

A Novel Approach to Patient Profiling

Duong Tran, Independent Contractor, London, UK

ABSTRACT

Patient profiles are individualized display of patient data, often in a very visualize format. It is primarily use for safety reviews and as part of a regulatory submission in the pharmaceuticals industry. There are numerous published SAS® papers on patient profiling. This is another that describes a novel approach that is simple and flexible. The technique is more about expressing your creativeness rather than your ability to programs.

INTRODUCTION

Software Applications are written to serve a particular purpose. Often therefore by utilising multi applications ones can improve the efficiency and quality of the task at hand. This paper discusses how to use SAS and EXCEL® together to produce stunning profiling report.

METHOD

Newer version of EXCEL can save file in HTML (Hypertext Markup Language), XML (Extendible Markup Language) formats etc. Given that these are just plain text - SAS can read and modify it. So first what you need to do is to design the template in EXCEL as an example below. Note [HGB1] etc is a unique tag to that cell. Then save the file in HTML format (see example in Appendix 1). This will be the template for the patient profile. You can then use SAS to post-process this file, replacing the tags with macro values. Typically there are not that many fields in a profile so this technique is more than practical for this purpose.

Example of a tagged patient profile template

Figure 1

Patient Profile							
	[PATIENT]						
	[ASRT]						
[TREAT]	[TREAT1]	[TREAT2]	[TREAT3]	[TREAT4]	[TREAT5]	[TREAT6]	[TREAT7]
Systolic	[SYS1]	[SYS2]	[SYS3]	[SYS4]	[SYS5]	[SYS6]	[SYS7]
Diastolic	[DIA1]	[DIA2]	[DIA3]	[DIA4]	[DIA5]	[DIA6]	[DIA7]
HR	[HR1]	[HR2]	[HR3]	[HR4]	[HR5]	[HR6]	[HR7]
AST (SGOT)	[AST1]	[AST2]	[AST3]	[AST4]	[AST5]	[AST6]	[AST7]
ALT (SGPT)	[ALT1]	[ALT2]	[ALT3]	[ALT4]	[ALT5]	[ALT6]	[ALT7]
Haemoglobin	[HGB1]	[HGB2]	[HGB3]	[HGB4]	[HGB5]	[HGB6]	[HGB7]
	0	14	28	42	56	70	84
	Study Day						
Adverse Event of Interest	Onset Day	Duration (days)	Severity				
RASH	[RASH1]	[RASH2]	[RASH3]				
DIZZINESS	[DIZZ1]	[DIZZ2]	[DIZZ3]				
▶ \ {patient} /							

Example of a populated patient profile

Figure 2

Patient Profile							
Patient: 001							
Age: 47; Sex: Male; Race: Caucasian; Treatment: Active							
Active	■	□	■	■	□	■	■
Systolic	102	120	90	150	110	99	170
Diastolic	132	130	80	120	130	89	110
HR	192	159	170	155	120	145	180
AST (SGOT)	108.3	.	89.5	.	115.3	.	95.9
ALT (SGPT)	78.1	.	67.2	.	54.9	.	45.6
Haemoglobin	10.40	11.50	13.20	11.90	12.60	12.10	10.90
	0	14	28	42	56	70	84
Study Day							
Adverse Event of Interest	Onset Day	Duration (days)	Severity				
RASH	3	10	Severe				
DIZZINESS	5	.	Mild				
001/							

