

Paper 261-2008

Experimenting with the ODS DATA Step Object (Part II)

Richard Koopmann Jr.

ABSTRACT

With ODS DATA Step Object, SAS programmers can turn raw data into production quality reports ready for distribution. The ODS DATA Step Object is a powerful Base SAS feature (available in preproduction for 9.1.3) that is able to generate production-quality PDF reports directly since it provides SAS programmers simultaneous control over report content, layout, and formatting.

This paper tracks the re-creation of a legacy report to a completely automated and scalable SAS report using the ODS DATA Step Object. Macros are used throughout the code to reduce code volume and complexity making code maintenance and modification more palatable.

The code discussed in this paper was developed with SAS 9.1.3-SP4 running under Windows XP Professional-SP2 and was developed specifically for the PDF destination, you may experience some nuances when applying these techniques to other destinations under different systems.

Keywords: ODS DATA Step Object, DATA Step Component Interface (DSCI), macro, DATA NULL

BACKGROUND

Recently, I was tasked with developing a SAS program to phase out a tedious, error-prone reporting process comprised of transcribing a few key numbers from hundreds of item summary tables to a dozen one-page summary tables. To reduce end-user confusion, the revised reports needed to resemble the current content, layout, and format as much as possible.

Several reporting methods were explored, specifically `DATA _NULL_` reporting which allows the conditional use of variables and provides the layout control required, and traditional ODS reporting with both `Proc REPORT` and `Proc TABULATE`.

Limitations of these methods were quickly discovered. The key statistics change across subgroups and would span numerous item summary tables, so a single tabulate or report procedure was not sufficient. Since the final report needs to be contained within a single page table in PDF, multiple procedure calls would not be possible. This limitation is overcome with `DATA _NULL_` reporting, but the monospace appearance of reports is unacceptable for current reporting needs.

Fortunately, ODS DATA Step Object's ability to blend the layout control of `DATA _NULL_` reporting with the formatting control of traditional ODS meets all current reporting requirements. For more background, see *Experimenting with the ODS DATA Step Object (Part I)*.

EXPLORING REPORTING OPTIONS

The features available via three reporting options were explored for re-creating the manual legacy report:

DATA _NULL_ The familiar DATA Step programming options are available to produce bland, text-only reports. These are very customizable in terms of content and layout, but lack variable-width font, rich-text formatting (colors, weights, decorations), and hyperlinking. In `DATA _NULL_` reporting, the programmer has great control over the layout and content of the report if they are willing to accept bland, monospaced format. In fact, the layout of the report is completely dependent upon the use of a monospace font—ever tried to copy and paste SAS output into an email? To meet the PDF requirement, however, we must *manually* convert the txt file to a PDF—not scalable.

Traditional ODS Wrapping an ODS envelope around existing reporting procedures (typically `Proc REPORT` or `Proc TABULATE`) is a quick way to get fair looking reports. These reports make use of variable-width fonts, rich-text formatting, and hyperlinking. The report layout is fairly locked, however, with no more than one procedure contributing to an output table. The two

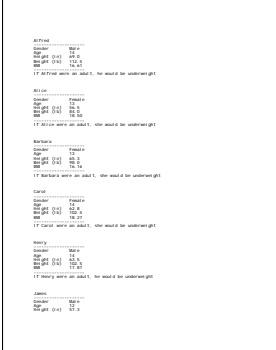
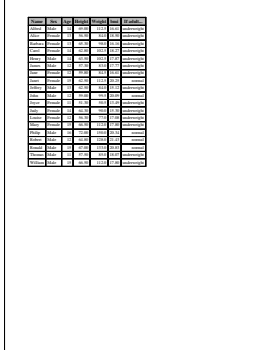
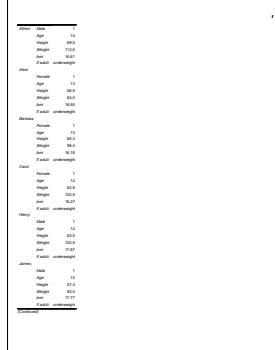
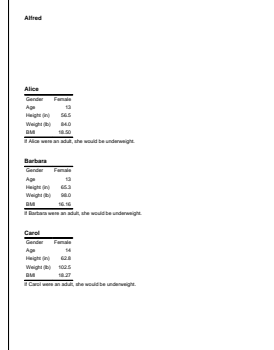
main reporting procedures have their benefits and limitations; `Proc REPORT` supports calculated variables but is limited to horizontal layout while `Proc TABULATE` allows for horizontal and vertical layouts, but not calculated variables.

ODS DATA Step Object As its name would suggest, the ODS DATA Step Object is a DATA Step object that blends `DATA _NULL_` reporting with typical ODS reporting features. The power of the DATA Step provides the programmer with significant control over the layout and content of a report while the ODS allows them with the usual control over typography. In the examples used herein, the ODS DATA Step Object is implemented within a `DATA _NULL_` data step, but could also be implemented in data steps producing SAS data sets.

