

Creating Hyperlinked PDF Graphical Patient Profiles with PROC REPORT

William Conover, Advanced Clinical, Bannockburn, IL

ABSTRACT

Using SAS® 9.1.3, SAS/GRAPH® and the ODS PDF destination, the methods described here create graphical patient profiles with hyperlinks to data listings.

INTRODUCTION

Patient profiles are customized data listings for clinical trials that give an overview of individual subject data. Quite often they are used by data management and safety review teams to monitor the data integrity and safety of subjects while the clinical trial is in progress. Additionally, medical writing teams use the profiles to assist in writing narratives. Depending on the requirements of the teams, design varies greatly. Some profiles display large amounts of numerical and text information, while others display mostly graphics. The profiles presented in this paper display a mix of graphics and text information with the added option of hyperlinks to detailed data listings. All output is in PDF format.

All SAS code was developed in PC SAS 9.1.3 using PROC REPORT and PROC PRINT as the reporting tools, utilizing the ODS PDF destination. A simple implementation of the %ANNOMAC system of macros available from SAS is also implemented to enhance the display. Graphic files for plots were produced separately, and are imported into the profiles using the ODS PDF TEXT statement.

The process for creating the patient profiles and listings is divided into three parts.

1. The first part focuses on the patient profile layout and functionality, PROC REPORT syntax, and the data manipulation prior to using PROC REPORT.
2. The second part focuses on special features like colored text, hyperlinks to data listings and adding graphic files/plots.
3. The third part presents a quick method for creating multiple detailed data listings with hyperlinks back to the patient profile document.

Our example incorporates data from 4 modules, demographics, adverse events, medication use and vital signs. Adding more modules is relatively straight forward after the initial concepts are understood. Note that a customized ODS STYLE template was used for the profiles, and is not discussed.

PART 1: LAYOUT, DATA MANIPULATION, SPECIAL FEATURES

LAYOUT

The first page per subject shows demographic, adverse event and medication use data.

1. **Blue text** is hyperlinked to the respective data listing, directly to the subject of interest. A hyperlink also exists on the data listing linking back to the same subject in the patient profile document.
2. The grey area represents the timeline of the study with visit numbers displayed along the top of each section (i.e. V1, V2, V3, etc.).
3. A color coded bar above the visit numbers is a visual aid to key portions of the study timeline.
4. Dashes “-“ in the grey area indicate the approximate start and stop of an event, with a description of the event directly left of the grey area.
5. A “<” or “>” before or after dashes indicates that an event started or stopped outside of the study timeline, respectively.

6. The last grey section on the bottom gives visit information details indicating the day of the study a visit occurred (D1, D4, D7, etc.) as well as the period (Period 1, Period 2, etc.).

An example of the first page per subject is shown below.

Protocol: XYZ123
Study XYZ-123

Advanced Clinical

Main Patient Profiles

Demographics [DM]

Patient ID: 700001

Serious AE? YES

Initials: XYZ, Age: 31, Gender: F
Race: OTHER, Race(if other): ASIAN / AMERICAN INDIAN, Ethnicity: NOT HISPANIC OR LATINO
Randomization Date: 20100201

Adverse Events [AE]

(SAE-Red Text)

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13/FU
HEADACHE													
DIZZINESS													
TOOTH EXTRACTION PAIN													
NAUSEA													

Prior and Concomitant Medications [CM]

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13/FU
TYLENOL W/ CODEINE													
SINEMET													
CLARITIN													

