

William R. Gjertsen, SAS Institute Inc.

ABSTRACT

SAS is widely used in business for computer performance tuning, market research, product planning, operations research and MIS reporting as well as for standard statistical applications. Three examples are presented. The second example uses the new ARRAY and DO...END capabilities of SAS 79.

The major point is that any scatter plot of 10,000 observations will be equally unilluminating. If however we first group CHANNEL 3 BUSY into 10% percentage classes 0-10%,...90%-100% and then do schematic plots of CPU BUSY for each 10% group, Figure 2, a very coherent plot indicative of a possible bottleneck in CPU BUSY whenever CHANNEL 3 BUSY is greater than 70% emerges. The simple SAS program which groups the data and produces the side-by-side box plots with PROC SPLOT is as follows:

EXAMPLE 1: USING BOX PLOTS FOR CHANNEL TUNING

Using schematic or box plots¹ is an excellent way to view large batches of univariate or bivariate data (with side-by-side box plots) in an exploratory manner. An example is shown which illustrates its value in a channel tuning application². Figure 1 illustrates what happens when we plot a week's worth of bivariate 5-second interval data. The data was extracted by a COMTEN 8016 hardware monitor and further condensed into a stored SAS data set.

```
DATA; SET SASSPACE.S410 (KEEP=C9 C11);
IF C11=100 THEN C11=99.9;
C11=10*INT(C11/10); *GROUP C11=CH3BUSY;
*INTO 10% CLASSES;
PROC SORT; BY C11;
PROC SPLOT; CLASSES C11; VAR C9;
TITLE SPLOT OF CPU BUSY VS CHANNEL 3 BUSY;
```

