

## Creating Case Report Tabulations (CRTs) for an NDA Electronic Submission to the FDA

Anita Rocha, STATPROBE, Inc., Tukwila, WA  
Paul Hamilton, STATPROBE, Inc., Tukwila, WA

### ABSTRACT

The Food and Drug Administration (FDA) now strongly encourages all new drug applications (NDAs) be submitted electronically. Electronic submissions could help FDA application reviewers scan documents efficiently and check analyses by manipulating the very datasets and code used to generate them. The potential saving in reviewer time and cost is enormous while improving the quality of oversight. In January 1999, the FDA released the *Guidance for Industry: Providing Regulatory Submissions in Electronic Format – NDAs*. As described, one important part of the application package is the case report tabulations (CRTs), now serving as the instrument for submitting datasets. CRTs are made up of two parts: first, datasets in SAS® transport file format and second, the accompanying documentation for the datasets. Herein, we briefly review the content and conversion of datasets to SAS transport file format, and then elaborate on the code that makes easy work of the accompanying dataset documentation (in the form of data definition tables) using the SAS Output Delivery System (ODS). The intended audience is SAS programmers with an intermediate knowledge of the BASE product used under any operating system and who are involved in the biotechnology industries.

### INTRODUCTION

Approval to sell a new drug or medical device in the United States is gained by formal application process through the Food and Drug Administration (FDA). In the hope of speeding this lengthy process and controlling costs, the FDA now strongly encourages all New Drug Applications (NDAs) be submitted electronically. Electronic submissions cut down on wading through paper reports, and use of linking minimizes searching. Moreover, FDA application reviewers can check analyses and summaries by manipulating the very datasets and code used to generate them, and in turn improve the quality of oversight. Randy Lewin, director for electronic review in the FDA's Office of Review Management, expects most submissions to be in an electronic format by the year 2002<sup>1</sup>.

Ideally, the electronic format both acts as and supports effective tools for the FDA application reviewers. Unfortunately, reviewers aren't equipped with uniform software. To broaden access, submission packages should contain easily downloadable files for use on reviewers' own platforms with their own software, available datasets and code, and a user-friendly guide to the datasets' contents and variable location. In January 1999, the FDA released the *Guidance for Industry: Providing Regulatory Submissions in Electronic Format – NDAs*<sup>2</sup>. As described, one important part of the application package is the case report tabulations (CRTs), now serving as the means for submitting datasets.

CRTs are made up of datasets and the accompanying documentation for the datasets. This paper details the development of CRTs for electronic submission to the FDA, including

- A brief review of the content of CRT datasets and their conversion to SAS transport file format with a link to a useful SAS Institute web site,
- SAS code to generate data definition tables using the Output Delivery System (ODS) in SAS version 7 or higher,
- Conversion of the data definition tables to Adobe Portable Document Format (PDF), and
- Examples of CRTs from an NDA developed in part at

STATPROBE for our client Salix Pharmaceuticals, Inc. The intended audience is SAS programmers with an intermediate knowledge of the BASE product used under any operating system and who are involved in the biotechnology industries.

### CONTENT OF DATASETS

While conducting a clinical trial, clinicians collect all kinds of patient data. Section K of the FDA's *Guidance for NDAs* describes criteria for CRTs and, in particular, outlines the contents of datasets submitted as part of the CRTs. In general, the following data (grouped on what were formerly known as CRF domains) should be provided as individual datasets.

- |  |                        |
|--|------------------------|
| ➤ Demographics   | ➤ Microbiology data    |
| ➤ Inclusion criteria   | ➤ Adverse Events       |
| ➤ Exclusion criteria   | ➤ Lab – chemistry      |
| ➤ Concomitant medication                                     | ➤ Lab – hematology     |
| ➤ Medical history  | ➤ Lab – urinalysis     |
| ➤ Drug exposure  | ➤ ECG                  |
| ➤ Disposition  | ➤ Vital signs          |
| ➤ Efficacy results   | ➤ Physical examination |
| ➤ Human pharmacology and bioavailability/bioequivalence data |                        |