Feature	DATA _NULL_	Traditional ODS	ODS DATA Step Object
Variable-width fonts		X	X
Rich-text formatting		X	X
Adjustable layout	X	(Proc TABULATE)	X
Variable content	X	(multiple procs)	X
Calculated variables	X	(Proc REPORT)	X
Multiple data sources	X	(multiple procs)	X
Production-quality output		X	X

SAMPLE SASHELP.CLASS REPORTS

I quickly developed four sample reports on the `SASHELP.CLASS` data set to demonstrate the features and limitations of the three reporting options. The code to produce these sample reports is presented in Listing 4 in the Appendix.

			
DATA _NULL_	Proc REPORT	Proc TABULATE	ODS DATA Step Object

PREPARING THE DATA

Rather than the source data directly, the current approach uses several intermediary data sets to construct a summary data set for writing the actual ODS DATA Step Object report. Summarization methods will depend on the structure of the source data and the desired summarization types, this report requires three different summarization types:

Top *n* The number of respondents selecting the top *n* response options are compared to the total number of respondents to that item. In a 5-point agreement scale, for example, one might collapse the top two or three into a single category.

Single Respondents selecting a pre-chosen response option are compared to the total number of respondents to that item. In a 5-point satisfaction scale, for example, one might want to determine the rate of *Very Dissatisfied* for a particular item.

Modal The most-frequently chosen response option is examined. Since the mode for an item likely changes by subgroup, this will require frequency rates for each response option of interest.

The summary data set for this report consists of too many variables to list (well over 600), but the pattern of variable names and meaning is consistent:

var The percent for the given subgroup (*var_1* / *var_n*).

var_1 The number of observations with the response of interest for the given subgroup.

var_n The number of observations with a valid response for the given subgroup.

var_cu The percent for the population overall (*var_cu_1* / *var_cu_n*).

var_cu_1 The number of observations with the response of interest for the population overall.

var_cu_n The number of observations with a valid response for the population overall.

There are a handfull of rank variables denoted by a preceding *r* in the variable name, these are used to generate tables rows in the ODS DATA Step Object report for **Top *n*** and **Modal** summarization variables. These are discussed further below.

ODS DATA Step Object

If you look at the ODS DATA Step Object code in Listing 4 (in Appendix), you'll note that there is a fair amout of repetitious code, start row, start cell, format text, close cell, close row, start row . . . :

Listing 1: Repetitive code in ODS DATA Step Object reporting

```
obj.row_start();
  obj.cell_start( inhibit: 'BLR' );
    obj.format_text( text: 'Gender', overrides: 'font_face=arial' );
  obj.cell_end();
  obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
    obj.format_text( text: put(sex, $sex.), overrides: 'font_face=arial' );
  obj.cell_end();
obj.row_end();

obj.row_start();
  obj.cell_start( inhibit: 'BLR' );
    obj.format_text( text: 'Age', overrides: 'font_face=arial' );
  obj.cell_end();
  obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
    obj.format_text( text: put(age, 3. ), overrides: 'font_face=arial');
  obj.cell_end();
obj.row_end();
```

To help tame the volume of coded needed to create the desired report, macros were developed to standardize and isolate code:

Listing 2: Custom ODS DATA Step Object report macros

```
%let h1 = %str(font=(arial, 14 pt, bold roman));
%let h2 = %str(font=(arial, 12 pt, italic narrow));
%let normal = %str(font=(times, 10 pt, roman));
%let head = %str(font=(arial, 10 pt, bold));
%let stub = %str(font=(arial, 9 pt));
%let cell = %str(font=(arial, 9 pt, narrow));
%let foot = %str(font=(arial, 8 pt, narrow) foreground=silver);
%let warn = %str(font=(arial, 10 pt, bold roman) foreground=red);

%macro textRow(text,indent=0);
  %let indent = %eval(&indent * 2);
  obj.row_start();
    obj.cell_start( inhibit: 'TBLR', overrides: 'cellwidth=5.35in', column_span: 4 );
      obj.format_text( text: "&text", overrides: "&cell leftmargin=&indent em");
    obj.cell_end();
  obj.row_end();
%mend textRow;

%macro varRow(var, label=%str( ),indent=0);
  %let indent = %eval(&indent * 2);
  %if &label eq %str( ) %then %let label = %vlabel(aos,&var);
  *****TABLE COLUMNS*****;
  obj.row_start();
    obj.cell_start( inhibit: 'BLR' );
      obj.format_text( text: "&label", overrides: "&stub leftmargin=&indent em" );
    obj.cell_end();
    obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
      obj.format_text( text: put(&var._cu, percent8.1), overrides: "&cell foreground=silver" );
    obj.cell_end();
```

```

obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
  obj.format_text( text: put(&var, percent8.1), overrides: "&cell" );
obj.cell_end();
obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
  obj.format_text( text: catx(' / ', put(&var._1, comma8.), put(&var._n, comma8.))
    , overrides: "&cell" );
  obj.cell_end();
obj.row_end();
%mend varRow;

%macro vlabel(dsn,var);
  %local dsid vlabel;
  %let dsid=%sysfunc(open(&dsn));
  %let vlabel=%qsysfunc(varlabel(&dsid,%sysfunc(varnum(&dsid,&var))));
  %let dsid=%sysfunc(close(&dsid));
  &vlabel
%mend vlabel;

```

The first batch of %let statements are just font definitions, similar to styles in most word processors.

