Paper 132-28

Using Formats and Other Techniques to Complete PROC REPORT Tables

David D. Chapman, US Census Bureau, Washington, DC

ABSTRACT

Calculating the totals correctly is not the end of a PROC REPORT project, it is the beginning of the end. A key part of completing a table is to decide how to present the numbers. This paper discusses two parts of a table: cells and footnotes. Displaying cells with only numbers, with only symbols, and with both numbers and symbols is discussed separately. The use of standard SAS formats and user defined formats to display standard numbers and numbers truncated, rounded, and expressed to two significant digits is demonstrated. Compute statements are used to illustrate how to display table results to two significant digits and to add warning flags. When numbers are not enough, cell notes are needed. Cell notes are numbers or symbols that replace or supplement numbers in a cell. Examples of cell notes are symbols to represent missing data (NA), and values suppressed to prevent disclosure (D). Examples are given of the use of the COMPUTE statement to create conditional cell notes to either replace a number with a cell note or to add an accompanying symbol to the number. A brief discussion of footnotes along and an example of PROC REPORT code showing how to display footnotes is given.

INTRODUCTION

While there are guidelines and styles for tables, organization do them differently. They are different because of the needs of the user, specialized content of the data they display, preferences of individuals developing them, and ultimately their purpose. Tables are usually of three types: Official documents, working tables, and presentation tables. Tables as official documents record results of a surveys, and experiments. Often they have an associated obligation to preserve privacy, warn users of potential problems, or explain when may be missing. Tables as reports of new information have different objectives of wanting the reader to compare data or to see a pattern. How the numbers are presented influence this. Often to make the patterns more recognizable, numbers are rounded, truncated, or expressed to two or three significant digits.

PARTS OF A TABLE

A table is compose of five basic parts: (1) Title, (2) Boxhead (columns), (3) Stub (rows), (3) Field (cells), and (5) Footnote. Each has a separate and distinct purpose.

Heading:

STUB-HEAD	Boxhead			
	Spanner			Column
	Column	Column	Column	
Row		CELL		

Footnote:

field

The heading consists of the table number, and headnote. The table number is an identifying method; the title is a brief statement of the nature, classification, and time reference of the information in the table and serves as a reference guide.

The boxhead describes the data contained in the individual column heads or captions. It may also contain banner and spanner heads that classify or qualify a caption spreading across two or more individual columns or lower level spanners. The individual column head gives a descriptive title to the individual column to which it refers. A banner head is a spanner which extends across the entire boxhead of the table.

The stub is almost always the first column on the left and contains the line captions and associated center heads and stubheads. The stub describes table data and makes clear the relationship among rows or lines. The line caption describes data on a particular line or row of cells. The stubhead describes the stub listing as a whole in terms of classifications presented. A block or center head may also be included in the stub.

The field is the depository of statistical information in the table and the area below the column headings and to the right of the stub. The field is made up of columns, rows, and cells The cell is the basic element in presentation in the table. Cells can contain numbers, words, symbols, or a combination of the three. A row or line is a group of cells in a horizontal line; a column is a group of cells in a vertical line.

The footnote is information put at the bottom of the table. It can be either a specific footnote, source note, or general note. A specific footnote explains a certain cell entry, column heading, line caption, or word. A source note explains where the information in the table comes from. A general note qualifies, describes, or explains the table as a whole.

This paper concerns the cells that make up the table's field and with the footnotes that explain notations in cells to users. There are a number of excellent references on the preparation of statistical tables (Census Bureau(1949), EIA (1970)).

THE PROBLEM AND A SOLUTION

The problem is that just tabulating the data is not enough. Missing values need to be dealt with, numbers need to formatted or rounded, strange values need to be explained, and special rules need to be applied. All these are non mathematical; but, all are necessary. The example below illustrates both the problem and a solution.

Problem

The problem below is missing values and explanation why there is not ratio for the South (S) region when data exists for both revenue and expense. Some think it is a cluttered and not easily read.