In practice, however, applicants may discuss what data to include as part of the CRTs with the FDA review division prior to the electronic submission. Note that some FDA reviewers' software require that files are first loaded into random access memory (RAM), the *Guidance* explains, so individual files cannot exceed 25 MB. To accommodate, programmers might need to break-up large datasets. In addition to raw and derived variables, each dataset should include core demographics information such as age, sex, race, ethnicity, and site location. This helps reviewers track and analyze basic information quickly. The *Guidance* provides further instructions for dataset contents. For the purposes of this paper, we will assume that appropriate datasets including raw, derived, and core demographic variables are prepared and exist as permanent SAS datasets.

For example, our client Salix Pharmaceuticals, Inc. with the intent of submitting an NDA, collected results from a randomized trial comparing Rifaximin to a standard treatment of Ciprofloxacin to manage traveler's diarrhea. A subset of the permanent SAS datasets, composed of raw, derived and core demographics variables, created as part of this submission package were named as follows.

- ADEVENT.SD2
- CONCOM.SD2
- DEMOGRA.SD2
- . . .
- VITAL.SD2

To accommodate general platforms, note that the featured dataset file names above do not exceed eight characters. A similar restriction is put on the lengths of variable names and labels. According to the accompanying text *Guidance for Industry: Providing Regulatory Submissions in Electronic Format – General Considerations*, variable names should not exceed eight characters and variable labels, although highly recommended, should not exceed 32 characters in length. At this point, it would be prudent to check variable names and label lengths and if necessary alter their lengths accordingly. A short macro to flag and print variable names that exceed eight characters or labels that exceed 32 follows.

```
%Macro ckname(dirname);
  /* Finds variable names > 8 chars and labels
  > 32 chars */
  /* DIRNAME = full path of directory of target
  datasets */
  Libname contents "&dirname";
  Proc contents data=contents._all_
  out=contents noprint;
  Run;
  Data check;
  Set contents(keep=memname name label);
  Length ckname kclabel $ 10.;
  If (length(name) le 8) and
  (length(label) le 32) then delete;
  Else do;
  If (length(name) gt 8) then ckname =
  'NAME > 8';
  If (length(label) gt 32) then kclabel =
  'LABEL > 32';
  End;
  Run;
  Proc print data=check;
  Run;
%Mend ckname;
```

Example:

```
%Ckname(c:\salix\1234\crt\datasets\ssd);
```

**CONVERSION TO SAS TRANSPORT FILE FORMAT**

After checking lengths and, if necessary, altering names and labels, it is straightforward to convert the datasets to SAS version 5 transport files. The FDA web site gives a link to a useful SAS Institute site that provides a comprehensive set of conversion macros<sup>3</sup>. Using a handy macro provided by the SAS Institute, we converted the datasets above to transport files, all with extension .XPT as specified by the FDA guidelines.

- ◆ ADEVENT.XPT
- ◆ CONCOM.XPT
- ◆ DEMOG.XPT
- ...
- ◆ VITAL.XPT

**CREATE DATA DEFINITION TABLES**

To accompany the datasets now in SAS transport file format, the FDA requests documentation in the form of two (or possible three) Portable Document Format (PDF) files. PDF files are created using the software package Adobe Acrobat<sup>4</sup> developed to help users share electronic information. Using this package, it is simple to convert HTML output generated by SAS ODS to PDF files.

**CRTs**

Datasets: SAS Transport File Format	Documentation: PDF
<ul style="list-style-type: none"> <li>▪ Data partitioned into datasets based on domains</li> </ul>	<ul style="list-style-type: none"> <li>▪ Patient Profiles (Optional)</li> <li>▪ Annotated CRFs</li> <li>▪ Data Definition Tables: DEFINE.PDF               <ol style="list-style-type: none"> <li>1. List of datasets</li> <li>2. List of variables for each dataset</li> </ol> </li> </ul>

Under the 'Documentation' portion of the diagram shown, one PDF file contains the patient profiles, displaying a collection of individual patient data over time. The FDA may or may not request this file, depending on the agreement between the applicant and the FDA review division. One required PDF file contains a complete set of annotated case report forms (CRFs), the forms originally used to collect the clinical information. Neither the development of the patient profiles nor the scanning and conversion of the annotated CRFs will be discussed here. Rather, we'll focus on creation of the third and essential PDF file

containing data definition tables called DEFINE.PDF, a file name specified by the FDA.