The %textRow macro generates code needed to produce a row of text in a table with its contents spanning four columns.

The %varRow macro generates code needed to produce a row summarizing a particular survey item for the current subgroup and provides an overall reference point.

The %vlabel macro simply returns the variable label for use in the DATA step.

After the supporting macros are defined, generating the ODS DATA Step Object report is relatively easy, although very iterative.

Listing 3: The ODS DATA Step Object report

```

1 ods escapechar="~";
2 ods listing close;
3 ods pdf file="ODSDSO_Report.pdf";
4 ods proclabel='Results from Alumni Outcomes Survey: One Year Out';
5 data _null_;
6   format &by $25.;
7   set &agg_by end=last ;
8   if _n_ = 1 then declare odsout obj();
9
10  obj.image( text: 'CU_Color_Horiz_Small.gif' );
11  obj.format_text( text: 'Results from Alumni Outcomes Survey: One Year Out', overrides: "&h2" );
12
13  *****SUBGROUP DESCRIPTION*****;
14  obj.format_text( text: group, overrides: "&h1" );
15  obj.format_text( text: catx(' ', 'Based on',put (didSubmit_1,comma8.-1)
16    , 'submissions from', put (didSubmit_n,comma8.-1), 'alumni'
17    , '( ' || put (didsubmit,percent7.1-1), 'response rate)'
18    , 'graduating from', put (graddate_min,yyq6.)
19    , 'thru', put (graddate_max,yyq6.) || '.' )
20    , overrides: "&normal");
21  obj.format_text( text: catx(' '
22    , catx(' ', '~S={foreground=red font_weight=bold}DO NOT CITE AFTER'
23    , put(intnx('year.7',"&sysdate9"d,1) - 1, mmdyy10.)
24    , '~S={foreground=black font_weight=light}Questions? Contact AssessmentOffice@capella.edu'
25    , "~S={font_style=italic}Generated &sysdate9.." )
26    , overrides: "&normal");
27
28  obj.layout_gridded ( width: '8.5in', height: '11in', columns: 1, row_gutter: '0in' );
29  obj.region();
30
31  obj.table_start( label: trim(group), overrides: 'cellpadding=1.5pt' );
32
33  obj.row_start();
34  obj.cell_start( inhibit: 'TLR', overrides: 'cellwidth=5.35in' );
35  obj.cell_end();
36  obj.cell_start( inhibit: 'TLR', overrides: 'cellwidth=0.55in just=right' );
37  obj.format_text( text: 'CU %', overrides: "&head foreground=silver" );
38  obj.cell_end();
39  obj.cell_start( inhibit: 'TLR', overrides: 'cellwidth=0.55in just=right' );
40  obj.format_text( text: '%1', overrides: "&head" );
41  obj.cell_end();
42  obj.cell_start( inhibit: 'TLR', overrides: 'cellwidth=0.80in just=right' );
43  obj.format_text( text: 'n / N1', overrides: "&head" );
44  obj.cell_end();
45  obj.row_end();
46
47  *****TABLE ROWS*****;
48  %varRow(Q1, label=Employed);