Region	firms	revenue	expense	Ratio
NE	200	1752	1293	1.35
MW	23	375	4291	0.09
S	150	.	2950	.
W	2	3156	0	.
Total	375	5283	8534	0.62

Solution
The solution is to change how the data is displayed and add comments. Revenue and Expense have been rounded to thousands, missing values identified by a symbol, value rounded to zero identified, and data associated with 2 or fewer firms suppressed. The meaning of the cell notes are explained in a specific footnote at the bottom of the table.

Region	firms	REVENUE (000s)	EXPENSE (000s)	RATIO
NE	200	2	1	1.35
MW	23	(Z)	4	0.09
S	150	(M)	3	(M)
W	2	(S)	(S)	(NM)
	375	5	9	0.62

(Z)-Rounded to Zero, (M)-Missing,
(NM)-Not Meaningful

The solution to many of the normal problems of preparing and presenting tables is to use a combination of formats (either SAS defined or user defined) and the PROC REPORT compute statement. Initial PROC REPORT tables present cell values that are usually numbers. The final table many times requires a table that has a number, a number and a symbol, or just symbol.

CELL NOTES

The cell is the basic object of most tables. A cell can contain:

number only	$123,000
number and unit-indicator	345,678 GB
number and symbol	2,234 r
symbol only	(D)

While the purpose of a statistical table is to display numeric data in table cells, special situations require comments be added to the data or in place of the data. These comments are called cell notes are needed because data is missing, has been change in some nonstandard way, are not meaningful, or the sampling error or imputation rate is too high.

CELLS WITH ONLY NUMBERS

Most cells in statistical tables contain only numbers. Changes to calculated values occur because you want to present the data more meaningfully by simplifying the numbers thru truncation, rounding, or significant digits. Truncation refers to either dropping some of the right most digits (123897 becomes 123) or replacing the dropped numbers with zeros; rounding refers to changing the right most digit displayed depending on whether the first drop digit was greater than or equal to 5 or less than 5 (123897 rounded to thousands becomes 124000); and significant digits refers to keeping only the first two or three digits regardless of the position in the number and changing the others to zero (123897 to two significant digits is 120000).

SAS Formats
SAS supplied formats provide an easy and simple way to increase the readability and understandability of data presented in a PROC REPORT table. Use of SAS defined formats allow the presentation of understandable standard numbers. Common SAS supplied defined formats used in tables are the COMMA and DOLLAR formats. European versions for these also exist (DOLLARXw.d and COMMANw.d). SAS comes with a wide variety of formats. They are described in excruciating detail in the SAS Language Reference Manual. Some common standard SAS formats are given below and show the same number displayed in different formats.

Display of Number -1234.56789 Using Different Formats		
Format	Description	Display
12.2	Standard numeric	-12334.57
COMMA12.2	Write numeric values with commas and decimal points	-1,234.57
COMMAX12.2	Writes numeric values with commas and periods.	-1.234,57
DOLLARr12.2	Write numeric values with dollar signs, commas, and decimal points	$-1,234.57
NEGPAREN12.2	Write negative numeric values in parentheses	(1,234.57)
E12.	Scientific notation	-1.23457E+03

Formats allow the programmer to customize a table to different audiences and make numbers more readable (dollar signs, comma separators and parentheses for negative numbers).

Truncating and Rounding
Rounding is usually expressed in terms such as “rounding to two decimal places” or “rounding to thousands” . Truncation refers to either dropping left most digits of a number or replacing the digits with zeros. In general formats can not be used for rounding; however, they can be used to round the decimal parts of a number. Formats don’t work for the integer part. If you need to round results in PROC REPORT table, the solution is use a compute statement. This is illustrated in the table below.

```
proc report data=example nowd;
column   row original truncate format;
define   row      / order;
define   original / display         'Original';
define   truncate / display  format=comma18.0
                                    'Truncated';
define   format   / display         'Rounded';
compute  truncate ;
         truncate = int (truncate /  100) *100;
endcomp;

compute format;
        format = int (( format + 50)/ 100) *100;
endcomp;

run;
```

The table produced by the PROC REPORT code above is given below.

row	Original	Truncated	Rounded
1	99123	99,100	99100
2	99133	99,100	99100
3	99143	99,100	99100
4	99153	99,100	99200
5	99163	99,100	99200
6	99173	99,100	99200
7	99183	99,100	99200
8	99193	99,100	99200
9	99203	99,200	99200