DEFINE.PDF is made up of two kinds of data definition tables. The first table is a list of all the datasets, their descriptions and locations. The second section is a set of tables, one for each dataset, containing variable names, labels, types, codes, and comments. These structures somewhat parallel the useful SAS dictionary tables, SASHELP.VTABLE and SASHELP.VCOLUMN. We will look at each part of the data definition tables in turn after a brief discussion of views.

**VIEWS**

The SASHELP data library contains permanent PROC SQL views allowing easy access to DICTIONARY tables. Here reside mountains of information about variables, datasets, libraries, options and more. In particular, SASHELP.VTABLE provides library names (LIBNAME), dataset names (MEMNAME) and dataset labels (MEMLABEL). Variable names (NAME), variable labels (LABEL), and variable format names (FORMAT) are found in SASHELP.VCOLUMN. Fortunately, these sources make up 90% of the information needed to create the data definition tables incorporated into CRTs.

**DATA DEFINITION TABLE PART 1 – DATASETS**

The first table names and defines all the submitted datasets and identifies subdirectory locations. An example is provided below.

*Datasets For Study 1234*

Dataset	Description of dataset	Location
Demo	Demographics	Crt/datasets/1234/demo.xpt
...	...	...

Assuming the SAS datasets reside in a single library (in this case a library called DSHOME), we used the following program to create a table using ODS. Notice the use of the view SASHELP.VTABLE.

```
%Let file1 = u:\_FILE TRANSFER\SUGI
25\Paper_ARPH\Output\Ddt\datasets.htm;
%Let study=Salix Protocol R/C-TD/01.97;
%Let locate=Crt\datasets\RCTD0197\;
```

```
Data tables1;
  Set sashelp.vtable(keep = libname memname
  memlabel);
  Where (libname eq 'DSHOME');
  Locate = "&locate" || trim(lowercase(memname))
  || '.xpt';
  Run;
```

```
Ods html file = "&file1";
Proc report data=work.tables1 ls=155 ps=70
split='|' headline center missing nowd;
  Column memname memlabel locate;
  Define memname / order format=$8. width=13
  spacing=2 left 'Dataset';
  Define memlabel / format=$30. width=35
  spacing=2 left
  'Description of Dataset';
  Define locate / format=$35. width=40
  spacing=2 left 'Location';
  Title "&study - Datasets";
  Run;
Ods html close;
```

Applying this code to our client's datasets resulted in an HTML table, a portion of which is displayed below. Note that in the "Location" column, the transport file format datasets reside in a particular subdirectory or folder. This location is relative to the submission package, not the storage system employed by the

applicants or consulting programmers. In other words, the location column should answer the question, "Where can a reviewer locate [dataset] in the electronic submission package?"

Salix Protocol R/C-TD/01.97 - Datasets		
Dataset	Description of Dataset	Location
<a href="#">ADEVENT</a>	Adverse Events	Crt\datasets\RCTD0197\adevent.xpt
<a href="#">CONCOM</a>	Concomitant Medications	Crt\datasets\RCTD0197\concom.xpt
<a href="#">PHYSICAL</a>	Physical Exam	Crt\datasets\RCTD0197\physical.xpt
<a href="#">VITAL</a>	Vital Signs	Crt\datasets\RCTD0197\vital.xpt

Note that the dataset column is a different color and underlined indicating a live link to the table that describes a particular dataset. Although the final product to the FDA is in PDF, the functionality will be virtually identical.

### DATA DEFINITION TABLE PART 2 – VARIABLES

The second set of tables is made up of specific dataset descriptors. Each table represents a single dataset describing its variables, labels, types, codes and additional comments. For example, the table below partially describes the contents of dataset DEMO.