```

```

49 %varRow(Q6_1, indent=1, label=%str(Full-time));
50 do; *modal summary;
51   if rq7_1f_1 = 1 then do; %varRow(Q7_1f_1, indent=1, label=Top job field: 1); end;
52   if rq7_1f_2 = 1 then do; %varRow(Q7_1f_2, indent=1, label=Top job field: 2); end;
53   if rq7_1f_3 = 1 then do; %varRow(Q7_1f_3, indent=1, label=Top job field: 3); end;
54   if rq7_1f_4 = 1 then do; %varRow(Q7_1f_4, indent=1, label=Top job field: 4); end;
55   if rq7_1f_5 = 1 then do; %varRow(Q7_1f_5, indent=1, label=Top job field: 5); end;
56 end;
57
58 do; *modal summary;
59   if rq8f_1 = 1 then do; %varRow(Q8f_1, indent=1, label=Top job class: 1); end;
60   if rq8f_2 = 1 then do; %varRow(Q8f_2, indent=1, label=Top job class: 2); end;
61   if rq8f_3 = 1 then do; %varRow(Q8f_3, indent=1, label=Top job class: 3); end;
62   if rq8f_4 = 1 then do; %varRow(Q8f_4, indent=1, label=Top job class: 4); end;
63   if rq8f_5 = 1 then do; %varRow(Q8f_5, indent=1, label=Top job class: 5); end;
64 end;
65
66 %varRow(Q9, indent=1, label=%str(Found job prior to starting their Capella degree));
67 %varRow(Q11, indent=1, label=%str(Reported annual salary of $50,000 or greater));
68 %varRow(Q27, indent=1, label=%str(Indicated Capella education prepared them for current job^{super 2}));
69 %varRow(Q14, indent=1, label=%str());
70 do; *modal summary;
71   if rQ15_1f_1 = 1 then do; %varRow(Q15_1f_1, indent=2, label=Top academic Rank: 1); end;
72   if rQ15_1f_2 = 1 then do; %varRow(Q15_1f_2, indent=2, label=Top academic Rank: 2); end;
73   if rQ15_1f_3 = 1 then do; %varRow(Q15_1f_3, indent=2, label=Top academic Rank: 3); end;
74   if rQ15_1f_4 = 1 then do; %varRow(Q15_1f_4, indent=2, label=Top academic Rank: 4); end;
75   if rQ15_1f_5 = 1 then do; %varRow(Q15_1f_5, indent=2, label=Top academic Rank: 5); end;
76 end;
77
78
79 %textRow(%str(Satisfied with aspects of current job^{super 3}), indent=1);
80
81 do i = 1 to 2; *top n summary;
82   if rq17_1 = i then do; %varRow(Q17_1, indent=2); end;
83   if rq17_2 = i then do; %varRow(Q17_2, indent=2); end;
84   if rq17_3 = i then do; %varRow(Q17_3, indent=2); end;
85   if rq17_4 = i then do; %varRow(Q17_4, indent=2); end;
86   if rq17_5 = i then do; %varRow(Q17_5, indent=2); end;
87   if rq17_6 = i then do; %varRow(Q17_6, indent=2); end;
88 end;
89
90 %varRow(Q12, indent=1, label=%str(Said their job is Directly or Moderately related to their Capella Degree field));
91
92 not_q1 = 1 - q1; not_q1_1 = q1_n - q1_1; not_q1_n = q1_n;
93 not_q1_cu = 1 - q1_cu; not_q1_cu_1 = q1_cu_n - q1_cu_1; not_q1_cu_n = q1_cu_n;
94 %varRow(not_Q1, label=Unemployed);
95 %varRow(Q2_1, indent=1, label=%str(Seeking employment))
96 %varRow(Q3, indent=2, label=%str(Seeking employment for 6 months of less));
97
98 %textRow(%str(Since earning their Capella University degree, graduates have^{super 4}));
99 do i = 1 to 2; *top n summary;
100   if rq18_1 = i then do; %varRow(Q18_1, indent=1); end;
101   if rq18_2 = i then do; %varRow(Q18_2, indent=1); end;
102   if rq18_3 = i then do; %varRow(Q18_3, indent=1); end;
103   if rq18_4 = i then do; %varRow(Q18_4, indent=1); end;
104   if rq18_5 = i then do; %varRow(Q18_5, indent=1); end;
105   if rq18_6 = i then do; %varRow(Q18_6, indent=1); end;
106 end;
107
108 %varRow(Q19, indent=1, label=%str(Upon earning Capella degree, sought professional licensure and/or certification));
109 %varRow(Q21, indent=1, label=%str(Ultimately got licenses and/or certified));
110
111 %varRow(Q23, indent=1, label=%str(Have or intended to complete additional degree/certification/course));
112
113 %textRow(Indicated their Capella University Degree provided them^{super 4});
114 do i = 1 to 4; *top n summary;
115   if rq25_1 = i then do; %varRow(Q25_1, indent=1); end;
116   if rq25_2 = i then do; %varRow(Q25_2, indent=1); end;
117   if rq25_3 = i then do; %varRow(Q25_3, indent=1); end;
118   if rq25_4 = i then do; %varRow(Q25_4, indent=1); end;
119   if rq25_5 = i then do; %varRow(Q25_5, indent=1); end;
120   if rq25_6 = i then do; %varRow(Q25_6, indent=1); end;
121   if rq25_7 = i then do; %varRow(Q25_7, indent=1); end;
122   if rq25_8 = i then do; %varRow(Q25_8, indent=1); end;
123   if rq25_9 = i then do; %varRow(Q25_9, indent=1); end;
124   if rq25_10 = i then do; %varRow(Q25_10, indent=1); end;
125   if rq25_11 = i then do; %varRow(Q25_11, indent=1); end;
126   if rq25_12 = i then do; %varRow(Q25_12, indent=1); end;
127   if rq25_13 = i then do; %varRow(Q25_13, indent=1); end;
128   if rq25_14 = i then do; %varRow(Q25_14, indent=1); end;
129   if rq25_15 = i then do; %varRow(Q25_15, indent=1); end;
130 end;
131