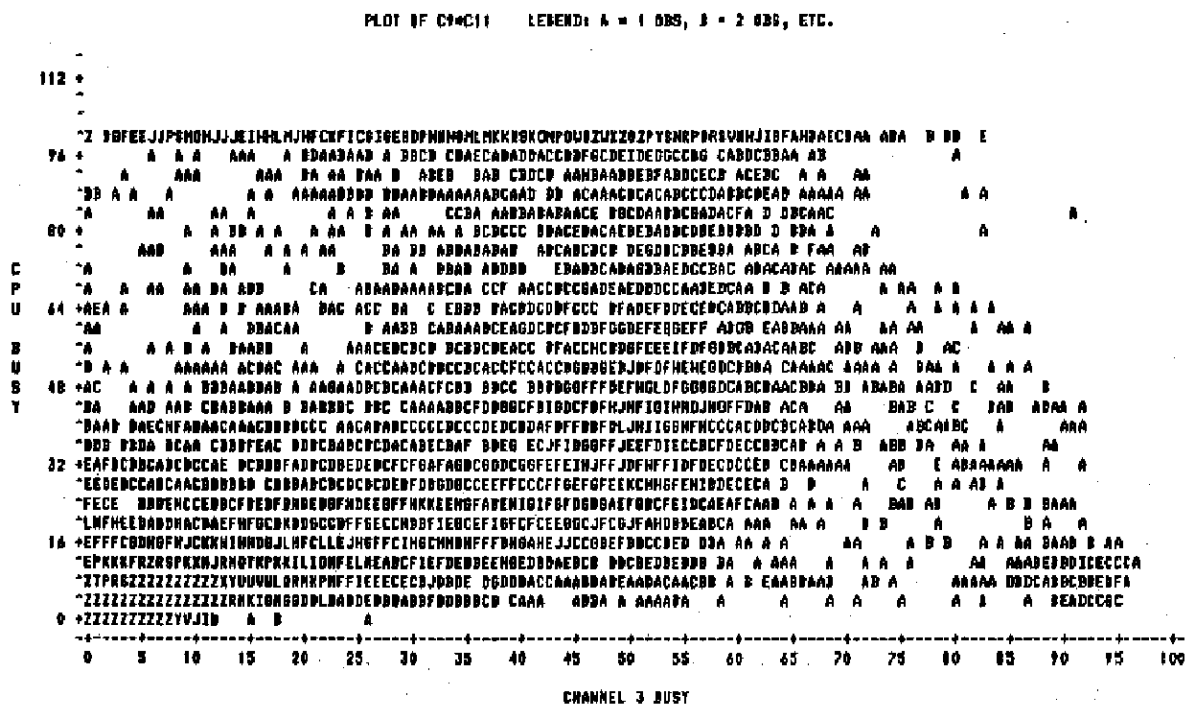


Figure 1: An Unilluminating Plot of 10,073 Points

SPLIT OF CPU BUSY VS CHANNEL 3 BUSY

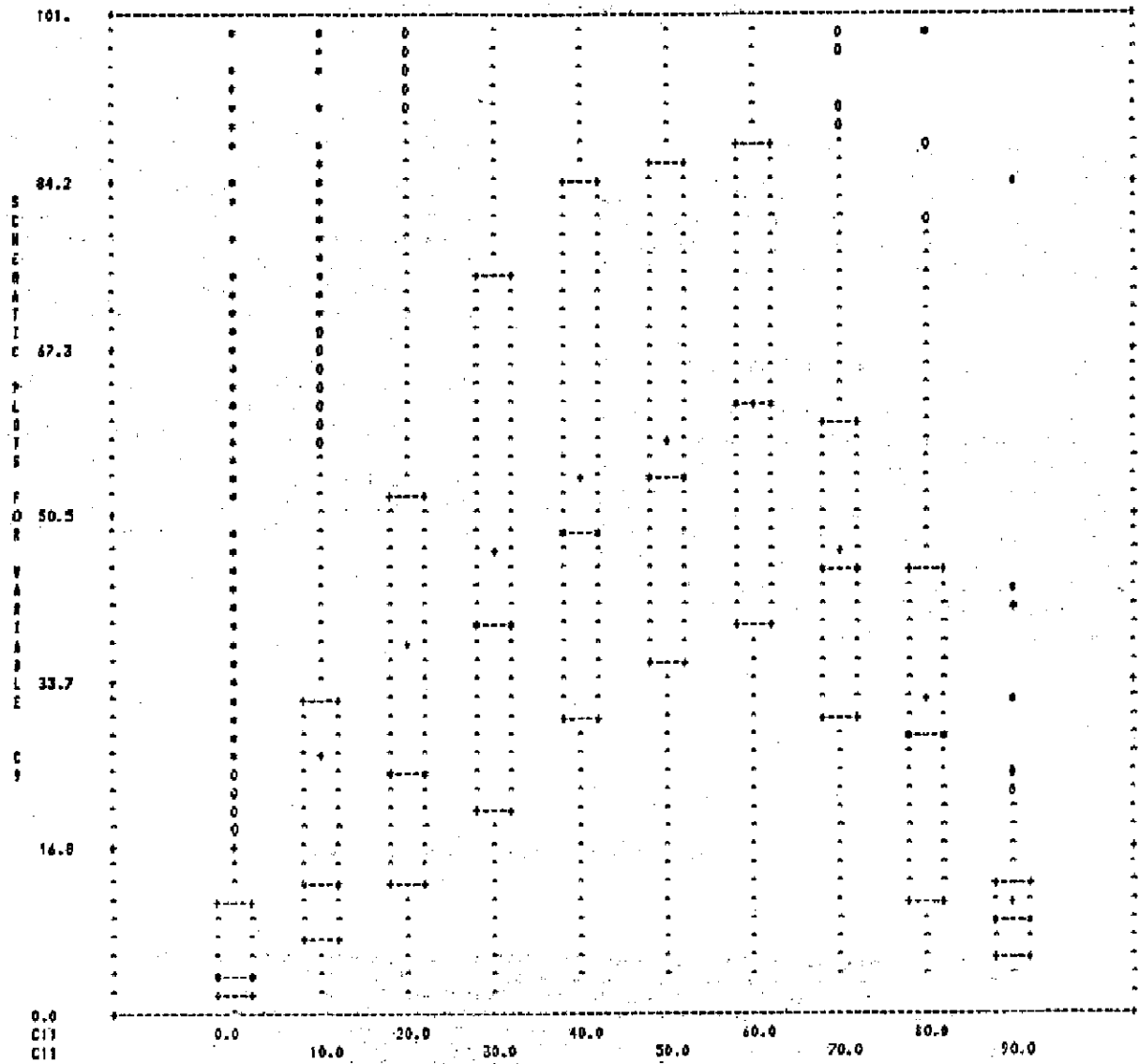


Figure 2: Informative Side-by-Side Schematic Plots of the Same Data

EXAMPLE 2: PRODUCING U. S. MAP REPORTS

The following two examples generate reports in the context of a user created pseudo SAS macro library. The steps in allocating space, storing the macros and calling them is given below:

Step 1: Setting up the Space for a SAS Macro Library

```
// EXEC PGM=IEFBRL4
//SASMCR0 DD UNIT=SYSDA,SPACE=(TRK,(1,1)),
// VOL=SER=DATA05,DSN=XXX.YYY.ZZZ,DISP=(,CATLG)
```

Step 2: Storing the Macro Definitions

```
// EXEC SAS
//SASMCR0 DD DSN=XXX.YYY.ZZZ,DISP=OLD
//SYSIN DD *
DATA;
FILE SASMACRO;
INPUT; PUT _INFILE_;
CARDS4;
:
SAS MACRO DEFINITIONS
:
:::
```

Step 3: To Use the Macros

```
// EXEC SAS,OPTIONS='NOSOURCE'
//SYSIN DD DSN=XXX.YYY.ZZZ,DISP=SHR
// DD *
```

SAS JOB INVOKING THE STORED MACROS

To produce a U. S. map filled with the number of SAS installations we first create a series of macros and place them in a macro library as outlined above. To put the final report, Figure 3, the final SAS job is:

```
// EXEC SAS,OPTIONS='NOSOURCE'
//SUGI DD DSN=....,DISP=SHR
//SYSIN DD DSN=....,DISP=SHR
// DD *
REDUCE; PUT_US_;
```

PUT_US_ is really a composite macro using calls to other building-block macros. These macros as well as REDUCE and PUT_US_ are given in detail in Appendix 1. A brief description of them follows:

REDUCE ...does a sort of the input data set by STATE and then does a match accumulation by STATE so that state totals (T1-T51) are totalled efficiently in a horizontal SAS array ready for formatting into the map. (This uses the ARRAY and DO...END features of SAS 79).

FILLUP ...fills up each state with its abbreviation and its state total

USTITL ...provides title lines as a header for the map

CENTRAL
WEST
MIDEAST
NORTHEAST
SOUTHEAST } macros which draw sections of the country with PUT statements

MAPGEN ...pieces together the different sections into one complete map of the U. S.