#### Study 1234 – DEMO: Demographics Dataset Variables

Variable	Label	Type	Codes	Comments
PATID	Patient ID number	Char		DEMO page 3
DUR	Duration of Treatment	Num		STOPDT – STARTDT
TRT	Treatment group	Num	0=placebo 5=5mg/day	@
...				

@Use footnotes to display longer comments.

Some of the information in the above table is easily garnered from the SASHELP library views (the variable name, label and type). Other information may not be fully available (codes column), or worse, completely unavailable in either the datasets or views (comments column). First, let us find a way to access information for the codes column.

### EXTRACTING CODES THROUGH FORMATS

In order to derive codes, we will need access to variable formats. It helps greatly if the formats are centrally stored in a single library catalog. Otherwise, manual data entry may be required (not shown here). The program below accesses formats, creates a character field for codes, and matches variables to the appropriate code. Notice the use of the view SASHELP.VCOLUMN.

```
Data tables2(keep=libname name memname memlabel
type label format fmtlist);
  Merge
    Tables1(keep=libname memname memlabel)
    Sashelp.vcolumn(where=(libname eq
'DSHOME'));
  By libname memname;
  Length fmtlist $ 9;
  Fmtlist = put(format, $9.);
Run;
```

```
/* This is the library containing formats */
Proc format library = library
Cntlout = fmts;
Run;
```

```
Data fmts(keep=fmllisto code);
  Set fmts;
  Length fmllisto $ 9 code $ 45;
```

```
Fmllisto = trim(left(fmtname)) || '.';
/* Convert format to character field */
If (verify(start, end) eq 0) then
  Code = trim(left(start)) || '=' ||
trim(left(label));
Else
  Code = trim(left(start)) || '-' ||
trim(left(end)) ||
  '=' || trim(left(label));
Run;

/* Compute the Cartesian product of */
/* formats and variable names */
Proc sql;
Create table cart as
  Select * from tables2 a, fmts b
  Where a.fmtlist = b.fmllisto;
Quit;
```

### ADDING A COMMENT FIELD

Populating the comment column may be among the most time consuming parts of creating data definition tables, particularly since the content varies depending on the associated variable. For instance, non-derived data originating from a CRF is sometimes called "raw data". The comment associated with a raw data variable typically describes its originating field location on the CRF, possibly including the CRF name or number, and the page number. For a derived variable, a comment may indicate which variables were used in its computation or perhaps the applied formula. Longer explanations can be displayed as footnotes. Unfortunately, the hapless programmer cannot avoid the painstaking process of entering this information. Luckily, there are multiple ways of jumping this hurdle. The programmer can enter comments in an EXCEL spreadsheet or into a database like ACCESS. We used a PROC FSP screen entry system, part of SAS/FSP®, which for brevity will not be described here.

By combining the comments, the code field created above in dataset CART, and dataset TABLES2, the complete information for the second set of data definition tables is available. Similar to the first data definition table, the PROC REPORT with an ODS envelope creates the desired HTML output.

### CONVERT TABLES TO PDF

Now we're ready to convert SAS output into the PDF file DEFINE.PDF required by the FDA. Using the Adobe Acrobat package, there are two ways to transform original material into PDF:

- PDFWriter is a virtual print driver installed during a typical Acrobat installation. One prints to PDFWriter much like any selected printer, but the result is a PDF file. This is an easy, one-step process, but some Acrobat users report problems with the quality of the resultant output.
- One can also print the original source material straight to a disk file, sending the Postscript code to a file instead of a Postscript printer. Then Adobe Distiller can convert the Postscript file to a PDF file. This may be accomplished interactively (either the usual File... Open or drag-and-drop techniques), or through the use of *watched folders*, where Distiller runs in the background and watches for .PS files to appear in a particular directory and then automatically converts them.

Source	Technique	Result
Traditional SAS output (V6)	PDFWriter or Distiller	PDF File (ordinary text)
SAS ODS Output	PDFWriter	PDF File (ODS look)
SAS ODS Output	Word 2000 ► PDFWriter	PDF File (Word table look)

Traditional SAS output can be converted to PDF by printing the text in a DMS OUTPUT window to a disk file (being careful to use a Postscript print driver), and then invoking Adobe Distiller on the resulting Postscript code. This results in a plain looking but faithful PDF version of the original output. When using Version 6 of the SAS System, this technique works well.

