data. The following lists the types of information which you need to be concerned about.

Metadata	Description
Company Name	This is the name of the organization that is submitting the data to the FDA.
Product Name	The name of the drug that is being submitted.
Protocol	The name of the study on which the analysis is being performed which includes this set of data.
Layout	The company name, product name, and protocol are all going to be displayed on the final documentation. The layout information will describe if it will be in the footnote or title and how it is aligned.

This high level metadata will be used in headers and footers on the final documentation.

STEP 2: Some of the dataset level information can be captured through PROC CONTENTS but others need to be defined when you are documenting your data definition. Some of the information includes:

Metadata	Description
Data Library	Library name defines what physical path on which server and where the data is located. This can also be in the form of a SAS LIBNAME.
Key Fields	Keys usually correlate to the sort order of the data. These variables are usually used to merge the datasets together.
Format Library	This is where the SAS format catalog is stored.
Dataset Name	The name of the SAS dataset that is being captured.
Number of Variables	A count of the number of variables for each dataset.
Number of Records	Number of observations or rows within each dataset.
Dataset Comment	A descriptive text describing the dataset. This can contain the dataset label and other descriptive text explaining the data.

SAS Tools such as PROC CONTENTS can contribute to most of these items. However, comments and key fields can be edited which may differ from what is stored in the dataset.

STEP 3: The last step and level to the domain documentation is the variable level. This includes the following:

Metadata	Description
Variable Name	The name of the SAS variable.
Type	The variable type which includes values such as Character or Numeric.
Length	The variable length.
Label	The descriptive label of the variable.

Format	SAS formats used. If it is a user defined format, it would need to be decoded.
Origins	The document where the variable came from. Sample values include: Source or Derived.
Role	This defines what type of role the variable is being used for. Example values include: Key, Ad Hoc, Primary Safety, Secondary Efficacy
Comment	This is a descriptive text explaining the meaning of the variable or how it was derived.

Similar to the data set level metadata, some of the variable level attributes can be captured through PROC CONTENTS. However, fields such as origins, role and comments need to be edited by someone who understands the meaning of the data.

MANUAL CAPTURING AND EDITING

Some of the metadata information can be captured by PROC CONTENTS as previously mentioned. However, other information has to be entered manually. It is therefore recommended that you have PROC CONTENTS create the initial set of the metadata. The rest can be manually entered. The following example shows you how you can capture this programmatically.

CODE EXAMPLE 1:

```
*** Capture the initial metadata ***;
proc contents data = sashelp.shoes
   out=work.shoes;
run;

*** Expoert this information to excel ***;
proc export data=work.shoes
   outfile="c:\temp\shoes.xls"
   dbms=excel
   replace;
run;
```

In this example, the metadata of the dataset “shoes” is exported into an Excel spreadsheet named “shoes.xls”. You can therefore edit the information in Excel. You can also cut and paste this into MSWord since the text editing of MSWord may be more flexible.

There are advantages and disadvantages to using this method.