PUT_US_ ...brings together macros FILLUP->MAPGEN to produce the final report

We first note that this map report can be easily generalized to display any measure of interest (i.e., percentages of total or rankings) by state. For percentage we may want to put the state abbreviation followed by percent in a 3.1 format. To make the appropriate changes first, define a new macro FT as:

```
MACRO FT 3.5 %
```

Then generalize macro FILLUP by replacing the 2. format for totals to FT. For instance the first line of FILLUP becomes:

```
PUT #3 @9 'AK' +1 T1 FT #8 @9 'WA' +1 T48 FT
```

The other minor problem is that 3.1 may be too large of a format. For example, Connecticut

might spill into Rhode Island. Therefore, further adjustments will need to be made to accommodate enlarged formats. Connecticut's percentage could be put out to the right along with the other small eastern states.

Also, the modular approach gives us the flexibility to picture just one section of the country (by changing MAPGEN) together with the option of perhaps listing detail records next to the picture by adding a few PUT statements.

EXAMPLE 3: PRODUCING CALENDAR REPORTS

Another popular type of report with various business applications is a calendar report. The following set of macros and the subsequent job which calls them illustrates how to produce a calendar for January, 1979 for some fictitious sales data (Figure 4). The macros stored on the macro library are given in detail in Appendix 2 and described in general below:

- F1,F2,B1, ...building block macros which when B2,F,&S put together form FIVELN and SIXLN
- FIVELN ...puts the outline of a five row (week) calendar
- SIXLN ...puts the outline of a six row (week) calendar
- JANFILL
FEBFILL
MARFILL
...macros which fill up the cells with the appropriate day number for each month
- DECFFILL
- CAL ...fills up the calendar with items desired to be displayed. The SAS code in this macro takes as input a vertically structured file (i.e., one obs. per day) and restructures it into a row by row report using PUT statements. For calendar reports each row corresponds to 7 cells. This SAS code was first suggested by A. J. Barr at the 1976 SUGI meetings.

The SAS job:

```
DATA A; INPUT DAY SALES CALLS LEADS;
CARDS;
:
SALES DATA FOR JAN. 79
:
DATA B (KEEP=TSALES TCALLS TLEADS);
SET A END=EOF;
TSALES+SALES; TCALLS+CALLS; TLEADS+LEADS;
IF EOF THEN OUTPUT;
DATA JANSALES; MERGE A B;
MACRO INDATA JANSALES %

MACRO LABELS
#4 @2 'DAY' @47 'MONTHLY TOTALS'
#6 @2 'SALES' @47 'TOT.SALES'
#7 @2 'CALLS' @47 'TOT.CALLS'
#8 @2 'NEWLDS' @47 'TOT.NEW LEADS' %
```

```

DATA _NULL_ ; SET INDATA; RETAIN C 1;
CAL;
H: PUT FIVELN LABELS JANFILL;
   PUT #6 @61 TSALES 3.
     / @60 TCALLS 4.
     / @60 TLEADS 4.;
RETURN;

```

For another application, to produce user-specified calendar reports that summarize monthly computer usage activity we might redefine macros CAL and LABELS and proceed as follows.

```

DATA A (DROP DATE M Y );
SET USUAGE.DAILY (KEEP=DATE JOBS ACTIVEHR
                 IDLEHR CPUHR IPLS);
M=MONTH(DATE); Y=YEAR(DATE);
IF M=2 & YEAR=1979;

DATA B (KEEP=TJOBS TACTIVE TIDLE TCPUHR TIPLS);
SET A END=EOF;
TJOBS+JOBS; TACTIVE+ACTIVEHR; TIPLS+IPLS;
TIDLE+IDLEHR; TCPUHR+CPUHR;
IF EOF THEN OUTPUT;

DATA C; MERGE A B;
PJOB=JOBS/TJOBS;

```

MACRO INDATA C %

MACRO LABELS

```

#4 @2 'JOBS'      @ 'TOT.JOBS'
#5 @2 'PCTJOBS'   @ 'TOT.PCTJOBS'
#6 @2 'ACTIVEHR'  @ 'TOT.ACTIVE'
#7 @2 'IDLEHR'    @ 'TOT.IDLE'
#8 @2 'CPUHR'     @ 'TOT.CPUHR'
#9 @2 'IPLS'      @ 'TOT.IPLS' %

```

MACRO CAL

```

FILE PRINT N=PS HEADER=H NOTITLES;
J=MOD (C,7); I=INT (C/7);
LI=7*I+15; CJ=5+9*J;
PUT #LI @ CJ JOBS      4.
   / @ CJ PJOBS       4.1
   / @ CJ ACTIVE HR   4.1
   / @ CJ IDLEHR      4.1
   / @ CJ CPUHR       4.1
   / @ IPLS           4. ;
C=C+1; RETURN; %

```

```

DATA _NULL_ ; SET INDATA;
RETAIN C 4 ; * FEB 1, 1979 IS A THURS;
             * SO CELL COUNTER C=4 INITIALLY;