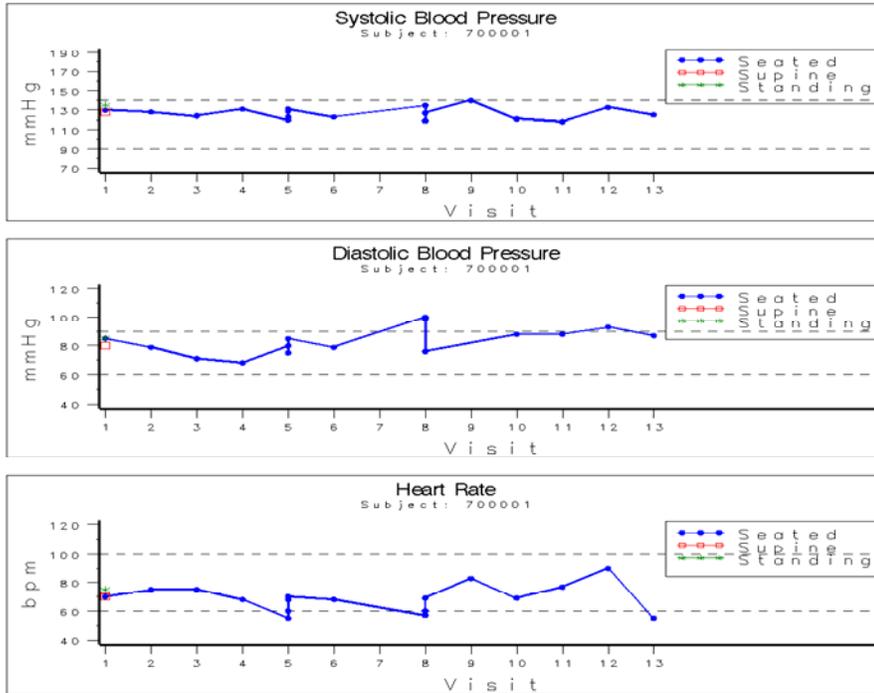
	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13/FU
D-10													
Period 1													
Period 2													
Period 3													
Period 4													

The second page per subject displays vital sign information graphically.

Protocol: XYZ123
Study XYZ-123

Advanced Clinical

Main Patient Profiles
Vital Signs [VS]



DATA MANIPULATION

The first page per subject uses only two display columns in the PROC REPORT statement, referred to as ColVar1 and ColVar2. A simple PROC PRINT of ColVar1 and ColVar2 from the above example is given below.

```

Obs    ColVar1                ColVar2

 1
 2
 3
 4
 5
 6
 7
 8
 9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

```

		Demographics [DM]												
		Patient ID: 700001												
													Serious AE? YES	
		Initials: XYZ, Age: 31, Gender: F												
		Race: OTHER, Race(if other): ASIAN / AMERICAN INDIAN, Ethnicity: NOT HISPANIC OR LATINO												
		Randomization Date: 20100201												
		Adverse Events [AE] (SAE=Red Text)												
		V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13/FU
10	HEADACHE													
11	DIZZINESS													
12	TOOTH EXTRACTION PAIN													
13	NAUSEA													
14		Prior and Concomitant Medications [CM]												
15		V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13/FU
17	TYLENOL W/ CODEINE													
18	SINEMET	<----->												
19	CLARITIN	<----->												
20														
21		V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13/FU
22														
23		D-10	D1	D4	D7	D14	D18D21	D28	D35	D42	D49	D56		
24														
25														

This output by itself is very descriptive, but lacks the additional features available in PROC REPORT. Below is a portion of the PROC REPORT code with similar output results.

```

proc report data=table1 split='~' style(column)=[LINKCOLOR = _undef_ ];
column subject hyplnk section ColVar1 ColVar2 ;
define subject / order noprint;
define hyplnk / order noprint;
define section / order noprint;
define ColVar1 / ' ' display style(column)=[cellwidth=1in font_face='Courier' font_size=.1 just=right];
define ColVar2 / ' ' display style(column)=[asis=on cellwidth=6.2in font_face='Courier' font_size=1.25];

```

Notice that the display width of ColVar2 is set at 6.2 inches. With the above “font face” and “font size” settings this translates, by trial and error, to approximately 88 spaces of text before wrapping occurs (in SAS 9.1.3 PDF destination). In order for the report to make sense visually, all items in ColVar2 must be scaled to fit within the 88 spaces. If the clinical trial study timeline is 88 days, then each day is 1 space. However, if the clinical trial is 65 days, for instance, each day is 88/65 spaces. For a 65 day study, the study day that an event occurred can be scaled to fit by simple math.

```
NewEventDay_val=int(EventDay*88/65);
```

To create the dashes associated with an event, the start day and end day of the event must be known. With that information the duration of the event can be calculated and scaled to fit as above. Sample SAS code below:

```
NewDay_start=int(StartDay*88/65);
NewDay_end=int(EndDay*88/65);
Duration=NewDay_end-NewDay_start+1
```

Using the REPEAT function, ColVar2 is assigned the correct number of dashes and spaces per event.

```
ColVar2=repeat(' ', NewDay_start) || repeat('-', Duration);
```

Note that the corresponding event name is stored in ColVar1.