```

```

132 %varRow(Q28, label=%str(Overall, rated Capella above average5));
133 %varRow(Q29, label=%str(Would return to Capella if they could start over6));
134 %varRow(Q30, label=%str(Percieve others view their Capella degree as Average quality or better7));
135 %varRow(Q31, label=%str(Have recommended Capella to family, friends, or acquaintances8));
136
137 *****NOTES*****;
138 obj.row_start();
139   obj.cell_start( inhibit: 'BLR', column_span: 4);
140   obj.format_text( text: catx('n'
141     , 'Note: CU% represents the overall % for all Capella alumni in the specified timeframe.'
142     , catx(' ', '{super 1} n = number of respondents described'
143     , 'N = number of people providing a valid response to the item; % = n / N')
144     , catx(' ', '{super 2} Selected: Exceptionally well + More than adequately + Adequately'
145     , 'Unselected: Less than adequately, Very poorly.')
146     , catx(' ', '{super 3} Selected: Very Satisfied + Satisfied'
147     , 'Unselected: Neutral, Dissatisfied, Very Dissatisfied.')
148     , catx(' ', '{super 4} Selected: Strongly Agree + Agree'
149     , 'Unselected: Disagree, Strongly Disagree.')
150     , catx(' ', '{super 5} Selected: Excellent + Good'
151     , 'Unselected: Average, Poor.')
152     , catx(' ', '{super 6} Selected: Definitely yes + Probably yes'
153     , 'Unselected: Uncertain, Probably no, Definitely no.')
154     , catx(' ', '{super 7} Selected: Above average quality + Average quality'
155     , 'Unselected: Below average quality, I don't know.')
156     , catx(' ', '{super 8} Selected: Yes with some reservations + Yes without reservations'
157     , 'Unselected: No.')
158     )
159     , overrides: "&foot");
160   obj.cell_end();
161   obj.row_end();
162   obj.table_end();
163   obj.layout_end();
164   if not last then obj.page(); *force a page break;
165 run;
166 ods pdf close;
167 ods listing;

```

Some note on Listing 3, by line:

- 1-4 Close the listing window to save processor time and set up the PDF destination for output.
- 8 Declare an ODS output object.
- 10-26 Add branding and describe the current subgroup and terms of use.
- 28-29 Set up a gridded layout as specified, then start a new region.
- 31-45 Start the main table and write out the table header.
- 48-49 Use the %varRow macro to write out two rows of **Single** summaries with the second being indented one level.
- 50-56 Write out one row of **Modal** summary each indented one level.
- 79 Write out a row of text with a footnote.
- 81-88 Write out **Top two** (do i = 1 to 2) summary rows indented two levels.
- 92-94 Define a set of new variables for output.
- 138-161 The final row of the table contains the footnotes.
- 162-163 End the table and layout.
- 164 Insert page breaks between subgroups.
- 166-167 Close the PDF destination and re-open the listing window.

LEGACY VS. ODS DATA Step Object REPORT

Below, the old version of the report is presented on the left and the current report is presented on the right. Much of the formatting and layout, and all of the content was carried over with modifications added to make the report easier to understand and navigate.

CAPPELLA UNIVERSITY		Alumni Outcomes Survey: One Year Out	
SUBGROUP Alumni Outcomes Survey Results Summary			
Based on <i>n</i> responses from NNN alumni (response rate= <i>m</i> %) graduating from YYYYQ thru YYYYQ.			
DO NOT CITE AFTER MM/DD/YYYY			
Career Outcomes	%	n	N ¹
Employed	xx%	m	NNN
Full time	xx%	m	NNN
Job field: top job field	xx%	m	NNN
Job class: top job class	xx%	m	NNN
Found job prior to starting their Capella University degree	xx%	m	NNN
Reported salary of at least \$50,000/year	xx%	m	NNN
Indicated Capella education prepared them for current job ²	xx%	m	NNN
Unemployed	xx%	m	NNN
Seeking employment	xx%	m	NNN
Seeking employment for 6 months or less	xx%	m	NNN
Said their job is Directly or Moderately Related to their CU Degree field	xx%	m	NNN
Teach in a post-secondary setting	xx%	m	NNN
Academic rank: top academic rank	xx%	m	NNN
Satisfied with aspects of current job ³	xx%	m	NNN
First ranked aspect	xx%	m	NNN
Second ranked aspect	xx%	m	NNN
Since earning their CU degree, graduates have ⁴			
First ranked accomplishment	xx%	m	NNN
Second ranked accomplishment	xx%	m	NNN
Upon earning CU degree, sought professional licensure/certification	xx%	m	NNN
Ultimately got licensure/certified	xx%	m	NNN
Learning Outcomes	%	n	N ¹
Have or intended to complete additional degree/cert/course	xx%	m	NNN
Indicated their CU Degree provided them ⁵			
First ranked achievement	xx%	m	NNN
Second ranked achievement	xx%	m	NNN
Third ranked achievement	xx%	m	NNN
Fourth ranked achievement	xx%	m	NNN
Reflections on Capella University	%	n	N ¹
Rated Capella University above average overall ⁶	xx%	m	NNN
Would return to Capella if they could start over ⁷	xx%	m	NNN
Perceive others' views of their CU Degree as Average quality or above ⁸	xx%	m	NNN
Have recommended CU to family/friends/acquaintance ⁹	xx%	m	NNN