```

```

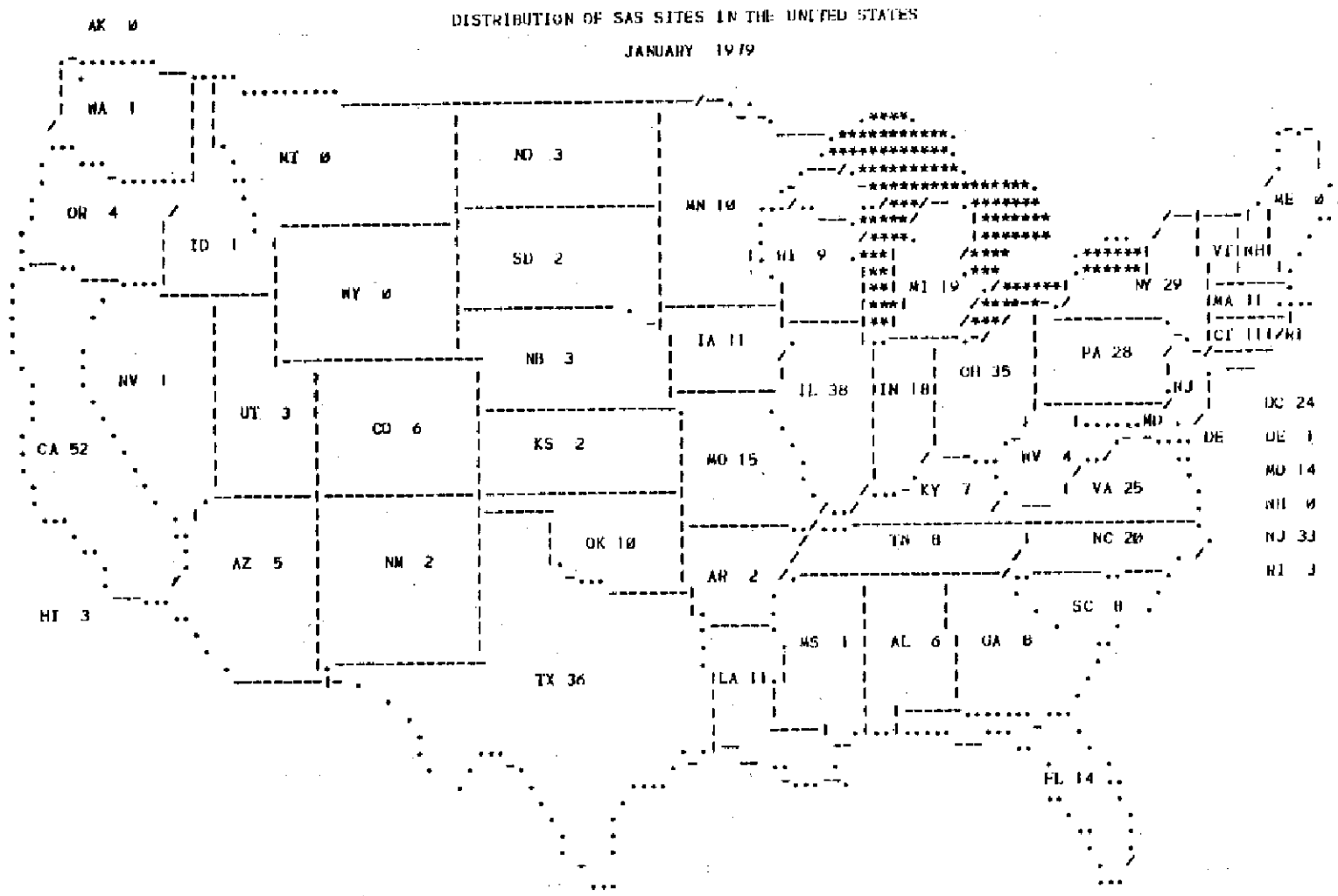
CAL;
H: PUT FIVELN LABELS FEBFILL;
   PUT #4 @58 TJOBS    6.
     // @58 TACTIVE 6.1
     / @58 TIDLE    6.1
     / @58 TCPUHR   6.1
     / @58 TIPLS    6. ;

```

We note that there are at most only 14 possible monthly calendars for each month (i.e., Jan 1 can fall on one of seven days and each year can be a leap year or a non-leap year). Thus to generalize the calendar reports to report on any month for any year we could use the month and year in question to link to the appropriate JANFILL, FEBFILL, ...DECFILL macro.

REFERENCES

1. Tukey, J. W., Exploratory Data Analysis, Addison-Wesley, Reading, Massachusetts (1977).
2. Cjertsen, W. R. and R. M. Gaddy, "SAS-A Unified Language for CPE," SAS Technical Report G-101. Raleigh: SAS Institute, 1978.



67

Figure 3: A U. S. Map Report

DAY	***** * J A N U A R Y * *****						MONTHLY TOTALS
SALES							TOT. SALES 145
CALLS							TOT. CALLS 779
NEWLDS							TOT. NEW LEADS 645
	S	M	T	W	T	F	S
		1	2	3	4	5	6
		0	9	7	3	2	5
		5	52	35	17	15	24
		1	10	9	32	33	13
7	8	9	10	11	12	13	
3	3	7	6	5	2	0	
16	9	36	29	37	22	7	
20	34	35	17	29	27	37	
14	15	16	17	18	19	20	
1	3	5	8	3	4	6	
13	16	27	41	15	21	27	
1	20	19	23	23	25	15	
21	22	23	24	25	26	27	
2	1	12	5	8	11	4	
18	11	33	24	51	44	22	
0	6	24	12	22	25	5	
28	29	30	31				
0	9	5	6				
4	41	36	31				
29	37	38	24				

