Split Street Address Information Using Do Loop and Array Processing

Winston Lai, Commercial Credit Corporation, Baltimore, MD

ABSTRACT

Many data entry systems for business applications allow the user to enter street address information using any convenient format. General address variables such as “LINE1” and “LINE2” are very common. Although this method simplifies data entry, it produces street address data that are difficult to use for matching, tracking, and solicitation projects. This paper demonstrates how to use DO loop and ARRAY processing in Base SAS® Software to split a free format street address into basic components of street name, building number, street direction and apartment number. The same logic can be used to split a person’s name or city-state-zip combination into components.

INTRODUCTION

Many data entry systems for business applications allow the user to enter street address information using any convenient format. General address variables such as “LINE1” and “LINE2” are very common. Although this method simplifies data entry, it could be considered to be a design flaw based on contemporary standards because it is not easy to identify a component such as street name without some further processing of the data. Even when there is a separate variable created to hold data for each component of the address, problems will arise when more than one element is squeezed into one variable due to data entry errors. The outcome in either case is that free format addresses are very difficult to use in business applications such as tracking, matching, and solicitation projects which require separate address components for street name, building number, street direction, and apartment number. Sometimes both the post office box and the street name and building number exist in the free format street address and the business application requires selection of only the post office box information. The situation is further complicated when post office box can be expressed in different ways such as POBOX, P.O.BOX, BOX, B0X, and POB, and when there is no space before the box number as in BOX234. Street directions are also a challenge. SOUTH could be presented as SOUTH, SO, and S; letter E could be a street direction or an apartment number.

Fortunately, DO loop and ARRAY processing in Base SAS® Software provide powerful tools in tackling such complicated situations. By working on an artificial data set, this paper demonstrates how to use these two SAS tools to split street address as a whole into separate components.

SOLUTION: AN EXAMPLE

The following table gives a list of street address (ADDR column) that need to be split into different components. The goal is to separate out building number, street name, street direction, and apartment number information, as listed in columns BLDG, STNM, STDIR, and APT respectively. Note that for this example when the address contains post office box information, PO BOX is used as the real street name, and the post office box number as the real building number.

<table>
<thead>
<tr>
<th>ADDR</th>
<th>BLDG</th>
<th>STNM</th>
<th>STDIR</th>
<th>APT</th>
</tr>
</thead>
<tbody>
<tr>
<td>BOX234</td>
<td>123</td>
<td>BANKER ST</td>
<td></td>
<td>PO BOX</td>
</tr>
<tr>
<td>RT 1 P O B 12D 3 N MAIN 12D</td>
<td>12D</td>
<td>PO BOX</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>PO BOX 11 RT 30 EAST 11</td>
<td>11</td>
<td>PO BOX</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>34 E U S HWY 30 34</td>
<td>34</td>
<td>U S HWY 30</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>132 CO RD NW 132</td>
<td>132</td>
<td>CO RD</td>
<td>NW</td>
<td></td>
</tr>
<tr>
<td>APT E 23 N WEST ST 23</td>
<td>23</td>
<td>WEST ST</td>
<td>N E</td>
<td></td>
</tr>
<tr>
<td>246 13TH ST APT 3 246</td>
<td>246</td>
<td>13TH ST</td>
<td>S 3</td>
<td></td>
</tr>
<tr>
<td>RURAL RTE 6 130 N E 130</td>
<td>130</td>
<td>RURAL RTE 6</td>
<td>NE</td>
<td></td>
</tr>
</tbody>
</table>

The following SAS code that accomplishes the “intelligent” task may look quite complicated at a glance, the underlying strategy is quite simple:

1. Use DO loop and SCAN function to scan through ADDR and store each word to an array element.
2. As scanning through ADDR, evaluate whether the information is a building number, street direction, or apartment number. If this is true, pass the value to BLDG, STDIR, or APT accordingly and set the array element to blank.
3. Concatenate all array elements (only non-blank ones count) together to get the street name.
4. Deal with “PO BOX” case specifically by creating an indicator POBOX.
5. Build in detailed provisions to handle specific situations (e.g., BOX234) as needed by the data, which is case-dependent.