For Adverse Events, additional text can be added after the dashes indicating severity.

```
ColVar2=ColVar2 || (' | aesev | ');
```

Demographic information is easily placed into the ColVar2 text string with basic programming methods, as well as adding the section titles. Visit information is essentially “hard coded” into ColVar2 using similar text string programming methods.

PART 2: SPECIAL FEATURES

PROC REPORT is used in conjunction with the ODS PDF destination to add additional reporting features of color text, grey background, hyperlinks and the color coded bar for visit information. The primary reason PROC REPORT is chosen for this task is the versatility of the COMPUTE BLOCK.

COLOR TEXT

Typical “traffic lighting” techniques are used for **color coding** and **bolding** text. For example, to indicate an adverse event is serious by displaying the text as **red**, a separate variable is created for formatting and assigned a value based on the seriousness of the event. In a previous data step, the variable Var1fmtVal is assigned to 'Yes' if the event is serious. Then, the compute block is used to assign text color (i.e. 'FOREGROUND=') via the STYLE element based on the value of Var1fmtVal. Notice in the below CALL DEFINE statement the syntax '**_col_-1**' and '**_col_-2**', this notifies PROC REPORT to add STYLE elements to the two variables preceding Var1fmtVal in the COULMN statement, ColVar2 and ColVar1 respectively.

```
proc report data=table1 split='~' style(column)=[LINKCOLOR = _undef_ ];
column subject hypLnk section ColVar1 ColVar2 Var1fmtVal;
define subject / order noprint;
define hypLnk / order noprint;
define section / order noprint;
define Var1fmtVal / noprint;
define ColVar1 / '' display style(column)=[cellwidth=1in font_face='Courier' font_size=.1 just=right];
define ColVar2 / '' display style(column)=[asis=on cellwidth=6.2in font_face='Courier' font_size=1.25 ];
```

```
compute Var1fmtVal ;
if Var1fmtVal='YES' then do; *← Yes for SAE;
CALL DEFINE(_col_-1, 'STYLE', 'STYLE=[FOREGROUND=red background=lightgrey]');
CALL DEFINE(_col_-2, 'STYLE', 'STYLE=[FOREGROUND=red]');
Else do;
CALL DEFINE(_col_-1, 'STYLE', 'STYLE=[FOREGROUND=black background=lightgrey]');
CALL DEFINE(_col_-2, 'STYLE', 'STYLE=[FOREGROUND=black]');
end;
```

Using another IF STATEMENT in the above COMPUTE BLOCK, this same variable can indicate other text formatting options like bold text, font size, etc. as seen in the demographics section of our profiles for 'Patient ID'. Adding the STYLE element 'background=lightgrey' to ColVar2 creates the grey timeline portion of the profile.

HYPERLINKS

Each section of the profile (Demo, AE, CM, etc.) is hyperlinked to a corresponding external data listing (PDF), directly to the Subject Number. The hyperlink is contained in the section title, and is colored **blue**. In ODS PDF this is the default color, but also contains a blue box surrounding the text. To remove the blue box around the text the STYLE element of 'LINKCOLOR=_undef_' is added to the PROC REPORT options, as seen in the above example code.

Linking to a specific point in a data listing from the profile page requires the use of 'anchors' in the hyperlink, and functions similar to a link on an internet web page. An 'anchor' is a placeholder in the document and must already exist in the destination file. This is discussed further in Part 3 of this paper. The full text of the hyperlink path to the data listing consists of:

1. directory path that the data listing file is located
2. name of the data listing file
3. 'anchor' name, preceded by the '#' symbol.