¹ into the number of respondents described. N = number of respondents presented the survey item.
² Selected: Exceptionally well + More than adequately + Adequately/Unselected: Less than adequately/Very poorly.
³ Selected: Very Satisfied + Satisfied/Unselected: Neutral/ Dissatisfied/ Very Dissatisfied.
⁴ Selected: Strongly Agree + Agree/Unselected: Disagree/ Strongly Disagree.
⁵ Selected: Excellent + Good/Unselected: Average/ Poor.
⁶ Selected: Definitely yes + Probably yes/Unselected: Uncertain/ Probably no/ Definitely no.
⁷ Selected: Above average quality + Average quality/Unselected: Below average quality/ I don't know.
⁸ Selected: Yes with some reservations + Yes without reservations/Unselected: No.

Capella Proprietary and Confidential Last updated: 2/5/2008 1:33 PM

CAPPELLA UNIVERSITY		Alumni Outcomes Survey: One Year Out	
HASOP-BIS			
Based on 163 submissions from 214 alumni (76.2% response rate) graduating from 2000Q1 thru 2002Q1.			
DO NOT CITE AFTER 06/30/2008 Questions: Contact AssessmentOffice@capella.edu Generated: 04/28/2008			
	CU %	n ¹	n / N ²
Employed	37.6%	59	157
Full-time	34.2%	54	158
Top job field: 5	97.3%	40	41
Top job class: 4	51.6%	39	76
Found job prior to starting their Capella degree	39.1%	61	156
Reported annual salary of \$50,000 or greater	38.6%	27	70
Indicated Capella education prepared them for current job ³	43.0%	67	154
Do you currently teach in a post-secondary setting?	37.1%	56	151
Top academic Rank: 3	38.0%	37	97
Satisfied with aspects of current job ⁴			
Working conditions	43.6%	68	156
Location	41.8%	64	153
Said their job is Directly or Moderately related to their Capella Degree field	36.1%	56	155
Unemployed	62.4%	98	157
Seeking employment	38.8%	59	152
Seeking employment for 6 months or less	33.8%	53	157
Since earning their Capella University degree, graduates have ⁵			
Changed my career or profession	49.7%	76	153
Assumed increased leadership roles	41.9%	67	160
Upon earning Capella degree, sought professional licensure and/or certification	40.4%	63	156
Ultimately got licensure and/or certified	35.9%	55	153
Have or intended to complete additional degree/certification/course	37.4%	58	155
Indicated their Capella University Degree provided them ⁶			
Recognition and respect from my colleagues	47.4%	74	156
Increased leadership abilities	43.0%	67	154
Career mobility	42.6%	66	155
Expertise in my field of work	41.9%	65	155
Overall, rated Capella above average ⁷	44.8%	69	154
Would return to Capella if they could start over ⁸	39.5%	60	152
Perceive others view their Capella degree as Average quality or better ⁹	43.2%	67	155
Have recommended Capella to family, friends, or acquaintances ¹⁰	35.9%	56	156

¹ CU % represents the total % of total responses for the associated question.
² N = number of respondents described. N = number of people providing a valid response to the item. % = n / N
³ Selected: Exceptionally well + More than adequately + Adequately/Unselected: Less than adequately/Very poorly.
⁴ Selected: Very Satisfied + Satisfied/Unselected: Neutral/ Dissatisfied/ Very Dissatisfied.
⁵ Selected: Strongly Agree + Agree/Unselected: Disagree/ Strongly Disagree.
⁶ Selected: Excellent + Good/Unselected: Average/ Poor.
⁷ Selected: Definitely yes + Probably yes/Unselected: Uncertain/ Probably no/ Definitely no.
⁸ Selected: Above average quality + Average quality/Unselected: Below average quality/ I don't know.
⁹ Selected: Yes with some reservations + Yes without reservations/Unselected: No.

END MATTER

A NOTE ON RANDOM DATA

In survey research, the data collection phase is distinct from the analysis phase. By the time the first submission is received (and preferably well before this), the analyst should have a good feel for what the data will look like:

- How are survey items represented in the data set?
- How are response options coded?
- How are missing values coded?
- How is item branching represented in the raw data?

These are all questions that an analyst should attempt to answer before results come in. While the survey is being fielded, the analyst can start developing reporting programs facilitated by a good set of random data.