Figure 4: A Calendar Report

MACRO REDUCE

```

-----*
* REDUCE DOES A SORT OF THE INPUT DS BY STATE THEN *
* DOES A TABLE LOOKUP ACCUMULATION BY STATE SO *
* THAT STATE COUNTS ARE TOTALLED EFFICIENTLY IN A *
* HORIZ. ARRAY READY FOR FORMATTING INTO THE MAP *
-----*

```

APPENDIX 1:
 MACROS FOR A U. S.
 MAP REPORT

```

PROC SORT DATA=SUGI.INSTALL OUT=INDATA; BY STATE;
DATA RDATA; SET INDATA(KEEP=STATE USE);
RETAIN K 1 T1-T51 0 S1 'AK' S2 'AL' S3 'AR' S4 'AZ' S5 'CA'
      S6 'CO' S7 'CT' S8 'DC' S9 'DE' S10 'FL' S11 'GA'
      S12 'HI' S13 'IA' S14 'ID' S15 'IL' S16 'IN' S17 'KS'
      S18 'KY' S19 'LA' S20 'MA' S21 'MD' S22 'ME' S23 'MI'
      S24 'MN' S25 'MO' S26 'MS' S27 'MT' S28 'NB' S29 'NC'
      S30 'ND' S31 'NH' S32 'NJ' S33 'NM' S34 'NV' S35 'NY'
      S36 'OH' S37 'OK' S38 'OR' S39 'PA' S40 'RI' S41 'SC'
      S42 'SD' S43 'TN' S44 'TX' S45 'UT' S46 'VA' S47 'VT'
      S48 'WA' S49 'WI' S50 'WV' S51 'WY' ;
ARRAY T (K) T1-T51;
ARRAY ST (I) S1-S51;
IF USE='Y' USE='N';
DO I=K TO 51;
  IF ST=STATE THEN DO; K=I; T+1; RETURN; END;
END;
%

```

MACRO FILLUP

```

-----*
* FILLUP FILLS UP THE MAP WITH STATE TOTALS *
-----*
PUT #3 09 'AK' +1 T1 2. #8 09 'WA' +1 T48 2.
#11 028 'MI' +1 T27 2. #51 'ND' +1 T30 2.
#14 068 'MN' +1 T24 2. #126 'ME' +1 T22 2.
#7 'OR' +1 T38 2.
#16 019 'ID' +1 T14 2.
#17 051 'SD' +1 T42 2. #77 'WI' +1 T49 2. #120 'VTINH'
#19 034 'WY' +1 T51 2. #90 'MI' +1 T23 2. #112 'NY' +1 T35 2.
#20 0120 'MA' +1 T20 2.
#22 069 'IA' +1 T13 2. #120 'CT' +1 T7 #127 'RI'
#23 052 'NB' +1 T28 2. #107 'PA' +1 T39 2.
#24 012 'NV' +1 T34 2. #95 'OH' +1 T36 2.
#25 079 'IL' +1 T15 2. #87 'IN' +1 T16 2. #116 'NJ'
#26 024 'UT' +1 T45 2. #125 'DC' +1 T8 2.
#27 037 'CO' +1 T6 2. #113 'MD'
#28 04 'CA' +1 T5 2. #53 'KS' +1 T17 2.
#119 'DE' #125 'DE' +1 T9 2.
#29 070 'MO' +1 T25 2. #101 'WV' +1 T50 2.
#30 0125 'MD' +1 T21 2.
#31 091 'KY' +1 T18 2. #108 'VA' +1 T46 2.
#32 0125 'NH' +1 T31 2.
#34 058 'OK' +1 T37 2. #88 'TN' +1 T43 2.
#108 'NC' +1 T29 2. #125 'NJ' +1 T32 2.
#35 023 'AZ' +1 T4 2. #38 'NM' +1 T33 2.
#36 070 'AR' +1 T3 2. #125 'RI' +1 T40 2.
#38 04 'HI' +1 T12 2. #106 'SC' +1 T41 2.
#40 079 'MS' +1 T26 2. #88 'AL' +1 T2 2. #97 'GA' +1 T11 2.
#42 053 'TX' +1 T44 2. #71 'LA' +1 T19 2.
#48 0103 'FL' +1 T10 2. %

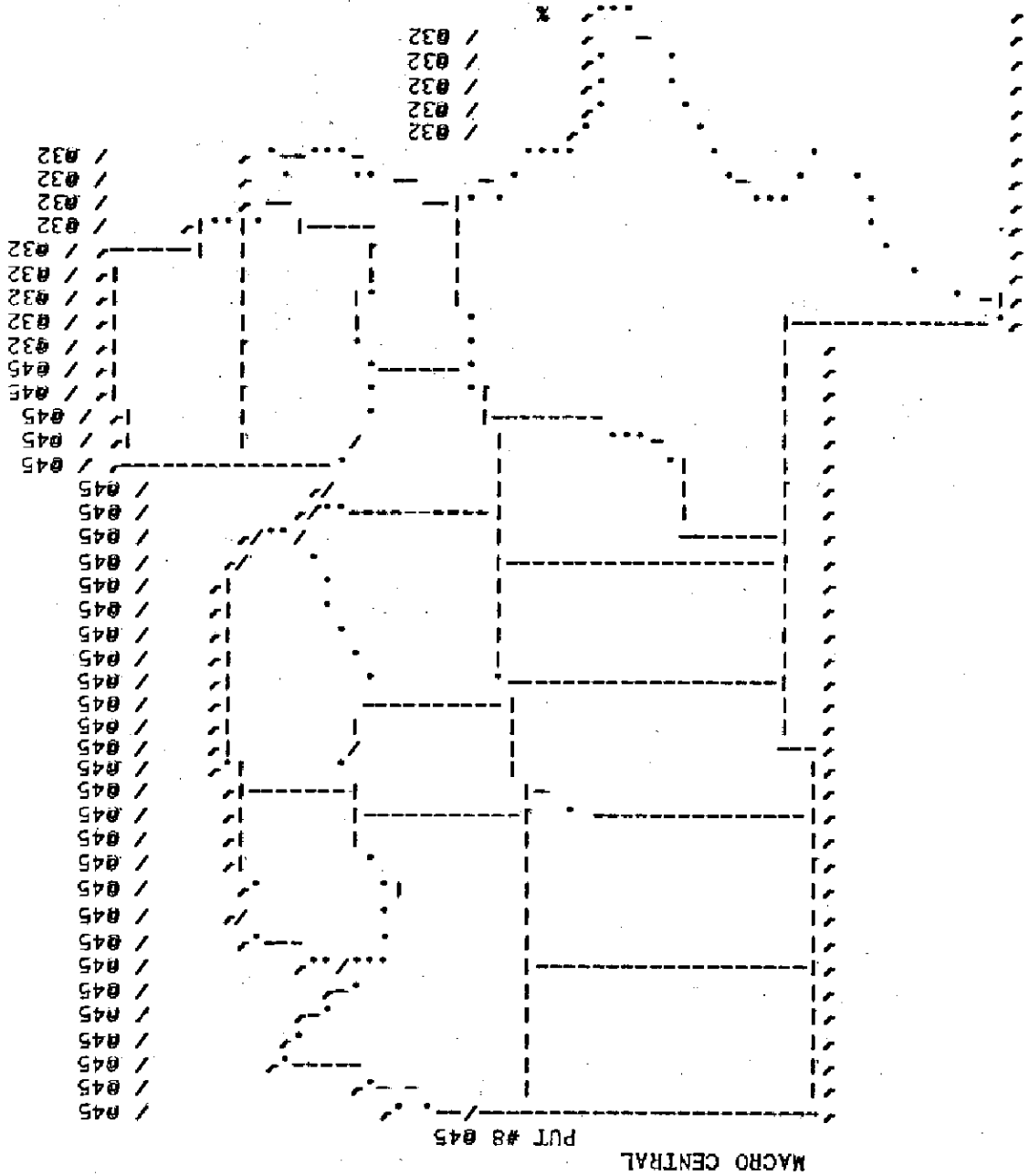
```