SAS code to fill the template

```

data pprofile;
  length treat $30;
  patient = '001';
  age = 47;
  sex = 'Male';
  race = 'Caucasian';
  treat = 'Active';
  treat1 = 'Y'; treat2 = 'N'; treat3 = 'Y'; treat4 = 'Y'; treat5 = 'N'; treat6 = 'Y'; treat7 = 'Y';
  sys1 = 102; sys2 = 120; sys3 = 90; sys4 = 150; sys5 = 110; sys6 = 99; sys7 = 170;
  dia1 = 132; dia2 = 130; dia3 = 80; dia4 = 120; dia5 = 130; dia6 = 89; dia7 = 110;
  hr1 = 192; hr2 = 159; hr3 = 170; hr4 = 155; hr5 = 120; hr6 = 145; hr7 = 180;
  ast1 = 108.3; ast2 = .; ast3 = 89.5; ast4 = .; ast5 = 115.3; ast6 = .; ast7 = 95.9;
  alt1 = 78.1; alt2 = .; alt3 = 67.2; alt4 = .; alt5 = 54.9; alt6 = .; alt7 = 45.6;
  hgb1 = 10.4; hgb2 = 11.5; hgb3 = 13.2; hgb4 = 11.9; hgb5 = 12.6; hgb6 = 12.1; hgb7 = 10.9;

  rash1 = 3; rash2 = 10; rash3 = 'Severe';
  dizz1 = 5; dizz2 = .; dizz3 = 'Mild';

output;
patient = '002';
treat = 'Placebo';
output;
run;

%macro pprofile(patient=);

  data _null_;
    set pprofile;

    if patient = "&patient";

    call symput('patient' , patient);
    call symput('asrt' , 'Age: ' || compress(put(age,3.)) ||
      '; Sex: ' || sex ||
      '; Race: ' || race ||
      '; Treatment: ' || treat
    );

    call symput('treat' , compress(treat));

  %do i = 1 %to 7;
    if treat&i = 'Y' then call symput("treat&i" , '%bquote(&#x25A0)');
    else call symput("treat&i" , '%bquote(&#x25A1)');
  %end;

```

```

call symput("sys&i", compress(put(sys&i, 8.)));
call symput("dia&i", compress(put(dia&i, 8.)));
call symput("hr&i", compress(put(hr&i, 8.)));

call symput("ast&i", compress(put(ast&i, 8.1)));
call symput("alt&i", compress(put(alt&i, 8.1)));

call symput("hgb&i", compress(put(hgb&i, 8.2)));

%end;
%do i = 1 %to 2;
call symput("rash&i", compress(put(rash&i, 8.)));
call symput("dizz&i", compress(put(dizz&i, 8.)));
%end;
call symput("rash3", rash3);
call symput("dizz3", dizz3);

run;

data ppfile2;
infile "patient_profile.xml" length=linelen lrecl=32767 recfm=v;
input linein $varying2000. linelen;
run;

data _null;
set ppfile2;

file "p&patient..xls" lrecl=2000;

select;
when(index(linein, "[PATIENT]")) linein = tranwrd(linein, "[PATIENT]" , "Patient: &patient" );
when(index(linein, "[ASRT]")) linein = tranwrd(linein, "[ASRT]" , "&asrt" );

when(index(linein, "[TREAT]")) linein = tranwrd(linein, "[TREAT]" , "&treat" );
when(index(linein, "{patient}")) linein = tranwrd(linein, "{patient}" , "&patient" );

%do i = 1 %to 7;
when(index(linein, "[TREAT&i]")) linein = tranwrd(linein, "[TREAT&i]" , "&&treat&i" );
when(index(linein, "[SYS&i]")) linein = tranwrd(linein, "[SYS&i]" , "&&sys&i" );
when(index(linein, "[DIA&i]")) linein = tranwrd(linein, "[DIA&i]" , "&&dia&i" );
when(index(linein, "[HR&i]")) linein = tranwrd(linein, "[HR&i]" , "&&hr&i" );
when(index(linein, "[AST&i]")) linein = tranwrd(linein, "[AST&i]" , "&&ast&i" );
when(index(linein, "[ALT&i]")) linein = tranwrd(linein, "[ALT&i]" , "&&alt&i" );
when(index(linein, "[HGB&i]")) linein = tranwrd(linein, "[HGB&i]" , "&&hgb&i" );
%end;
%do i = 1 %to 3;
when(index(linein, "[RASH&i]")) linein = tranwrd(linein, "[RASH&i]" , "&&rash&i" );
when(index(linein, "[DIZZ&i]")) linein = tranwrd(linein, "[DIZZ&i]" , "&&dizz&i" );
%end;
otherwise;
end;

linelen =length(linein);
put linein $varying2000. linelen;
run;

%mend;

%ppfile(patient=001);
%ppfile(patient=002);

```

SUMMARY

The technique presented in this paper is simple but very flexible. The beauty with this technique is that the presentation display lies within Excel capabilities and your creativeness to exploit them with minimum programming involvement.