Advantages

1. Does not require any additional software so it is the most cost effective way.

2. Familiarity with existing tools such as Excel and Word.

Disadvantages

1. When the data is updated, updates to the documentation are difficult since the export wipes out the entered data.

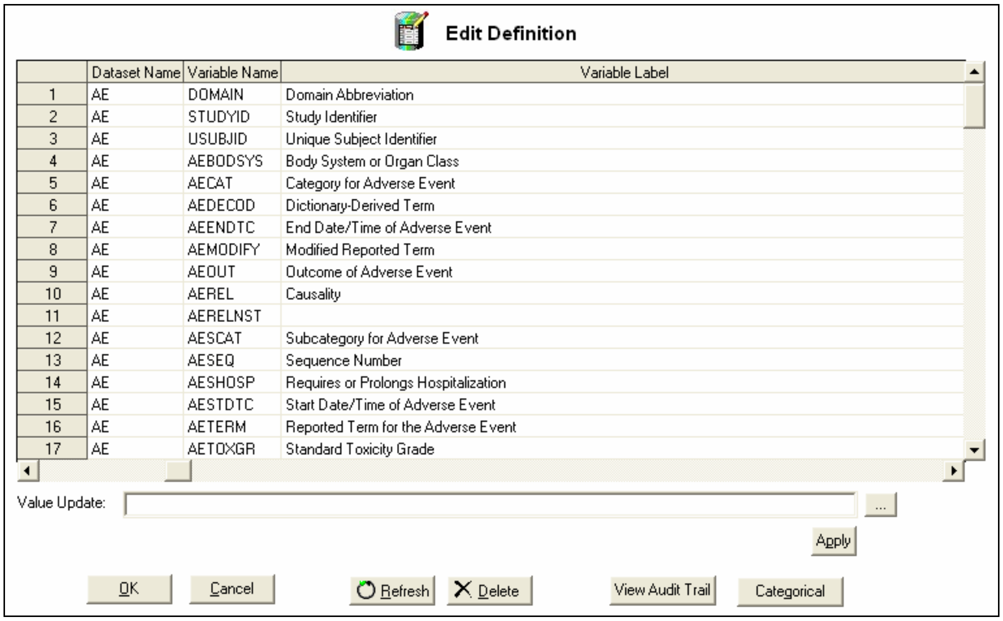
2. There is a lack of guidance as to what values are to be entered. This is left as free text that the user can enter. This is prone to data entry errors.

3. The PROC CONTENTS produces extra information which has to be deleted and new fields have to be added beyond what PROC CONTENTS provides.

4. There is no audit trail documenting what has been entered, by whom and at what time.

AUTOMATING CAPTURING AND EDITING

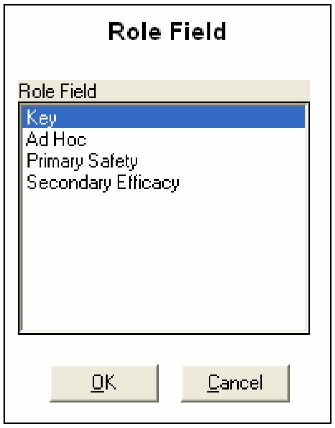
Tools such as PROC CONTENTS and Excel do have capabilities to customize and automate the documentation to a degree. They are not however intended specifically for creating data definition documentation. These tools therefore have limitations. A tool that was developed entirely in SAS specifically for generating this type of documentation is Defindoc™**.** This tool contains both a graphical user interface and a macro interface to fit the user's requirements. The tool addresses all the disadvantages of the manual methods. It uses a similar PROC CONTENTS type of mechanism of capturing the initial metadata. However, it only retains the specific information that is pertinent to the data definition documentation.



	Dataset Name	Variable Name	Variable Label
1	AE	DOMAIN	Domain Abbreviation
2	AE	STUDYID	Study Identifier
3	AE	USUBJID	Unique Subject Identifier
4	AE	AEBODSYS	Body System or Organ Class
5	AE	AECAT	Category for Adverse Event
6	AE	AEDECOD	Dictionary-Derived Term
7	AE	AEENDTC	End Date/Time of Adverse Event
8	AE	AEMODIFY	Modified Reported Term
9	AE	AEOUT	Outcome of Adverse Event
10	AE	AEREL	Causality
11	AE	AERELNST	
12	AE	AESCAT	Subcategory for Adverse Event
13	AE	AESEQ	Sequence Number
14	AE	AESHOSP	Requires or Prolongs Hospitalization
15	AE	AESTDTC	Start Date/Time of Adverse Event
16	AE	AETERM	Reported Term for the Adverse Event
17	AE	AETOXGR	Standard Toxicity Grade

Value Update:

Definedoc automatically captures attributes pertaining to information captured by PROC CONTENTS. For other values, it presents possible values that users can select for more consistency.