MACRO USTITL

```

-----*
* TITL SUPPLIES TITLE LINES AS A HEADER FOR *
* THE MAP REPORT *
-----*
PUT #3 045 'DISTRIBUTION OF SAS SITES IN THE UNITED STATES'
#5 062 'JANUARY 1979' %

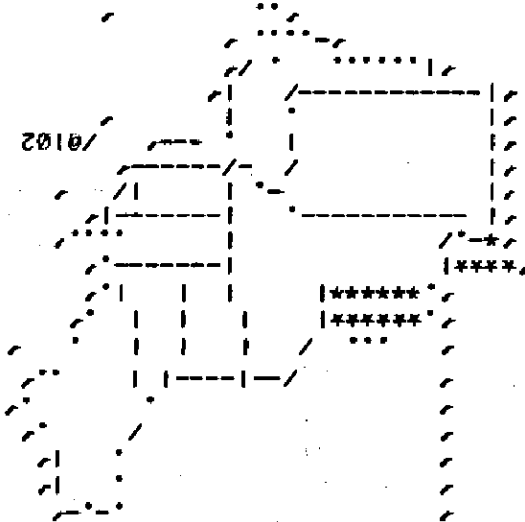
```



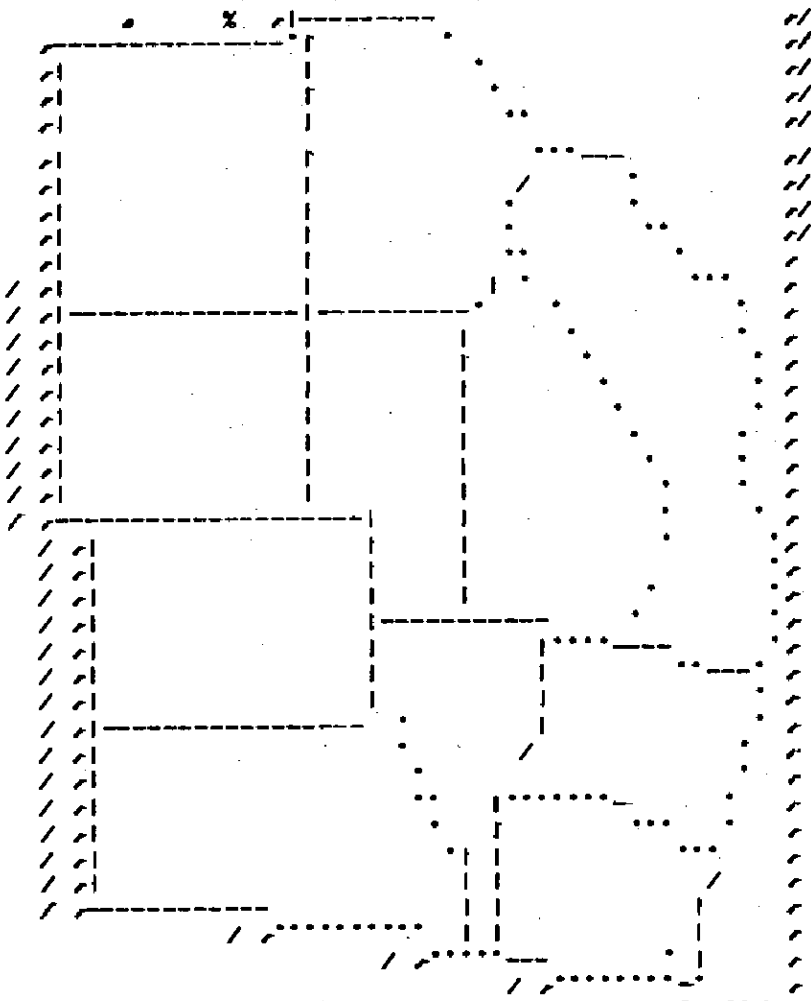

```

/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0106
/ 0113
/ 0116
%