Truncation can also be done with a user defined format based on a picture statement. A format to truncate numbers to thousands is given below;

```
proc format;
picture trunc
0='0'
1-999='Z'
1000-999999='009'                          (mult=.001)
1000000-999999999='000,009'                (mult=.001)
1000000000-999999999999='000,000,009'      (mult=.001);
run;
```

Effect of Using the "trunc12." Format	
Original Number	Truncated to thousands using trunc12. format
1.23456789	Z
12.3456789	Z
123.456789	Z
1234.56789	1
12345.6789	12
123456.789	123
1234567.89	1,234

SIGNIFICANT DIGITS

Some reports are for the record and data must be expressed precisely. The purpose of other reports is to show relationships and patterns in the data. To do this, some table makers argue that data should be shown with only two significant digits. Displaying data with a lot of digits they argue suggests spurious accuracy and makes meaningful comparisons difficult. The example below shows a table with the original data displayed in one row and the data displayed to one, two and three significant digits in the other rows.

Examples of One, Two, and Three Significant Digits			
Original Number	12345.78	1234.578	0.0012345678
One Digit Significant	10000.00	1000.000	0.001
Two Digits Significant	12000.00	1200.000	0.0012
Three Digits Significant	12300.00	1230.000	0.00123

Significant digits refer to the digits of a number from left to right starting with the first nonzero digit. Some examples are given below. Numbers can be expressed in a PROC REPORT table based on significant digits three different ways: user defined format using a value statement, user defined format using picture statement, and a compute statement. A PROC REPORT table showing the three different approaches is given below.

		FORMATS		
GROUP	ORIGINAL	PICTURE	BRUTE FORCE	COMPUTED
1	1,757,239	1,700,000	1,700,000	1,700,000
2	374,824	370,000	370,000	370,000
3	186,945	180,000	180,000	180,000
4	1,921,500	1,900,000	1,900,000	1,900,000
5	394,672	390,000	390,000	390,000
	13,502,022	13,000,000	13,000,000	13,500,000

PROC FORMAT with Picture STATEMENT – This is the most logical approach; but, it is suffers from the problem that you need to substitute the capital letter "O" for the number "0" to display trailing "zeros". This is needed because of a limitation of PROC FORMAT. To use this you need to know the maximum value you will encounter. A user defined format to do this is given below.

```
PROC FORMAT ;
PICTURE   DIGIT
0  = 9
1-9  ='09'
10-99  ='99'
100-999 = '99O'                          (MULT =.1)
1000 - 9999 = '9,9OO'                    (MULT =.01)
10000 - 99999  = '09,OOO'                (MULT =.001)
100000 - 999999 = '09O,OOO'              (MULT =.0001)
1000000 - 9999999 = '0,9OO,OOO'          (MULT =.00001)
10000000 - 99999999 = '09,OOO,OOO'       (MULT =.000001);;
RUN;
```

PICTURE FORMAT Using Value Statement – This approach is based on creating a format using an SAS data with the "cntlin" option. The trick is to write data step code to generate a SAS data set that will be read in the "cntlin" option. The SAS data step code is given below. A portion of the SAS data set produce by that code is given below. It creates a SAS format called "BRUTEw.".

start	end	label	fmtname
970000	979999	970,000	brute
980000	989999	980,000	brute
990000	999999	990,000	brute
1000000	1099999	1,000,000	brute
1100000	1199999	1,100,000	brute
1200000	1299999	1,200,000	brute

This SAS data set can be create use the following code. This file can be used to convert any number from zero to 10^{13} to a number with two significant digits. A new file would be needed if you wanted to expressed number to three significant digits or for bigger numbers..

```
data fmt_brute;
do i     =  0  to 9 ;
   j     =  0 ;
start    =  i  ;
end      =  I ;
y        =  i ;
x        =  put ( y, comma10.);
label    =  x ;
fmtname  = 'brute';
output;
end;

do j     =  1 to 12;
do I     = 10 to 99;
start    = i*10**(j-1) ;
end      = i*10**(j-1) + (10**(j -1)-1 );
y        = i*(10**(j-1) );
x        = put ( y, comma10.);
label    = x;
fmtname ='brute';
output;
end;
end;
run;

proc format  cntlin=fmt_brute;
run;
```