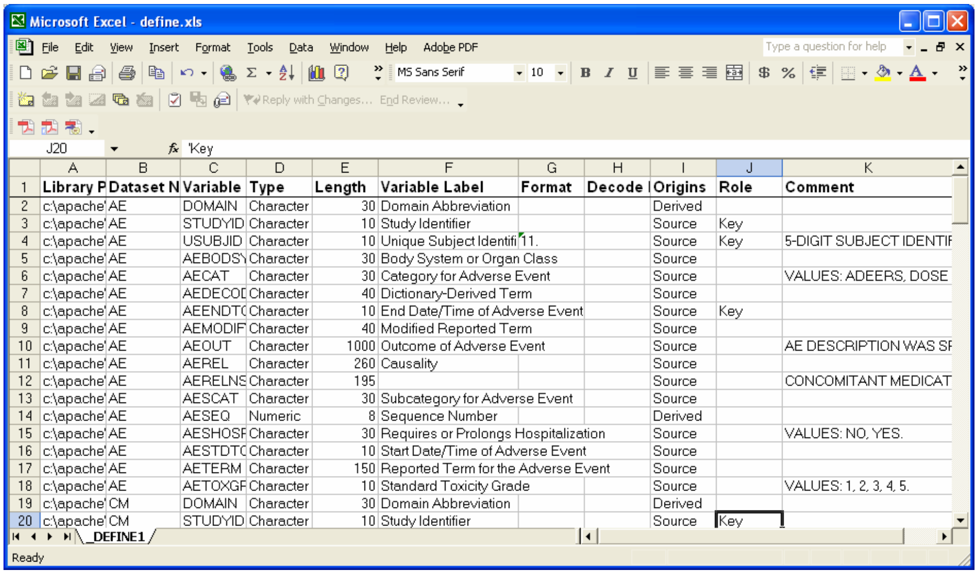


Role Field

Key
Ad Hoc
Primary Safety
Secondary Efficacy

The tool also keeps track of all edits in an audit trail capturing who has updated what column so that if anything goes wrong, it can easily be traced back and fixed. One of the main advantages is that if any of the variable attributes are updated, this can be "refreshed" with a click of a button. It will not affect those fields that the user has entered, but rather, it updates other attributes such as variable names and labels.

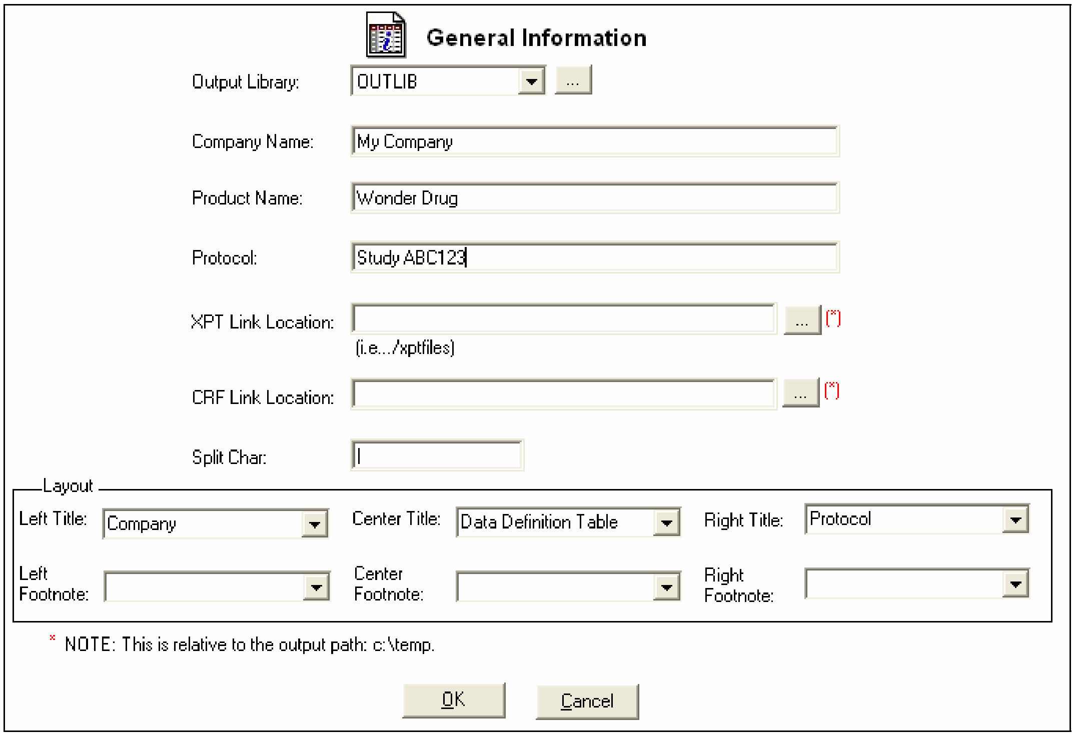
Definedoc has the flexibility of exporting the pertinent information to an excel spreadsheet so that those users who prefer to edit their values within Excel can do so.


