ABSTRACT
This paper is about dynamic column formatting in relation to AE and Conmed listings. By using a macro which can locate the last space before the end of the column and replace it with a 'split' character, this method can automatically solve the problem of a variable splitting in a less than optimal location.

INTRODUCTION:
It is well known that if a variable is longer than the column width specified, Proc Report will automatically find the last space before the end of the column, split the variable at that spot and flow the rest of the variable to the next line. In this case, it will only break the line between words instead of within a word (assuming the column width is at least as wide as the longest individual word).

This is not the case, however, when the split character specified by proc report is contained within the variable text. Under this circumstance Proc Report will generally not identify the most appropriate line breaking point. The result is that the column will be filled to its length and the remaining characters will be put in a new row. Most likely a word will be split into two, which makes the word more difficult to recognize.

This situation is quite common when we deal with AE and Conmed listings. Since most of the time we report Body System and Preferred Term in one column, we need to concatenate these two variables with a split character in between which leads to the failure of the automatic splitting. For example, the summary might look like this if we don’t use the dynamic column breaking macro:

The solution proposed here is to use the following macro which takes two parameters. The first parameter is the variable which needs to be formatted. The second parameter is the width of the column in Proc Report. The value of the first input variable will be overwritten by the return value.
%macro Break(var, len);
  strlgh = length(&var);
  index = 1; count=1;
  do while(index le strlgh);
    *********************************************************
    *****Search for the next segment if find \\
    *********************************************************
    if substr(&var, index, 1) eq '\' then count = 1;
    *********************************************************
    *****Move down the string by increment count**************
    *********************************************************
    if count lt &len then do;
      savechar = substr(&var, index, 1);
      count = count + 1;
      end;
    else if count eq &len and substr(&var, index, 1) eq ' ' then do;
      savechar=substr(&var, index, 1);
      temp = substr(&var, 1, index-1)||'\'||substr(&var, index+1);
      &var= temp;
      index = index + 3; **** adding this one is optional;
      count = 1;
      end;
    else if count eq &len then do;
      svcnt=count;
      count = 1;
      ************************************************************
      *****Moving backward to search for separators by decrement index**
      **************************************************************
      do while(substr(&var, index, 1) not in(' ',' ',' ','(',')','-') and index gt 0);
        index = index - 1;
        savechar=substr(&var, index, 1);
        end;
      else if substr(&var, index, 1) eq ' ' then do;
        temp = substr(&var, 1, index)||'\'||substr(&var, index+1);
        &var= temp;
        index = index + 3; **** adding this one is optional;
        end;
      else if substr(&var, index, 1) eq '\' then do;
        temp = substr(&var, 1, index)||'\'||substr(&var, index+1);
        &var= temp;
        index = index + 3; **** adding this one is optional;
        end;
      else if substr(&var, index, 1) eq ' ' then do;
        temp = substr(&var, 1, index-1)||'\'||substr(&var, index+1);
        &var= temp;
        index = index + 3; **** adding this one is optional;
        end;
      else index = strlgh;
      end;
    end;
    *********************************************************
    *****Move along the string by increment index**************
    *********************************************************
    index = index + 1;
  end;
%
*End prefverb alignment logic**
As can be seen from the code above, the functionality of the macro is to identify certain characters (space, “,” and “\”, etc) to be considered as ‘appropriate’ points to break to a new line (by inserting a split character where needed). The macro relies primarily on two variables to implement this algorithm. The variable index represents the location of the character to be examined. Another variable count represents the width of the search interval. The macro searches the characters backward within the first search interval for the first appropriate breaking point, and substitutes it with a split character. A new search then starts within the next search interval (beginning at the location of the last substitution). The process is repeated until the end of the string is reached. By using the macro, our new listing will look like this:

<table>
<thead>
<tr>
<th>Subject Site Number</th>
<th>Age/ Treatment</th>
<th>Preferred Term [1] (Verbatim)</th>
<th>Start Date</th>
<th>Stop Date</th>
<th>Severity</th>
<th>Relationship to Study</th>
<th>Drug</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>x/M A</td>
<td>POSTURAL ORTHOSTATIC TACHYCARDIA SYNDROME (ORTHOSTATIC TACHYCARDIA)</td>
<td>DDMMYY/xx:xx</td>
<td>DDMMYY/xx:xx</td>
<td>INTERMITTENT</td>
<td>MILD</td>
<td>NONE</td>
</tr>
<tr>
<td>002</td>
<td>x/M B</td>
<td>BLOOD CREATINE PHOSPHOKINASE INCREASED (INCREASED CPK)</td>
<td>DDMMYY/xx:xx</td>
<td>DDMMYY/xx:xx</td>
<td>ONE EPISODE</td>
<td>MILD</td>
<td>NONE</td>
</tr>
</tbody>
</table>


Cross-reference: CRF Page xx
Program: l_ae.sas     Version: DDMMYY xx:xx

CONCLUSION

By using this macro we are able to dynamically determine the most appropriate breaking point in a multi-line summary table (e.g. presenting Preferred Term and Verbatim within the same variable). This relieves programmers from extensive manual formatting in tables of this type.

CONTACT INFORMATION

Fan Ping               Richard Schiefelbein
Analysis Programmer    Director of Early Development Services
PRA International      PRA International
9755 Ridge Drive       9755 Ridge Drive
Lenexa, KS 66219       Lenexa, KS 66219
913-410-2012           913-410-2331
www.prainternational.com www.prainternational.com