COMPUTE STATEMENT – The PROC REPORT code below actually creates the table. It demonstrates calculating the significant digits in a compute statement. One advantage of the compute statement is that you can express basic rows to two significant digits and the grand total to three significant digits. When you use formats, you are locked in to the same level of precision for a variable through out the table.

```
DATA example;
DO I=1 TO 14;
X=( UNIFORM(0)  ) *10**6;
Y=int( X );
z=int( X );
w=int(x);
OUTPUT;
```

```
OUTPUT;
END;
RUN;
options ls=64;
proc report data=example  nowd  HEADLINE
HEADSKIP  ;
COLUMN  I x ("_FORMATS_ " y z)   wW ;
define  i   / group "GROUP";
DEFINE  X   / ANALYSIS                "ORIGINAL"
                      FORMAT=COMMA10.;
DEFINE Y    / ANALYSIS                "PICTURE"
                      FORMAT=DIGIT10.;
DEFINE Z    / ANALYSIS                "BRUTE FORCE"
                      FORMAT=BRUTE10.;
DEFINE WW   /  COMPUTED               "COMPUTED"
                      FORMAT=COMMA10.;
RBREAK   AFTER / OL SUMMARIZE ;

COMPUTE WW ;
     DIGITS=2;
IF  _BREAK_    NE "_RBREAK_" THEN DO;
     DIGITS=2;
     IF LOG10(X.SUM)>0 THEN
        WW = INT  ( X.SUM /
     (10**(INT(LOG10(X.SUM))-(DIGITS-1))))*
        10**( INT(LOG10(X.SUM))-(DIGITS-1)));
     IF X.SUM < 10 THEN WW=X.SUM;
END;

IF  _BREAK_    EQ "_RBREAK_" THEN DO;
     WW=X.SUM;
     DIGITS=3;
     IF X.SUM < 10 THEN WW=X.SUM;
     IF LOG10(X.SUM)>0 THEN
        WW = INT  (X.SUM /
       (10**( INT(LOG10(X.SUM))-(DIGITS-1))))*
       (10**( INT(LOG10(X.SUM))-(DIGITS-1)));
END;
ENDCOMP;
RUN;
```

This approach takes the logarithm of the number to determine how many digits the number has. The number of significant digits is subtracted from the number of digits to determine how many zeros are needed. The original number is divided by 10 raised to the number of zeros wanted. The integer part of the number is the number of significant digits.

CELL WITH NUMBERS AND CHARACTERS

Two common situations where cells have both numbers and symbols are when the figures displayed have units of measurement (220 Acres) or a conditional flag (267 r) that alerts the reader to some special conditions.

Units of Measurement

When all cells in a column have the same unit of measurement (e.g. Millions) or the same flag (e.g. r - Revised Data), the information is made part of the column or row heading explicitly and referenced using a footnote symbol. Often, a column has cells that contain different units of measurement. An example is table where the rows represent foreign countries and the column is sales expressed in different currencies or the column is disk space expressed in bytes, megabytes(MB), gigabytes(GB), or terabytes(TB).

Some units of measurement can be added to a cell with a picture format. This is particularly true when there is a metric unit of measure. Other types of units of measurement require the use of the PROC REPORT COMPUTE statement.

The user defined format below illustrate how to display a cell with both numbers and symbols. In the example below, disk space in bytes is displayed depending on the value of the cell in either bytes, kilobytes, megabytes, or gigabytes.

```
PROC FORMAT ;
PICTURE  BYTES
LOW - <0 ='.'
 0           - <1             = '0'
 1           - 999            = '009  '
 1000        - 999999         = '009 KB'  (MULT=.001)
 1000000     - 999999999      = '009 MB'  (MULT=.000001)
 1000000000  - 999999999999   = '009 GB'   (MULT=.000000001);
RUN;
```