Let us first read through the SAS code to see how this strategy is implemented:

DATA ADDRESS;
SET ADDRESS;
ATTRIB  BLDG    LENGTH = $6
STNM    LENGTH = $30
STDIR   LENGTH = $2
APT     LENGTH = $4;
LENGTH A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 $15;
ARRAY A(10) A1-A10; /* require a prelim.
run to det. the # of elements needed */
DROP  A1-A10;
DROP  I P T APT_IND POBOX;
DO I=1 TO 10;
A(I)=SCAN(ADDR, I, ' ');
IF A(I)=' ' THEN GOTO OUT1;
IF I=1 THEN DO;
***********************************;
* Number appearing at 1st position;*
* is the first option for bldg #. *
***********************************;
IF COMPRESS(SUBSTR(A(1, 1,1), '1234567890')='1234567890')="'
THEN DO;
BLDG = A(1);
IF I=1 THEN DO;
******************************************************************************;
* Number appearing at 1st position*;
* is the first option for bldg #. *;
******************************************************************************;
END;

Artificial Sample Data
IF A(1) IN ('EAST', 'EST', 'SOUTH', 'WEST', 'NORTH', 'E', 'S', 'SO', 'W', 'N', 'SW', 'SE', 'NW', 'NE')
THEN LINK DIR;
END;

IF I>1 THEN DO;
    ****************************************
    * when number appears at later pos., * 
    * it usually has a variety of meanings* 
    * which can be determined by the * 
    * preceding word.                     *
    ****************************************
    IF BLDG = ' ' THEN DO;
        IF COMPRESS(SUBSTR(A(I), 1,1),
            '1234567890') = ' ' 
            AND A(I-1) NOT IN 
            ('RT','RTE','ROUTE','HWY', 
            'RD','R','RR', 'RD','SR', 
            'APT','RM','ROOM') THEN DO;
            BLDG = A(I);
            A(I)=' ';
            END;
        END;
    IF A(I) IN ('EAST', 'EST', 'SOUTH', 'WEST', 'NORTH', 'E','S','SO', 'W','N', 
    'SW', 'SE', 'NW', 'NE') 
    AND A(I-1) NOT IN ('APT') 
    AND STDIR = ' ' THEN LINK DIR;

    IF A(I) IN ('E','S','W','N')
    AND A(I-1) = ' ' THEN DO;
        IF TRIM(STDIR)||TRIM(A(I)) 
        IN ('SW', 'SE', 'NW', 'NE') 
        THEN DO;
            STDIR = TRIM(STDIR)||TRIM(A(I));
            A(I) = ' ';
            END;
        END;
    END;
    END;

    /* identify APT case */
    IF A(I) = 'APT' THEN DO;
        APT_IND=1;
        T = I;
        END;

    /* identify "PO BOX" case */
    IF COMPRESS(A(I), '1234567890') IN 
    ('BOX', 'POBOX', 'POB', 'POX', 
    'BAX', 'BOX', 'BFX', 'BIX', 'OBOX', 
    'BK', 'P.O.BOX', 'BKO') THEN DO;
        PBOX = 1;
        P = I;
        END;
    END;

    OUT1:
    IF APT_IND = 1 THEN DO;
        APT = A(T+1);
        A(T) = ' ';
        A(T+1) = ' ';
    END;

 This code is adapted from a real-life program dealing with more sophisticated situation than the example presented here, so it may look more complicated than needed. The purpose of this over-complicated version is to show how to adapt the basic logic to different situations by building detailed provisions into the strategic skeleton. For example, provisions are made to identify different expressions for “ROUTE” so that the following number would not be treated incorrectly as a building number. Provisions also are made to concatenate “N E” into “NE”, and not to use E as the street direction when it is preceded by APT. Of course these examples are very case-specific and depend on the nature of the data in application. You may not have to deal with such complications at all. However, you may have to develop additional code to create a separate PBOX variable instead of using it as building number, or to cope with more complications in building/apartment number.

This paper uses the street address as an example. The same logic can apply to the splitting of a person’s name, and city-state-zip combination.

REFERENCES


ACKNOWLEDGMENTS

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

CONTACT INFORMATION

Winston Lai
Commercial Credit Corporation
300 St. Paul Place, BSP07D
Baltimore, MD 21202
email: laiw@primerica.com