Microsoft Excel - define.xls
File Edit View Insert Format Tools Data Window Help Adobe PDF
Type a question for help
MS Sans Serif 10
Reply with Changes... End Review...
_DEFINE1
Ready

J20 'Key

	A	B	C	D	E	F	G	H	I	J	K
1	Library P	Dataset N	Variable	Type	Length	Variable Label	Format	Decode	Origins	Role	Comment
2	c:\apache'	AE	DOMAIN	Character	30	Domain Abbreviation			Derived		
3	c:\apache'	AE	STUDYID	Character	10	Study Identifier			Source	Key	
4	c:\apache'	AE	USUBJID	Character	10	Unique Subject Identifi	11.		Source	Key	5-DIGIT SUBJECT IDENTIF
5	c:\apache'	AE	AEBODSY	Character	30	Body System or Organ Class			Source		
6	c:\apache'	AE	AECAT	Character	30	Category for Adverse Event			Source		VALUES: ADEERS, DOSE
7	c:\apache'	AE	AEDECOD	Character	40	Dictionary-Derived Term			Source		
8	c:\apache'	AE	AEENDTC	Character	10	End Date/Time of Adverse Event			Source	Key	
9	c:\apache'	AE	AEMODIF	Character	40	Modified Reported Term			Source		
10	c:\apache'	AE	AEOUT	Character	1000	Outcome of Adverse Event			Source		AE DESCRIPTION WAS SF
11	c:\apache'	AE	AEREL	Character	260	Causality			Source		
12	c:\apache'	AE	AERELNS	Character	195				Source		CONCOMITANT MEDICAT
13	c:\apache'	AE	AESCAT	Character	30	Subcategory for Adverse Event			Source		
14	c:\apache'	AE	AESEQ	Numeric	8	Sequence Number			Derived		
15	c:\apache'	AE	AESHOSF	Character	30	Requires or Prolongs Hospitalization			Source		VALUES: NO, YES.
16	c:\apache'	AE	AESTDTC	Character	10	Start Date/Time of Adverse Event			Source		
17	c:\apache'	AE	AETERM	Character	150	Reported Term for the Adverse Event			Source		
18	c:\apache'	AE	AETOXGF	Character	10	Standard Toxicity Grade			Source		VALUES: 1, 2, 3, 4, 5.
19	c:\apache'	CM	DOMAIN	Character	30	Domain Abbreviation			Derived		
20	c:\apache'	CM	STUDYID	Character	10	Study Identifier			Source	Key	

This provides the best of both worlds. It captures just the values that you want and exports this to Excel for those who prefer this interface. Once you are finished with editing the information in Excel, the same spreadsheet can be re-imported so that the information is handled centrally. Besides the dataset and variable level metadata information, Definedoc also helps automate the capture of the high level general information.


General Information

Output Library: OUTLIB

Company Name: My Company

Product Name: Wonder Drug

Protocol: Study ABC123

XPT Link Location: (*)
(i.e.../xptfiles)

CRF Link Location: (*)

Split Char: |

Layout

Left Title: Company
Center Title: Data Definition Table
Right Title: Protocol

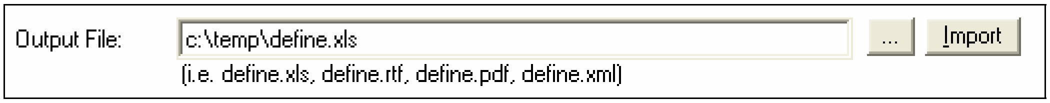
Left Footnote:
Center Footnote:
Right Footnote:

* NOTE: This is relative to the output path: c:\temp.