```



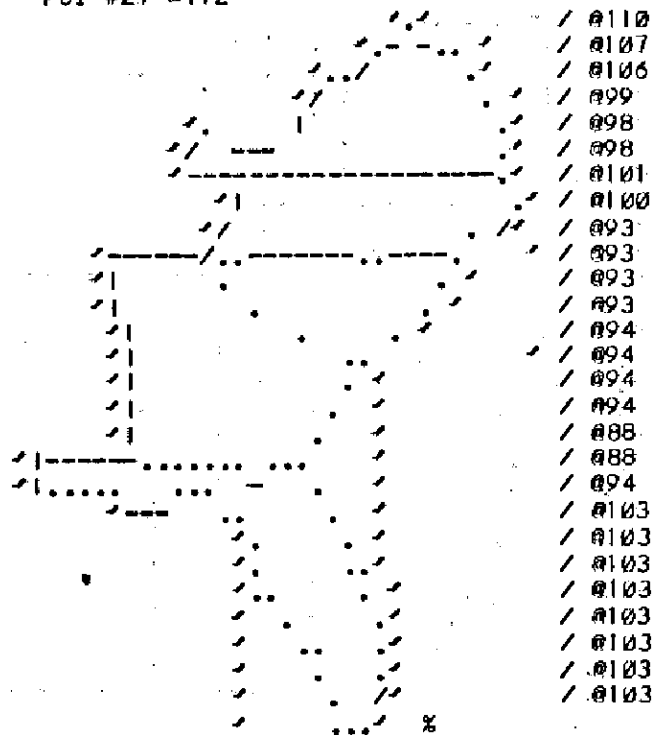
MACRO NORTH
PUT #10 0106



MACRO WEST
PUT #5

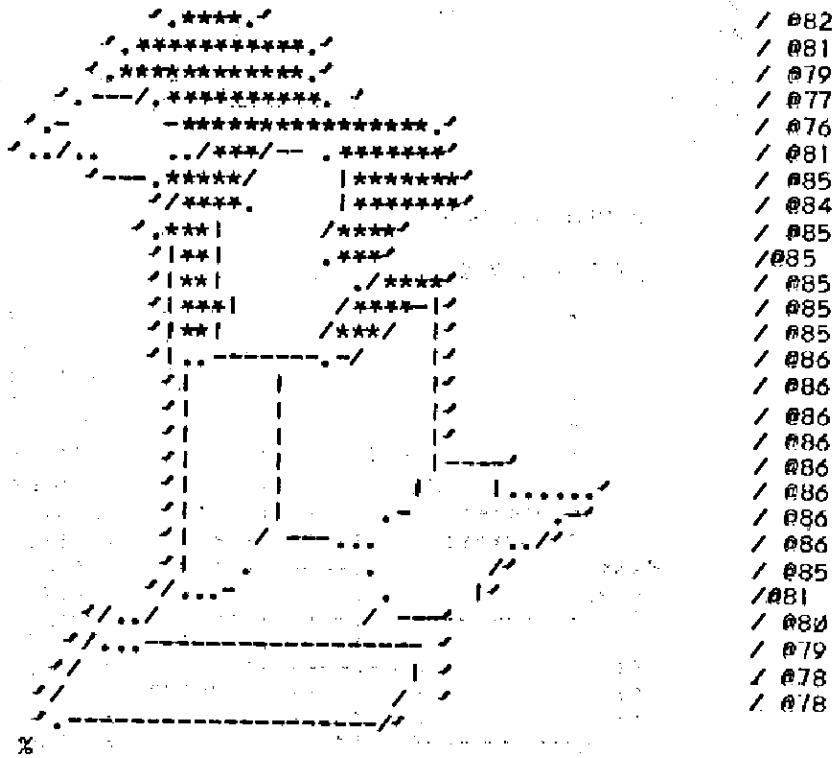
MACRO SOUTHEAST

PUT #27 #112



MACRO MIDEAST

PUT #9 #85



MACRO MAPGEN

CENTRAL;WEST;MIDEAST;SOUTHEAST;NORTHEAST; %

MACRO PUT_US_

```
-----*
* PUT_US IS THE MACRO WHICH BRINGS TOGETHER *
* ALL THE PREVIOUS MACROS TO GENERATE THE *
* REPORT. TO PRODUCE FINAL MAP REPORT JUST *
* INVOKE: REDUCE PUT_US_ *
*-----*
DATA _NULL_ SET RDATA;
FILE PRINT N=PS HEADER=H NOTITLES;
FILLUP; RETURN;
H:USTITL;MAPGEN;RETURN; %
```