The compute statement is needed to display cells where you need to convert distance in feet to feet, yards, rods, chains, and miles.

```
proc report data=example nowd;
column  row ("_Distance_"  distance xx );
define row / display;
define distance /analysis  format=comma12.
"(feet)";
define xx /computed "(Recoded)";
compute xx  /char  length=16  ;
if distance.sum =0 then xx =0;
if distance.sum >0 and distance.sum =<3
   then xx= (right(put(int((distance.sum/1)
   *10) /10, comma6.1) || ' feet';
if distance.sum >=3 and distance.sum <16.5
   then xx= (right(put( int((distance.sum/1 )
   *10) /10, comma6.1) || ' yards'));
if distance.sum >=16.5 and distance.sum  <66
   then xx= (right(put( int((distance.sum/16.5)
   *10) /10, comma6.1) || ' rods'));
if distance.sum >=66 and distance.sum  <5280
   then  xx=(right(put(int((distance.sum/66)
   *10)/10, comma6.1) || ' chains'));
if distance.sum >=5280
   then
xx=(right(put(int((distance.sum/5280)
   *10)/10,comma6.1)||' miles');
endcomp;
rbreak after /ol;
run;
```

The table produced by this SAS code is given below. The cells are expressed with different units of measurement.

	Distance	
row	(feet)	(Recoded)
1	.5	0.5 feet
2	1	1.0 feet
3	6	6.0 yards
4	12	12.0 yards
5	62	3.7 rods
6	123	1.8 chains
7	617	9.3 chains
8	6,173	1.1 miles
9	12,345	2.3 miles

More complex compute statements can produce cells with combinations of units of measure such as 7 yards 2.1 feet.

Warning Flag

A common use of symbols in statistical tables is to alert readers to special situations or to warn users of the reliability problems in a cell. Some organizations require a warning flag when the sampling error or imputation rate is high. The cell symbol is conditional on the value of one or more other cells in the row. For impuation rate it could be the value of the number of records in the cell and the number of those records imputed. With conditional cell flags you almost always need to use a COMPUTE statement. An example of this type of PROC REPORT table is given below. In the table below, a flag is displayed if the imputation rate was greater than 50 percent.

```
proc report data=sugi28.virginia_pop nowd   ;
   column  county  pct_imputed  population pop ;
   define    county     / group;
   define    population / analysis
                    format=comma15. "Population";
   define    pct_imputed/ format=percent9.3;
   define    Pop        / computed "Population";
   compute pop /char  length=23;
        if percent_imputed.sum > .5  then
                pop = (right(put(population.sum,
                      comma15.))|| '  !!' ) ;
        else    pop = (right(put(population.sum,
                      comma15.))||'      ' ) ;
   endcomp;
   rbreak after / summarize ol ;
run;
```

This code produces the following table.

COUNTY	Population	
1	9,160	
2	5,762	
3	1,902	
4	1,484	!!
5	7,601	
	25,909	

!! Percent Imputed greater than 50 percent

Warning flags displayed either before or after the number are used to alert the table reader to reliability problems with respect to complete or item nonresponse error, coverage error, or elevated sampling error. These conditional cell notes are similar to conditional notes used when a number is suppressed and only a cell note displayed.

CELLS WITH ONLY CHARACTERS

Symbols are displayed in cells of statistical table to explain the values of the cell. When the same explanation can be given for all cells in the table, or all cells in a row or column of cells, the explanation can be put in the headhote, or the column or row caption. Often however, the symbols are specific to a given cell. When it is, a symbol is used to interpret the cell for the user. The symbol placed in the cell is explained in a specific footnote. These specific cell notes can be either conditional or unconditional cell notes. Unconditional means that the symbol displayed depends only on the value of cell. Missing values and rounding to zero are examples of this. Examples of unconditional symbols are symbols based on the value of the cell such as a missing value, an extremely small value, and units of measurement based on the cell value. When the value of the cell is determined by the value of other associated cells is called a conditional cell note. The dividing line may get a little murky. Examples of unconditional cell notes are notes related to cell quality or reliability, whether they are meaningful, and other qualities.