The development data set should resemble the final data set in most meaningful ways, including variable names, variable and response option labels, and valid response ranges. The code for the development data set used for this paper is presented in Listing 5 in the Appendix.

AN ALTERNATIVE APPROACH

While it was not discussed in this paper, it is always worth exploring alternate approaches. In the data summarization, for example, a programmer might create a single intermediate data set consisting of binary variables (akin to OLAP cubes) rather than generating separate data sets for each survey item and then merging them. Aggregations could then be made via PROC SUMMARY with grouping variables listed on the CLASS statement. This would have the added benefit of multi-way aggregations, which is what is most often sought.

SOME RESOURCES

Hamilton, J. (2004). *Using New Features in ODS to Create Master/Detail Reports*. SUGI 29, Montréal, Québec, Canada.

<http://www.excursive.com/sas/p229-29.pdf>

Koopmann, R. Jr. (2007). *Experimenting with the ODS DATA Step Object (Part I)*. PNWSUG 2007, Seattle, WA.

http://www.pnwsug.org/Conference_2007/Proceedings/Papers/RichardKoopmann-ExperimentingwiththeODSDATAStepObject.pdf

Koopmann, R. Jr. (2007). *ODS DATA Step Object 101: Tip Sheet*

http://www.sascommunity.org/mwiki/images/b/be/ODS_DATA_Step_Object_Method_101-Tip_Sheet.pdf

Ladan, A. I. (2006). *The Absolute Nitty-griddy of ODS Layout: Part I*. SESUG 2006, Atlanta, GA.

http://www8.sas.com/scholars/Proceedings/2006/Posters/P002_06.PDF

O'Connor, D. (2003). *Methods for the ODS DATA Step Object*.



<http://support.sas.com/rnd/base/topics/datastep/dsobject>

O'Connor, D. (2003). *Next Generation DATA _NULL_ Report Writing Using ODS OO Features*. SUGI 28, Seattle, WA.

<http://www2.sas.com/proceedings/sugi28/022-28.pdf>

CONTACT INFORMATION

Questions and comments are always welcome, so feel free to contact the author:

Name	Richard Koopmann Jr.		
Company	Capella University		
E-mail	richard [dot] koopmann [at] capella [dot] edu		
sasCommunity	User:RichardK		

COLOPHON

This document was typeset in the (MiKTeX) \LaTeX environment using the following document class, packages, and editor:

sugconf A custom document class for SAS User Group Conference proceedings.

listings Include source code from the source files directly into \LaTeX document... can Word do that?

textcomp \LaTeX automatically converts straight quotes to curly quotes, which SAS does not like. This package reverses this effect to make included code more portable through copy-paste-run.

graphicx Allows graphics to be included from a variety of file formats, including PDF.

hyperref Fine-tune PDF (document properties, link properties, etc.).

PSPad An extremely powerful freeware text editor supporting over 100 programming and scripting languages along with syntax highlighting (for Windows only).

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.

APPENDIX

Listing 4: Simple reports for SASHELP.CLASS

```
proc format;
  value bmi
    /* http://www.cdc.gov/nccdphp/dnpa/bmi/adult_BMI/about_adult_BMI.htm */
    low  < 18.5 = 'underweight'
```



```

    18.5 -< 25.0 ='normal'
    25.0 -< 30.0 ='overweight'
    30.0 - high ='obese';
value $pronoun
  'F' = 'she' 'M' = 'he';
value $sex
  'M' = 'Male' 'F' = 'Female';
run;

*****
**** TRADITIONAL DATA NULL REPORTING ****
*****
data _null_;
  file 'data_null.txt';
  set sashelp.class end=last;
  bmi = weight / (height*height) * 703;
  note = catx(' ', 'If', name, 'were an adult,'
    , put(sex, $pronoun.), 'would be', put(bmi,bmi.));
  put name
    / '-----'
    / 'Gender'      @15 sex $sex.
    / 'Age'        @15 age 3.
    / 'Height (in)' @15 height 5.1
    / 'Weight (lb)' @15 weight 5.1
    / 'BMI'        @15 bmi 5.2
    / '-----'
    / note
    ///;
run;

*****
**** TRADITIONAL ODS REPORTING WITH REPORT ****
*****
ods pdf file='ODS_Report.pdf';
proc report data=sashelp.class nowd;

column name sex age height weight bmi note;

define name / display;
define sex / display format=$sex.;
define age / analysis sum;
define height / analysis sum format=6.2;
define weight / analysis sum format=6.1;
define bmi / computed format=6.2;
define note / computed format=bmi. 'If adult...';

compute bmi;
  bmi = weight.sum / (height.sum * height.sum) * 703;
endcomp;

compute note;
  note = bmi;
endcomp;
run;
ods pdf close;

*****
**** TRADITIONAL ODS REPORTING WITH TABULATE ****
*****
data class; set sashelp.class; bmi=weight/(height*height)*703;
ods pdf file='ODS_Tabulate.pdf' style=journal;
proc tabulate data=class;
  class name sex;
  format sex $sex.;
  var age height weight bmi;
  keylabel n=' ' sum=' ';
  table name=' '
    * (sex=' '*f=3. age*f=3. height*f=5.1 weight*f=5.1 bmi*f=5.2 bmi='If adult'*f=bmi.)
    , all=' ';
run;
ods pdf close;

*****
**** ODS DATA STEP OBJECT ****
*****
ods pdf file='ODS_Data_Step_Object.pdf';
ods proclabel='ODS DATA Step Object Example Report';
data _null_;
  set sashelp.class end=last;