Below is an example of a hyperlink path named HYPLNK that links Subject=700001 from the profiles to the AE listing, AE.pdf, in a subdirectory '_Data_Listings':

```
hypLnk='_Data_Listings\AE.pdf#700001'
```

The Compute Block can then assign this hyperlink path to the section title using the 'URL' functionality of the CALL DEFINE statement. In this application, the hyperlink variable, HYPLNK, is unique for every combination of Subject and Section.

```
proc report data=table1 split='~' style(column)=[LINKCOLOR = _undef_ ];
column  subject hyplnk section ColVar1 ColVar2 Var1fmtVal;
  define subject /      order noprint;
  define hyplnk /      order noprint;
  define section /      order noprint;
  define Var1fmtVal /  noprint;
  define ColVar1 /'' display style(column)=[cellwidth=1in font_face='Courier' font_size=.1 just=right];
  define ColVar2 /'' display style(column)=[asis=on cellwidth=6.2in font_face='Courier' font_size=1.25 ];

compute ColVar2 ;
  call define(_col_,"URL",hyplnk);
  CALL DEFINE(_col_,'STYLE','STYLE=[background=lightgrey FOREGROUND=black]);
endcomp;
```

ADDING GRAPHIC FILES/PLOTS

The SAS system of macros in Base SAS® invoked by the statement %ANNOMAC allows us to easily create the color coded bar representing various key time points in the study. The bar image is output as the JPG file 'back_img.jpg' and looks like:



Code to do this is below.

```
filename backgrnd 'back_img.jpg';
%Annomac;      *<-- SAS ANNOTATE system macros;
data anno;
  length color $6.;
  %bar(0,0,40,45,orange,0,s);*<-- horizontal bar from 0%-40% (vertical 0%-45%);
  %bar(40,0,90,45,green,0,s);*<-- horizontal bar from 40%-90% (vertical 0%-45%);
  %bar(90,0,100,45,yellow,0,s);*<-- horizontal bar from 90%-100% (vertical 0%-45%);
run;

goptions device=gif gsfname=backgrnd hsize=7.7in vsize=0.15in ; *<-- height/width of actual
image;
proc gslide anno=anno;
run;
quit;
```

Utilizing the formatting variable, Var1fmtVal, the image is selectively imported to a specific row via the COMPUTE BLOCK, after a 'placeholder' text value. Note that the placeholder text is the same color as the background and thus does not visibly show up on the profile (i.e. foreground= lightgrey, background=lightgrey), but is visible in the PROC PRINT output displayed earlier.

```
proc report data=table1 split='~' style(column)=[LINKCOLOR = _undef_ ];
column  subject hyplnk section ColVar1 ColVar2 Var1fmtVal;

  define ...
  define ...
  define ...

compute Var1fmtVal ;
  if Var1fmtVal='WHITEb' then do;
    CALL DEFINE(_col_ -1, 'STYLE',
      'STYLE=[FOREGROUND=lightgrey background=lightgrey postimage="back_img.jpg"]');
  end;
endcomp;
```

Adding vital signs plots is a relatively straightforward task, once the plots have been created and stored as external files. For simplicity, our example creates a new PDF file for each subject containing the plots. Using the ODS PDF TEXT statement, section information with hyperlinks is first displayed, followed by three graphic files.

The below code gives an example of how ODS PDF TEXT completes this task for a specific subject, 700001. Note that the imported graphic files are named 'vs700001.png', 'vd700001.png' and 'vh700001.png'.

```
%let vs_link='_Data_Listings\VS.pdf#700001';          *<-- hyperlink to listing VS and subject;
ods pdf file='_Patient_Profiles\Subject_700001_1VS.pdf' notoc;

ods pdf text="^S={just=center FOREGROUND=blue font_weight=bold
font_face=Courier font_size=1.25 URL=&vs_link} Vital Signs [VS]";

ods pdf text=" ";
ods pdf text="^S={just=center postimage='Graph_Tmp\vs700001.png'} ";
ods pdf text=" ";
ods pdf text="^S={just=center postimage='Graph_Tmp\vd700001.png'} ";
ods pdf text=" ";
ods pdf text="^S={just=center postimage='Graph_Tmp\vh700001.png'} ";
ods pdf close;
```

COMBINE ALL PROFILES

In our example, all patient profile PDF files are combined into one master PDF file. Although there are various external software solutions to do this, the general steps are outlined below.