Commonly Used Cell Notes	
CellNote	Descriptions
NM	Not Meaningful
B	Base less than ___
D	Data withheld to avoid Disclosure
NS	Not Significant
X	Not Applicable
S	Data does not meet publication standards
-	Zero or rounds to zero
NA	Not Available
R	Revised
M	Missing
t	Less than one-half unit of measurement (trace)

A variety of symbols commonly appear in tables. Statistical groups that use many of these symbols in their tables include US Department of Agriculture, Census Bureau, Bureau of Economic Analysis, National Institute of Transportation Statistics, and the Environmental Protection Association.

Missing Values
A common situation is for cells to have missing values. The general solution is often to use a simple user defined format. The format below replaces all missing values with the symbol (M) and most other cells with data having a “dollar12.” format. A simplistic but often effective solution to missing values is to redefine the missing value symbol using the options statement. The default missing value symbol of a number is “.”. The statement below redefines the missing value symbol to an “M” and any table cell with a missing value displays the value “M”.

```
options missing='M';run;
```

The missing value symbol can also be incorporate by creating a simple user defined format. The code below inserts an “(M)” when the value is missing.

True Zero versus Round Zero
An important distinction to many users is whether a cell is missing, reported zero, or rounded to zero. A cell note is commonly used to make the distinction. The cell note “(t)” is used to represent values rounded to zero.

Generic Format
A generic format that unconditionally creates a cell note for missing values, true zero, values rounded to zero, and all other values is given below.

```
PROC FORMAT
        VALUE REVENUE
          .    = '(M)'
           0   =  '0'
     <0 - <.5 = '(t)'
  .5  - high = [dollar12.];
Run;
```

Conditional Cell Notes
Conditional cell notes depend on the values of other cells on the same row. The table below uses conditional cell notes to identify cell with firms considered reliable and a ratio that is not meaningful because the denominator is zero.

County	Firms	Revenue	Expenses	Revenue Expense Ratio
1	27	27,534,999	19,982,298	1.38
2	12	(S)	(S)	(S)
3	32	33,464,667	(Z)	(NM)
4	33	32,705,477	17,814,355	1.84
5	43	33,248,724	15,279,846	2.18

(Z) -- Zero
(S) -- Suppressed because firms less than 15
(NM)-- Not Meaningful

The PROC REPORT code that producers the table is given below.,

```
PROC REPORT DATA = EXAMPLE  NOWD
                SPLIT = "*"  HEADSKIP HEADLINE ;
COLUMN   CTY FIRMS REVENUS EXPENSES R E RATIO  ;
DEFINE   CTY       / group          "County"  ;
DEFINE   Firms     / ANALYSIS   "Firms"  ;
DEFINE   Revenue   / ANALYSIS   "Revenue"      NOPRINT;
DEFINE   Expenses /  ANALYSIS   "Expenses"     NOPRINT;
DEFINE   R         / COMPUTED "Revenue";
DEFINE   E         / COMPUTED "Expenses" ;
DEFINE   ratio     / COMPUTED "Revenue*Expense*RATIO";
COMPUTE  R       / CHAR   LENGTH=13;
            R = LEFT( PUT(REVENUE.SUM , COMMA13.0));
           IF FIRMS.SUM < 15 THEN R = '(S)';
ENDCOMP;
COMPUTE  E   / CHAR   LENGTH=13 ;
            E = LEFT( PUT(EXPENSES.SUM , COMMA13.0));
            IF          EXPENSES.SUM=0 THEN E="(Z)";
            ELSE IF FIRMS.SUM < 15 THEN E = '(S)';
ENDCOMP;

COMPUTE  ratio  /CHAR  LENGTH=23 ;
   RATIO=LEFT(PUT((REVENUE.SUM/EXPENSES.SUM),COMMA7.2));
              IF EXPENSES.SUM=0 THEN   RATIO= "(NM)" ;
              IF FIRMS.SUM < 15    THEN   RATIO=   '(S)';
ENDCOMP;
COMPUTE   AFTER   ;
  LINE '                                  ';
  LINE '(Z)--Zero                         ';
  LINE '(S)--Suppressed because Firms less than 15';
  LINE  '(NM)-- Not Meaningful                ';
```

```
 LINE  '                          ';
ENDCOMP;
RBREAK AFTER / OL ;
RUN;
```

Conditional cell notes allows great flexibility in putting cell notes in tables.