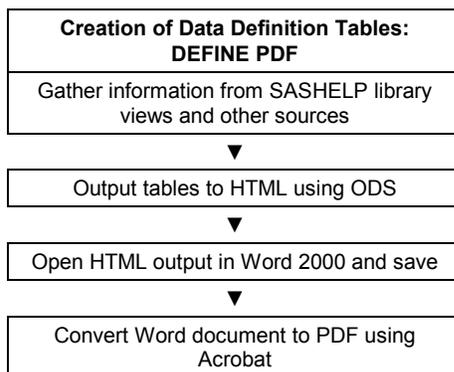
When using Version 7 or higher, one can print ODS output directly to PDFWriter from the Results Browser, which produces a satisfactory PDF file. However, using current versions of all relevant software packages, some of the HTML look and feel is lost, and the table title and column headings are available only on the first page of each output. Another technique is to open the HTML output from Version 7 directly in Microsoft Word 2000, and the HTML table converts automatically into a Word table. When this is converted to a PDF file (via PDFWriter or Distiller), one advantage is realized: Word repeats the table column headings at the top of each page of output, which facilitates reading multiple-page outputs. For the purposes of data definition tables in this paper, we used the latter technique because we thought the output appealing for CRTs.

### LINKING

One of the advantages of PDF parallels HTML functionality: links embedded in the document can transfer the reader to another document location with a single mouse click. The inexpensive, simple, and tedious way to accomplish this in native Acrobat is through the Acrobat application. The FDA recommends links be established between each entry in the data definition table that lists out all the datasets and the corresponding table defining the individual dataset structure. Further, variable names can be linked to the corresponding data collection fields on the annotated CRFs in PDF. This can be accomplished with no additional software through mouse and menu commands in Acrobat. Working this way is a strictly manual process, however, and should only be undertaken when the output being linked is in final, perfect form. The result is concise, accessible information, that together with the SAS transport file format datasets make up CRTs that conform to FDA guidelines.

### CONCLUSION

Converting SAS datasets to SAS transport file format was easy. On the other hand, the technique we used to arrive at PDF files of data definition tables required a number of intermediate steps. The diagram below outlines our effort.



In the new version of SAS, it is our understanding that ODS with added functionality will allow direct output to PDF files. Clearly the fewest number of steps, involving the fewest number of software packages, is preferable. Still a question is whether the format we found most appealing for CRTs, the standard paging with repeatable column headings, will result. We look forward to seeing upcoming ODS capabilities.

### REFERENCES

<sup>1</sup> De Bruce, O, "Data Standard Eases FDA Drug Approval", *Federal Computer Week*, May 24, 1999.

<sup>2</sup> [www.fda.gov/cder/guidance/index.htm](http://www.fda.gov/cder/guidance/index.htm). See Electronic Submissions.

<sup>3</sup> [www.sas.com/software/industry/pht/fda/macro.html](http://www.sas.com/software/industry/pht/fda/macro.html)

<sup>4</sup> [www.adobe.com](http://www.adobe.com)

### ACKNOWLEDGEMENTS

We would like to thank Salix Pharmaceuticals, Inc. for their kind permission to use their data and, in particular, Alvaro Carvajal for useful input into the development of the CRTs.

### TRADEMARK CITATION

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 registered trademarks or trademarks of their respective companies.

### CONTACT INFORMATION

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

Anita Rocha or Paul Hamilton  
 STATPROBE, Inc.  
 16000 Christensen Rd., Bldg. 2, Suite 150  
 Tukwila, WA 98188  
 Work Phone: (206) 242-0300  
 Fax: (206) 242-0312  
 Email: [anita.rocha@se.statprobe.com](mailto:anita.rocha@se.statprobe.com),  
[paul.hamilton@se.statprobe.com](mailto:paul.hamilton@se.statprobe.com)  
 Web: [www.statprobe.com](http://www.statprobe.com)