APPENDIX 2: MACROS FOR MONTHLY CALENDAR REPORTS

MACRO CALEXPLN

```
-----*
* THE FOLLOWING SET OF MACROS PRODUCE CALENDAR REPORTS *
* THE MAJOR MACROS ARE FIVELN & SIXLN: (FOR 5 OR IN THE *
* CASE OF SEP & DEC 79 6 LINE CALENDARS) LABELS: FOR *
* LABELLING WHAT GOES IN THE BOXES: JANFILL-DECFILL *
* GIVES MONTH TITLES AND FILLS IN THE DAYS. CAL: FILLS *
* IN THE REST OF BOXES WITH THE INFORMATION DESIRED TO *
* BE DISPLAYED. CAL DOES NOT HOWEVER DO THE ENTIRE *
* REPORT SINCE EACH MONTH STARTS ON A DIFFERENT DAY. *
* FOR INSTANCE TO PUT A CALENDAR REPORT FOR JAN79 USE *
* THE FOLLOWING STATEMENTS: *
*   MACRO INDATA JANSALLES *
*   MACRO LABELS <HEADER LABEL FORMATS> *
*   DATA _NULL_ SET INDATA RETAIN C I *
*   CAL *
*   H: PUT FIVELN LABELS JANFILL *
*   PUT #6 @61 TSALES 3./ @60 TCALLS 4. *
*   / @60 TLEADS 4. RETURN *
*-----* %
```

```
MACRO F1 '+-----+' %
MACRO F2 '+-----+' %
MACRO B1 '+-----+' %
MACRO B2 '+-----+' %
MACRO F '+-----+' %
MACRO S '+-----+' %
```

MACRO FIVELN

```
#3 F1 046 F2
/ B1 046 B2
/ B1 046 B2
/ B1 046 B2
/ B1 046 B2
/ B1 046 B2
/ B1 046 B2
/F
/S
/ 06 'S' 015 'M' 024 'T' 033 'W' 042 'T' 051 'F' 060 'S'
/S
/F
/S/S/S/S/S/S/F
/S/S/S/S/S/S/F
/S/S/S/S/S/S/F
/S/S/S/S/S/S/F
/S/S/S/S/S/S/F %
```

MACRO SIXLN FIVELN /S/S/S/S/S/S/F %

```
MACRO JANFILL #3 023 /*****
/ 023 /*
/ 023 /* JANUARY /*
/ 023 /*
/ 023 /*****
```

```
#15 011 '1' +8 '2' +8 '3' +8 '4' +8 '5' +8 '6'
#22 02 '7' +8 '8' +8 '9' +8 '10' +7 '11' +7 '12' +7 '13'
#29 02 '14' +7 '15' +7 '16' +7 '17' +7 '18' +7 '19' +7 '20'
#36 02 '21' +7 '22' +7 '23' +7 '24' +7 '25' +7 '26' +7 '27'
#43 02 '28' +7 '29' +7 '30' +7 '31' %
```

```
MACRO FEBFILL #3 022 /*****
/ 022 /*
/ 022 /* FEBRUARY /*
/ 022 /*
/ 022 /*****
```

```
#15 038 '1' +8 '2' +8 '3'
#22 02 '4' +8 '5' +8 '6' +8 '7' +8 '9' +8 '10'
#29 02 '11' +7 '12' +7 '13' +7 '14' +7 '15' +7 '16' +7 '17'
#36 02 '18' +7 '19' +7 '20' +7 '21' +7 '22' +7 '23' +7 '24'
#43 02 '25' +7 '26' +7 '27' +7 '28' %
```

```
MACRO MARFILL #3 025 /*****
/ 025 /*
/ 025 /* MARCH /*
/ 025 /*
/ 025 /*****
```

```
#15 038 '1' +8 '2' +8 '3'
#22 02 '4' +8 '5' +8 '6' +8 '7' +8 '8' +8 '9' +8 '10'
#29 02 '11' +7 '12' +7 '13' +7 '14' +7 '15' +7 '16' +7 '17'
#36 02 '18' +7 '19' +7 '20' +7 '21' +7 '22' +7 '23' +7 '24'
#43 02 '25' +7 '26' +7 '27' +7 '28' +7 '29' +7 '30' +7 '31' %
```

MACRO CAL

```
FILE PRINT N=PS HEADER=H NOTITLES;
J=MOD(C,7); * J COUNTS HOW FAR YOU ARE ALONG EACH WEEK;
I=INT(C/7); * I INCREMENTS THE LINE;
LI=7*I+17; * LI IS THE LINE TO SKIP DOWN TO;
CJ=3+9*J; * CJ IS THE COLUMN TO SKIP OVER TO;
PUT #LI @CJ SALES 2.
/ @CJ CALLS 2.
/ @CJ LEADS 2.;
C=C+1; * C IS THE COUNTER OF 'CELLS' (IE. 0 TO 34 FOR A 5 WEEK MO.);
RETURN;
%
```