```

```

if _n_ = 1 then declare odsout obj();
/* http://www.cdc.gov/nccdphp/dnpa/bmi/adult_BMI/about_adult_BMI.htm */
bmi = weight / (height*height) * 703;

obj.layout_gridded ( width: '4in', height: '2.0in' );
obj.format_text( text: name, overrides: 'font=(helvetica,14 pt,bold)');
obj.table_start( label: name );

obj.row_start();
obj.cell_start( inhibit: 'BLR' );
obj.format_text( text: 'Gender', overrides: 'font_face=arial' );
obj.cell_end();
obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
obj.format_text( text: put(sex, $sex.), overrides: 'font_face=arial' );
obj.cell_end();
obj.row_end();

obj.row_start();
obj.cell_start( inhibit: 'BLR' );
obj.format_text( text: 'Age', overrides: 'font_face=arial' );
obj.cell_end();
obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
obj.format_text( text: put(age, 3. ), overrides: 'font_face=arial');
obj.cell_end();
obj.row_end();

obj.row_start();
obj.cell_start( inhibit: 'BLR' );
obj.format_text( text: 'Height (in)', overrides: 'font_face=arial' );
obj.cell_end();
obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
obj.format_text( text: put(height,5.1), overrides: 'font_face=arial' );
obj.cell_end();
obj.row_end();

obj.row_start();
obj.cell_start( inhibit: 'BLR' );
obj.format_text( text: 'Weight (lb)', overrides: 'font_face=arial' );
obj.cell_end();
obj.cell_start( inhibit: 'BLR', overrides: 'just=right' );
obj.format_text( text: put(weight,5.1), overrides: 'font_face=arial' );
obj.cell_end();
obj.row_end();

obj.row_start();
obj.cell_start( inhibit: 'LR' );
obj.format_text( text: 'BMI', overrides: 'font_face=arial' );
obj.cell_end();
obj.cell_start( inhibit: 'LR', overrides: 'just=right' );
obj.format_text( text: put(bmi,5.2), overrides: 'font_face=arial' );
obj.cell_end();
obj.row_end();

obj.table_end();
obj.format_text( text: catx(' ', 'If', name, 'were an adult,'
, put(sex,$pronoun.), 'would be', put(bmi,bmi.) || '.')
, overrides: 'font_face=arial');
obj.layout_end();
*if not last then obj.page(); *force a page break;
run;
ods pdf close;

```

Listing 5: Generating the Development Data Set

```

data aos(drop=q qsub);
format id z4. newschool $5. degree $3. specialization $7.;
do id = 1 to 4000;
  graddate = intnx('month','01jan2000'd, ceil(ranuni(1)*4*6));
  newschool = put(ranuni(2), rand_school.); *assign school;
  degree = put(ranuni(3), rand_degree.); *assign degree;
  specialization=catx(' ', 'Spec', put(10+ranuni(4)*5, z2.)); *assign specialization;
  didSubmit = (ranuni(5) <= 0.75); *response rate;
  q=.;
  array sub (33) _temporary_ (1,2,1,2,1,2,2,1,1,4,1,1,2,1,2,4,6,6,1,1,1,4,1,1,15,3,1,1,1,1,1,1);
  if didSubmit then
  do q = 1 to 33;
    do qsub = 1 to sub(q);
      if sub(q) = 1 then item = cats('q', put(q,3.));
      else item = cats('q', put(q,3.), '_', put(qsub,3.));
      if q = 11 then value = ceil(ranuni(6) * 11);
      else value = ceil(ranuni(6) * 5);
      if ranuni(7) < 0.05 then value = .; *randomly set 5% to missing;
      output;
    end;
  end;
end;

```

```
        end;
    end;
    else do;
        item = 'q1';
        value = .;
        output;
    end;
end;
format item $6. graddate yyq6. newschool degree $4.;
run;

proc transpose data=aos out=aos(drop=_NAME_);
    by id;
    id item;
    var value;
    copy newschool degree specialization didsubmit graddate;
run;

data aos(label='Fake AOS data');
    set aos;
    format q4_1 q5 q7_1 q8 q15_1 options.;
    by id;
    if first.id;
```