1. Name the individual output files so that they are sorted correctly alphabetically
2. Combine all PDFs into a single PDF file using external software
3. Name the single PDF file '**Patient Profiles.pdf**'
4. Place the single PDF file in the appropriate directory for hyperlinks to function.

Although combining all the PDF files into one is not necessary, it does reduce the task of naming hyperlinks to and from the patient profiles. An alternative approach is to create a separate document that is solely used as a table of contents for subject numbers with hyperlinks to all relevant PDF files associated with them. This could be accomplished using the ODS PDF TEXT statement described previously, but is outside the scope of this paper.

PART 3: DATA LISTINGS

The requirements of the data listings are that:

1. Each data listing file represents each section in the profiles (i.e. DM.pdf, AE.pdf, CM.pdf, etc.)
2. Listings are divided by subject, with a new subject starting a new page.
3. The 'anchors' are the subject numbers
4. Hyperlinks exist in the listings that link back to the same subject number in main patient profile PDF.

An example of the AE data listing page for subject=700001 is below.

Protocol: XYZ123
Product: Compound ABC

Advanced Clinical

Dataset: AE

Unique Subject Identifier=700001

Reported Term for the Adverse Event	Serious Event	AE Start Date	AE End Date	Outcome	Intensity	Causality	If not related, then suspected cause	Treatment Required
HEADACHE	N	20100215	20100215	RECOVERED/RESOLVED WITH SEQUELAE	MILD	POSSIBLY RELATED		NONE
DIZZINESS	Y	20100301	20100303	RECOVERED/RESOLVED WITH SEQUELAE	MODERATE	POSSIBLY RELATED		HOSPITALIZATION REQUIRED
TOOTH EXTRACTION PAIN	N	20100302	20100305	RECOVERED/RESOLVED WITH SEQUELAE	MODERATE	NOT RELATED	TOOTH EXTRACTION	OTHER, SPECIFY
NAUSEA	N	20100401		NOT RECOVERED/NOT RESOLVED	MILD	NOT RELATED	UPSET STOMACH	MEDICATION REQUIRED

[Return to Main Patient Profile Listing](#)

The SAS procedure PROC PRINT serves our needs for this task.

1. Item 1 above is accomplished with the ODS PDF "file=" statement.
2. Item 2 is accomplished by using the BY statement in PROC PRINT.
3. For item 3, anchors are assigned using the ODS PDF "anchor=" statement.
4. In the same way ODS PDF TEXT was previously used in the vital signs section title to link to a data listing, item 4 is satisfied by linking back to the profiles.

Below is sample PROC PRINT and ODS code for the AE data listing. This code is part of a MACRO with the values of "AE" and "700001" as macro parameters passed to the MACRO. The values of these parameters are shown here for illustration purposes.

```
ods listing close;
ods pdf file="_Data_Listings\AE .pdf" ;

ods noproctitle;
ods escapechar '^';
ods pdf anchor="700001";

%let subhyplk=%str('&masterfile.#700001');

proc print data=temp1(where=(Subjid="700001")) split='~' label
  style(table)=[OUTPUTWIDTH=10in ]
  style(data)=[just=left ]
  style(HEADER)=[just=left font_weight=bold] noobs;
by Subjid;
var &varlst / style(column)=[just=left BACKGROUND=white ];
run;
ods pdf text=" ";
ods pdf text="^S={just=center FOREGROUND=blue font_size=4 font_weight=bold URL=&subhyplk }
  Return to Main Patient Profile Listing";

ods pdf close;
ods listing ;
```

Note that the above code references a few externally defined MACRO variables, namely:

1. **&masterfile** – the directory path and name of the profiles document relative to the listing (ex: **.\Patient Profiles.pdf**)
2. **&varlist** – variables to be printed by PROC PRINT

Using the MACRO facility to produce data listings simplifies the addition of listings and variables within the each listing, as well as the ability to run all data listings within one file. Sample SAS code to produce data listings for adverse events, demographics, medication use and vital signs is given below.

```
options mprint nonumber notes linesize=256 ps=68 pageno=1 nodate
orientation=landscape nocenter missing='';          *<-- standard options;
*****
** Company:   Advanced Clinical
** Author:    wconover
** Date :     11-7-2009
** Purpose:   Creates Listings from SAS datasets with hyperlinks to MASTERFILE
** Notes:     user must define:
**             input LIBNAME,
**             masterfile,
**             MACRO calls to produce individual listings (end of program)
```

```

**          title/footnote information (search for 'MODIFY HERE')
*****;

*--MODIFY HERE-----*;
libname indata 'data';
%let masterfile=..\Patient Profiles.pdf;*<-- Name of main Patient Profile document;
*--end MODIFY HERE-----*;

*-----*
macro: PRINTALL for
outputting PDFs of datasets
*-----*;
%macro printall(datain=datain,
                outname=outname,
                prog_name=prog_name,
                byvar=byvar,
                sortvar=sortvar,
                varlst=varlst);

*--MODIFY HERE: title/footnote information-----*;
    title1 "Protocol: XYZ123" justify=right "Advanced Clinical" ;
    title2 "Product: Compound ABC";
    title4 justify=center "Dataset: &outname";
    footnote1 "Run Date: &sysdate-&sysstime" j=r "Page ^{thispage} of ^{lastpage}";
*--end: MODIFY HERE title/footnote information-----*;

*---nested macro for PROC PRINT-----*;
%macro printmac(subj); *<-- creates new page for each subject;
    ods pdf anchor="&subj";
    %let subhyplk=%str('&masterfile.#&subj');
    proc print data=temp1(where=(&byvar="&subj"))
        split='~'
        label
        style(table)=[OUTPUTWIDTH=10in]
        style(data)=[just=left ]
        style(HEADER)=[just=left font_weight=bold] noobs;
    by &byvar;
    var &varlst / style(column)=[just=left BACKGROUND=white ];
    run;
    ods pdf text=" ";
    ods pdf text="^S={just=center FOREGROUND=blue font_size=4
font_weight=bold URL=&subhyplk } Return to Main Patient Profile Listing";

```

```

    %mend printmac;
*---end: nested macro for PROC PRINT-----*;

data temp1;
    set &datain; *<-- bring in dataset for listing;
    keep &byvar &sortvar &varlst;
run;
proc sort;
    by &byvar &sortvar;
run;
proc sort data=temp1 out=temp2(keep=&byvar) nodupkey;
    by &byvar;
run;

*---create listing-----*;
ods listing close;
ods pdf file="_Data_Listings\&outname..pdf" ;
ods noproctitle;
ods escapechar '^';
data temp2;
    set temp2;
    call execute('%printmac('||&byvar||')'); *<-- call for each subject;
run;
ods pdf close;
ods listing ;
*---end: create listing-----*;
%mend printall;
*----end: macro PRINTALL-----*;

*-----MACRO: %printall() definitions-----*
MARCO: PRINTALL to create listings
%printall( datain=%str( ), /*input dataset for listing*/
           outname=, /*Output File Name, identical to RAW dataset name,
                    also used in title*/
           prog_name=, /*Program name used in footnote*/
           byvar=, /*PROC PRINT By Variable, typically SUBJECT*/
           sortvar=, /*sort order for listing within BYVAR*/
           varlst=) /*variables to be displayed in addition to BYVAR*/
*-----MODIFY BELOW HERE-----*;

%printall(datain=%str(indata.ae),
          outname=AE,