This handles both the editing of the information and layout of the final report.

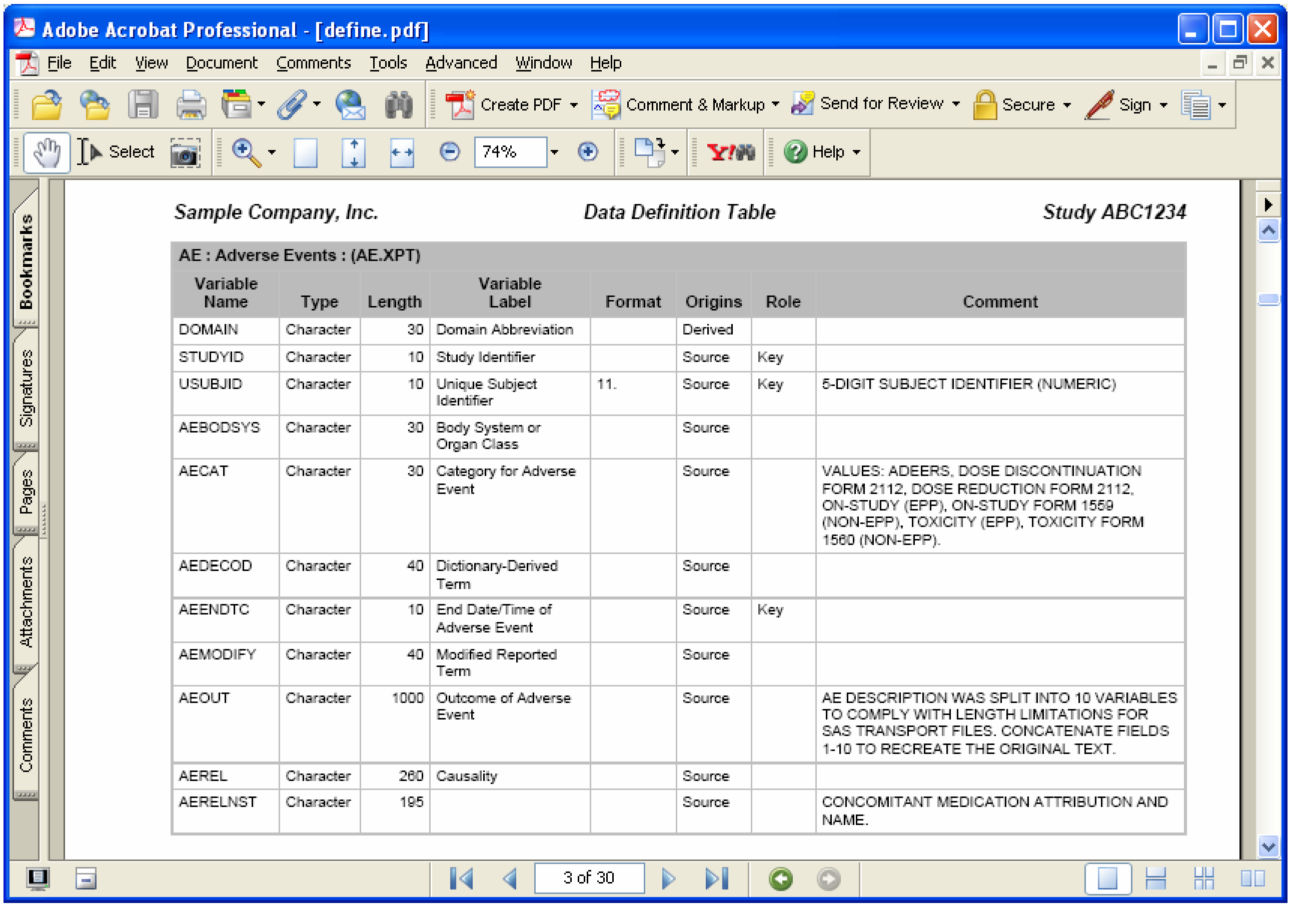
GENERATING DOCUMENTATION

The last step in the process is to generate the documentation in either PDF or XML format. The challenge is that in order to make the documentation useful, it requires hyperlinks to link the information together. The manual method does allow you to format the information in Word and this can be converted into PDF format. Even though Word and Excel can generate XML, it does not have the proper schema so there is no manual way of generating the XML version of the report. Definedoc has the flexibility of generating the report in Excel, RTF, PDF and XML.



Output File: c:\temp\define.xls
(i.e. define.xls, define.rtf, define.pdf, define.xml)

It utilizes ODS within SAS to produce the output in all these formats. In addition to the XML file, Definedoc also produces the accompanying cascading style sheet to format the XML so that you can view this within a browser in a similar format as in a web browser. An example PDF output would look like:



Sample Company, Inc. Data Definition Table Study ABC1234

AE : Adverse Events : (AE.XPT)

Variable Name	Type	Length	Variable Label	Format	Origins	Role	Comment
DOMAIN	Character	30	Domain Abbreviation		Derived		
STUDYID	Character	10	Study Identifier		Source	Key	
USUBJID	Character	10	Unique Subject Identifier	11.	Source	Key	5-DIGIT SUBJECT IDENTIFIER (NUMERIC)
AEBODSYS	Character	30	Body System or Organ Class		Source		
AECAT	Character	30	Category for Adverse Event		Source		VALUES: ADEERS, DOSE DISCONTINUATION FORM 2112, DOSE REDUCTION FORM 2112, ON-STUDY (EPP), ON-STUDY FORM 1559 (NON-EPP), TOXICITY (EPP), TOXICITY FORM 1560 (NON-EPP).
AEDECOD	Character	40	Dictionary-Derived Term		Source		
AEENDTC	Character	10	End Date/Time of Adverse Event		Source	Key	
AEMODIFY	Character	40	Modified Reported Term		Source		