FOOTNOTES

A statement or note inserted at the bottom (foot) of a table is a footnote or source note. Specific footnotes define symbols (S) or numbers (2) appearing usually in one or more cells. For tables that are entirely on one page, specific footnotes go at the bottom of the table. For tables that span multiple pages, a common practice is to put a standard message such as "See footnotes at the end of the table." at the bottom of each page except the last and a complete listing of all footnotes at the bottom of the table on the last page. An example table is given below.

```
 dept      firms   revenue    expenses
_____________________________________________
  9        2351      2823        3016
 10        3086      1424        1823
 11        2082      2269        2945
 12        2060      2879        2300
See Notes at end of table
------------- page break -------------
 dept      firms   revenue    expenses
_____________________________________________
  11       2082      2269        2945
  12       2060      2879        2300
           31481     28610       27662
 (Z) Zero
 (S) Suppressed
```

The code below shows the use of the COMPUTE and LINE statement to add a specific footnote to a PROC REPORT table. The key is the use of the "after" and "_page_ options to the COMPUTE statement. The "after" option executes at the end of the table; The "after _page" executes at the end of each page of the table.

```
proc report data=example  nowd;
column  dept   firms revenue expenses ;
define dept     / group;
define firms    / analysis sum ;
define revenue  / analysis sum  format=bdat7.;
define expenses / analysis sum;
rbreak after /summarize;
compute before;
    hold=' ';
endcomp;
compute after  ;
    LINE @2 "(Z) Zero ";
    LINE @2 "(S) Suppressed";
    LINE @2  "    ";
      Hold='x';
endcomp;
compute after _page_ / left ;
    line "See Notes at end of table ";
endcomp;
run;
```

CONCLUSION

Formats and the PROC REPORT compute statements are excellent tools for add the necessary detail to make tables more readable, more usable, and more understandable.

REFERENCES

Guide to Tabular Presentation, Example of Table Structure, Statistical Abstract of the United State 2000, US Census Bureau, Page ix-x

Ehrenberg, A.S.C. (1977), Rudiments of Numeracy, J. R. Statistical Society A, 140, Part 3, pp 277-297

Enrenberg, A.S.C. Reading a Table: An Example, Applied Statistician (1986) V 35, No 3, pp 237 - 244

Ryder, K. (1986) Numerical Tables: Guidelines For Presentation, Journal of Applied Statistics ,Vol 13, No. 1, 1986, Page 77-78

EIA Publishing Style Guide, Chapter 5, Tables and Graphs, http:/www.eia.doe.gov/neic/pubstyle/ch5_1.htm, 8/12/2002

SAS Institute Inc (1999) PROC FORMAT, Chapter 19, SAS Procedures Guide Version 8, Cary NC

SAS Institute Inc (1999) PROC REPORT, Chapter 32, SAS Procedures Guide Version 8, Cary NC

2001 Statistical Abstract of the United States (http://www.census.gov/prod/2002pubs/01statab/stat-ab01.html (12/12/02)

2001 Statistical Abstract of the United States, Guide to Tabular Presentation, Example of Table Structure, http://www.census.gov/prod/2002pubs/01statab/tabular.pdf (12/12/02)

DISCLAIMER

This paper reports the results of research and analysis undertaken by Census Bureau staff. It has undergone a more limited review by the Census Bureau than its official publications. This report is release to inform interested parties and to encourage discussion.

ACKNOWLEDGMENTS

I would like to acknowledge the help of the NESUG 15 - 2002 Code Clinic team (John Cohen, Elizabeth Axilrod, and Bob Virgil) in coming up with the idea of a one-to-one format to convert number to two significant digits.

CONTACT INFORMATION

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

David D. Chapman, Chief
Frame Development Staff
Economic Statistical Methods and Programing Division
U.S. Census Bureau

Washington, DC 20233-6400
Work Phone: 301-457-4904
Fax: 301-457-1382
Email:david.d.chapman@census.gov