Note graphics have not been demonstrated. This was left out for you to experiment.

CONCLUSION

This paper has demonstrated that by using the processing power of SAS and the presentation capabilities of EXCEL couple with your creativeness, you can achieves stunning results.

In this paper I have applied the discussed technique to produce Patients Profile but obviously this can be applied in other applications.

REFERENCES

- [1] SUGI 31: Subjects Profiles Generated with Macro Profile System (MPS) (John Iwaniszek)
- [2] SUGI 30: ODS MARKUP: The SAS® Reports You've Always Dreamed Of (Eric A Gebhart)
- [3] SUGI 26: A Novel Approach to Developing a Patient Profile Reporting Application (Steve Light and Paul Gilbert)
- [4] Pharmasug 2007: A Low Cost Graphical Patient Profiler with SAS MetaViewApplet (Ya Huang)
- [5] Pharmasug 2006: Patient Profile - A Menu-Driven System (Adel Fahmy)
- [6] Pharmasug 2005: Use ODS Generating Patient Profiles (Haibin Shu, Young Zhuge)
- [7] Pharmasug 2003: A Modular Approach to Develop Patient Profile Application (Fuping Peng, Carlos Perdomo)
- [8] PhUSE 2007: %RiTEN (Duong Tran)
- [9] www.w3.org

ACKNOWLEDGMENTS

I would like to thank Gerry Downey for his comments on this paper.

CONTACT INFORMATION

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

Duong Tran

Phone: 07752326413

Email: trand000@aol.com

www.tranz.co.uk

Brand and product names are trademarks of their respective companies.

APPENDIX 1

Extract of Excel File Saved in HTML Format

```
<html xmlns:o="urn:schemas-microsoft-com:office:office"
xmlns:x="urn:schemas-microsoft-com:office:excel"
xmlns="http://www.w3.org/TR/REC-html40">

<head>
<meta http-equiv=Content-Type content="text/html; charset=windows-1252">
<meta name=ProgId content=Excel.Sheet>
<meta name=Generator content="Microsoft Excel 11">
<link rel=File-List href="duml_files/filelist.xml">
<link rel=Edit-Time-Data href="duml_files/editdata.mso">
<link rel=OLE-Object-Data href="duml_files/oledata.mso">
<!-- [if !mso] >

<meta http-equiv=Content-Type content="text/html; charset=windows-1252">
<meta name=ProgId content=Excel.Sheet>
<meta name=Generator content="Microsoft Excel 11">
<link rel=File-List href="patient_profile_files/filelist.xml">
<link rel=Edit-Time-Data href="patient_profile_files/editdata.mso">
<link rel=OLE-Object-Data href="patient_profile_files/oledata.mso">

</head>

<body link=blue vlink=purple>

<table x:str border=0 cellpadding=0 cellspacing=0 width=566 style='border-collapse:

<tr height=17 style='height:12.75pt'>
<td height=17 class=xl25 style='height:12.75pt'>[TREAT] </td>
<td class=xl39 style='border-top:none'>[TREAT1] </td>
<td class=xl39 style='border-top:none;border-left:none'>[TREAT2] </td>
<td class=xl39 style='border-top:none;border-left:none'>[TREAT3] </td>
<td class=xl39 style='border-top:none;border-left:none'>[TREAT4] </td>
<td class=xl39 style='border-top:none;border-left:none'>[TREAT5] </td>
<td class=xl39 style='border-top:none;border-left:none'>[TREAT6] </td>
<td class=xl39 style='border-top:none;border-left:none'>[TREAT7] </td>
</tr>

</table>

</body>

</html>
```