```

```

        prog_name=AE,
        byvar=USUBJID,
        sortvar=USUBJID AESTDT AETERM,
        varlst= AETERM AESER AESTDT AEENDT AEOU AESEV AEREL AESUSP AETRT );

%printall (datain=%str(indata.CM),
          outname=CM,
          prog_name=CM,
          byvar=USUBJID,
          sortvar=USUBJID CMSTDT CMENDT CMTRT,
          varlst= CMNONE CMTRT CMSTDT CMENDT CMONGYN CMDOSE CMDOSU CMOTHU CMDOSFRQ
CMFRQOTH CMROUTE CMROUOT CMINDC);

%printall (datain=%str(indata.dm),
          outname=DM,
          prog_name=DM,
          byvar=USUBJID,
          sortvar=USUBJID ,
          varlst= RFSTDT AGE AGEU SEX RACE RACEOTH ETHNIC);

%printall (datain=%str(indata.VS),
          outname=VS,
          prog_name=VS,
          byvar=USUBJID,
          sortvar=USUBJID VSdT,
          varlst= VSdT VSHT VSHT_U VSWT VSWT_U VSTEMP VSTEMPU VSRR VSBMI VSHR VSSYS
VSDIA);
*-----end: MODIFY HERE-----*

```

CONCLUSION

One thing to note before venturing off to program patient profiles is that with the advent of SDTM datasets, and the tools available in SAS 9.2, MACROS are currently available from the SAS website to produce very descriptive profiles with very little programming effort. A value of the methods discussed in this paper is that the source datasets do not need to be in SDTM format, and an older version of SAS can be used, namely 9.1.3.

The scope of this paper is to present the wide variety of SAS tools available to develop user friendly profiles that are visually aesthetic, functional, and easy to use. With the combination of graphic data summaries linked to detailed listings, the profiles become a powerful tool that deliver information quickly compared to traditional methods.

The working example outlined in this paper, including test SAS datasets and SAS programs, is available by contacting the author.

REFERENCES

[Make Your Tables Pop: Embedding Micrographics in PROC REPORT Output](#) - Pete Lund, Looking Glass Analytics, Olympia, WA.

[PDF Can be Pretty Darn Fancy: Tips and Tricks for the ODS PDF Destination](#) - Pete Lund, Looking Glass Analytics, Olympia, WA

Advanced PROC REPORT: Getting Your Tables Connected Using Links - Part II Practicum - Arthur L. Carpenter
California Occidental Consultants

Building and Using User Defined Formats - Arthur L. Carpenter California Occidental Consultants, Oceanside,
California

Create Descriptive Documentation with Hyper-links for SAS Database - Don (Dongguang) Li, NCIC-CTG, Queen's
University, Kingston, Canada

Beyond the Basics: Advanced PROC REPORT Tips and Tricks - Allison McMahon, SAS Institute Inc., Cary, NC

Incorporating PROC FORMAT, PROC REPORT, and ODS Style Definitions to Track Sources of Records with Poor
Data - Thaer Baroud, MHSA, MA, John Senner, PhD, Paul Johnson, MSPH, Little Rock, Arkansas

Graphical Patient Profiles - David Stokar, Enzon Pharmaceuticals Inc., Piscataway, NJ 08817

Subject Profiles Generated with Macro Profile System (MPS) - John Iwaniszek, Stat-Tech Services, LLC, Chapel Hill,
North Carolina

Don't Be a SAS Dinosaur: Modernize Your SAS Programs - Warren Repole

Getting Started with, and Getting the Most out of, SAS ODS PDF: No Mastery of PROC TEMPLATE Required -
LeRoy Bessler, Assurant Health, Milwaukee, USA

ACKNOWLEDGMENTS

Special thanks are due to Daniel Bonzo, Marichu Escobar, Tosh Kameda and Gie Hubilla of the Xenoport Biometrics and Data Management team for the profile design and functionality specifications. Also, thanks to the Advanced Clinical Biostatistics team for their review and feedback of this paper and SAS programming techniques.

CONTACT INFORMATION

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

Name:	William Conover
Enterprise:	Advanced Clinical
Address:	1200 Lakeside Drive
City, State ZIP:	Bannockburn, IL, 60015
E-mail:	wconover@acrxs.com
Web:	http://www.advancedclinical.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

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