USE OF A DATE MACRO ALGORITHM TO ESTABLISH A MOVING BUT FIXED LENGTH DATE RANGE
Joseph Pasquale
Carpenter Technology Corporation
Reading Pennsylvania

I will bet that YOU, like every SAS user, have a calendar on the wall near your workstation. Some of you, like me, might even have last years calendar still hanging there. We all find ourselves using these calendars to determine the starting and ending dates for a new fixed length date range every time we run our programs.

A typical data step looks like this ...

```sas
%LET START DATE = %SYSFUNC(DAYNAMES(4)) - 28 + %SYSFUNC(DAYNAMES(4)) - 7;
%LET END DATE = %LET START DATE - %SYSFUNC(WEEKDAY("27JUL95")) - 2;
```

SAS dates are really numbers that have been formatted to make them look like dates that the user can recognize. Since dates are numbers, they can be used in formulas to calculate the starting and ending dates of a range. In this example, the starting and ending dates for the four week period are Mondays.

The formula on line 4 calculates the ending date (ED) for the range. The SYSDATE value and the WEEKDAY function are used to establish the current day of the week and how far to count back to determine what the date is for the Monday of last week. Sounds complicated? Just run through the math assuming that today is Thursday July 27, 1995:

```sas
ED=27JUL95D - (7+WEEKDAY("27JUL95")) - 2
```

27JUL95 (a Thursday) is the 5th day of the week. Plugging in the numbers our formula looks like this:

- `ED=27JUL95D - (7 + 5 - 2)`
- `=27JUL95D - 10 days` or `=17JUL95` (the previous week's Monday)

%LET will put the formulas for calculating the start date (SD) and end date (ED) into memory and pull them out where needed elsewhere in the program.

The data step and PROC PRINT sections need to be modified to utilize the values stored by %LET.

Here is how the data step (line 12) will look ...

```sas
IF REPORTED LE '17JUL95':
RUN;
```

If we attempt to use the MACRO variables SD and ED in our title statement, SAS will simply insert the formulas. The DATA _NULL_ section takes the calculated dates (formula results) and saves them to new MACRO variables (FD and LD) by using CALL SYMPUT ....
Original date references in the title line are replaced with &FD and &LD.

Here is how the PROC PRINT section (line 27) will look...

```
DATA _null_;
CALL SYMPUT('FD',PUT(STD,DATE.));
CALL SYMPUT('LD',PUT(STD,DATE.));
```

Note that double quotes are used on line 27.

All the end user has to do is submit the program. No changes are needed because dates will automatically be calculated by the program. The program is so automatic that the end user can even have the computer room operator run them as part of the production stream.