AEOUT	Character	1000	Outcome of Adverse Event		Source		AE DESCRIPTION WAS SPLIT INTO 10 VARIABLES TO COMPLY WITH LENGTH LIMITATIONS FOR SAS TRANSPORT FILES. CONCATENATE FIELDS 1-10 TO RECREATE THE ORIGINAL TEXT.
AEREL	Character	260	Causality		Source		
AERELNST	Character	195			Source		CONCOMITANT MEDICATION ATTRIBUTION AND NAME.

The documentation can be generated through a graphical user interface. This makes it easy for a novice to learn the process. However, experienced users prefer to operate in batch mode for production work. This allows the work to be processed at greater efficiencies. The interface for this batch processing is via a macro call.

CODE EXAMPLE 2:

```
*** Example generation of Excel file ***;
%definepdf(data = dataware,
   source = C:\path\to\source\data,
   fmtlib = library.formats,
   output = define.xls,
   keys = ptid);

*** Generate the OUTPUT file from existing definition ***;
%definepdf(outlib = mylib,
   output = c:\mydir\define.pdf);
```

CONCLUSION

It is a common mistake to underestimate the amount of resources required to generate data definition documentation. This is due to the fact that it is an iterative process due to the dynamic nature of changing data. The result can be very resource intensive, especially if it is done manually. By performing cost analysis on the amount of time lost by manually updating the information accompanied by the resulting error prone documentation as compared to automated tools, it is more cost effective to invest in automated tools such as Definedoc. This provides flexibility and features that are not otherwise possible. The savings can be significant when it comes to time and to ensuring that there is accuracy and integrity in the documentation.

REFERENCES

ABOUT THE AUTHOR

Sy Truong is President of MXI (Meta-Xceed, Inc.) They may be contacted at:

Sy Truong
1751 McCarthy Blvd.
Milpitas, CA 95035
(408) 955-9333
sy.truong@meta-x.com