

Market Research Using SAS/AF®

James F. Moore
MND Service Inc

Purpose

The Woodlands Corporation, a division of Mitchell Energy and Development, is an internationally acclaimed real estate development company. The primary emphasis of the company is on the development of The Woodlands, a master-planned community just north of Houston, Texas. Additional developments are Pirate's Beach, located on Galveston Island, and Cape Royale, on Lake Livingston, as well as several smaller developments.

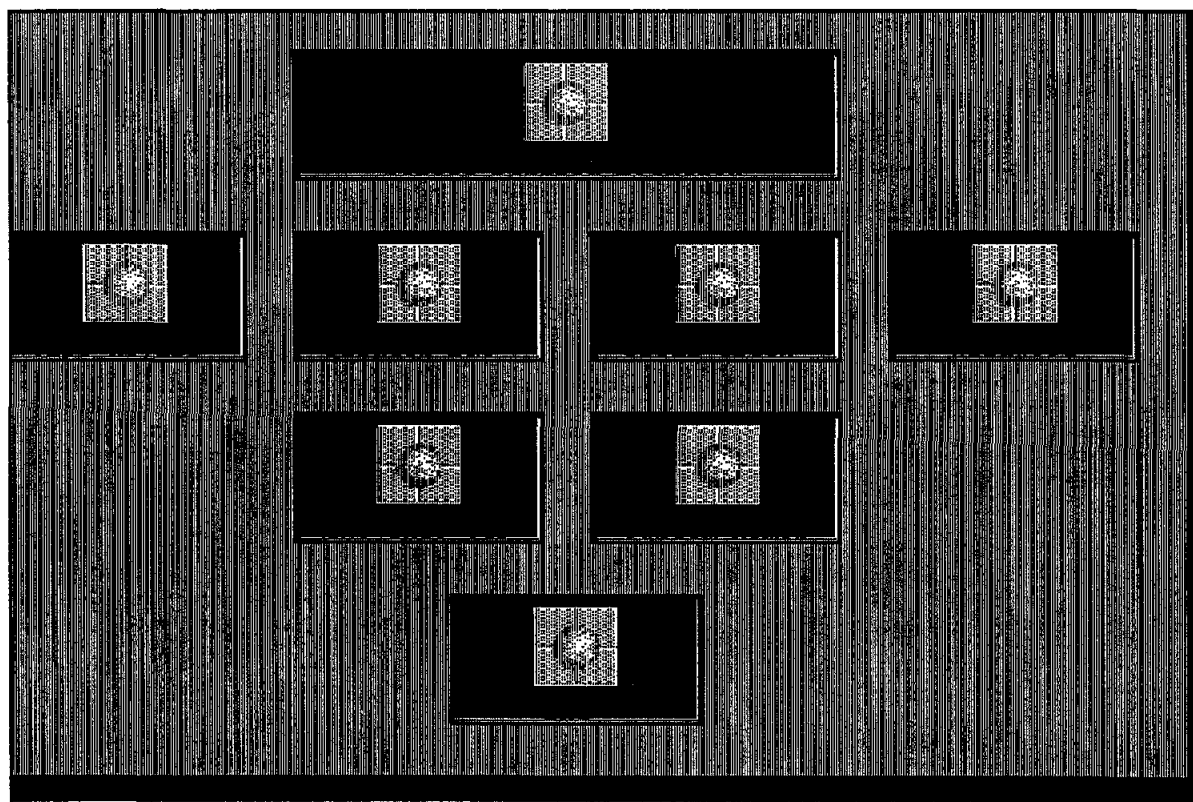
At each of these locations an information center is maintained to provide assistance to visitors. When a visitor arrives a brief questionnaire about the visitor's interests and current housing plans is filled out. This form is a valuable source of information for housing inventory control and planning, and also provides demographic information for market research and strategy. To record such data a computer-based application was developed using a combination of SAS® products. SAS-AF® was used to create a set of point-and-click windows used by the Information Center's staff to:

- Enter survey responses
- Update survey fields and Reports
- Produce Periodic Management Reports
- Develop Reports and Lists based on Query-By-Example

Reports are generated by Proc Report and SAS/FSP® is used for basic edit functions.

Functions

Using the SAS-AF® Block function a general menu was designed to set file allocations based upon the real estate development chosen. The visitor registration and survey entry share all other capabilities.



Main Menu

Since it is possible, and in some cases likely, that a visitor may visit one or more of the information centers, it was decided that all sites would share a common base record, which contains basic information such as name and address. As these records are generated the system generates a unique ID number (BASECODE). To enter base information the user enters the last name of the visitor in the Registration window.

The screenshot shows a 'Registration Window' with a title bar 'entering visitor base window'. Below the title bar is a 'Function menu' with options: 'SEARCH', 'ADD', 'DELETE', 'EDIT', 'PRINT', 'QUIT'. The main area contains a form with the following fields: 'NAME' (with a label 'Last Name'), 'ADDRESS' (with a label 'Street Address'), 'PHONE' (with a label 'Phone Number'), 'CITY' (with a label 'City'), 'STATE' (with a label 'State'), 'ZIP' (with a label 'Zip'), 'FAX' (with a label 'Fax'), 'E-MAIL' (with a label 'E-mail Address'), and 'URL' (with a label 'Web Site'). There are also checkboxes for 'Male' and 'Female'. At the bottom, there is a 'Work Center or Desired Function and Other Notes' field.

Registration Window

When an enter key is sent the Base Record dataset is searched for unique matches on the entered field(s) and a selection list is presented to the user. The user may enter either a name or BASECODE.

The screenshot shows a 'Selection List' window. It has a title bar 'entering visitor base window' and a 'Function menu' with options: 'SEARCH', 'ADD', 'DELETE', 'EDIT', 'PRINT', 'QUIT'. The main area displays a list of search results. The first entry is highlighted. The list includes columns for 'NAME', 'ADDRESS', 'PHONE', 'CITY', 'STATE', 'ZIP', 'FAX', 'E-MAIL', and 'URL'. At the bottom, there are navigation buttons: '< > << >> < > << >>'.

Selection List

After a visitor is identified, the user can access either the base record or survey record. The push buttons at the bottom of the Registration window, or Function keys may be used to initiate other windows.

EDIT/ADD Base Record

Base Record

The Base record fields **Zip Code** and **Employer** are examined for matches to entries in a zipcode and employer tables. If matches are not made the user is prompted to either update the tables or select a table entry.

Employer Update

Survey Data

The screenshot shows a terminal window titled "Survey Data". It contains a form with several sections. The top section has fields for "Home Price" (with a list of ranges from 0-60 to 400+), "Home Type" (with options like New Home, Resale Home, etc.), and "How Did You Hear..." (with options like Newspaper Ad, Radio, etc.). There are also fields for "Household Type" (Family, Couple, Single) and "Age" (0-60 to 60-90). The bottom of the window shows navigation instructions: "F2 - Canc", "F3 - End".

Survey Data

The survey window consists of a set of *point-and-click* fields that reflect the questions on the survey form. Multiple responses are allowed on the "Home Price", "Home Type" and "How Did You Hear..." fields.

The screenshot shows a terminal window titled "Information Center Visitor Profile". It contains a form with several sections. The top section has fields for "Home Price" (with a list of ranges from 0-60 to 400+), "Home Type" (with options like New Home, Resale Home, etc.), and "How Did You Hear..." (with options like Newspaper Ad, Radio, etc.). There are also fields for "Household Type" (Family, Couple, Single) and "Age" (0-60 to 60-90). The bottom of the window shows navigation instructions: "F2 - Canc", "F3 - End".

Detail Prompts

Additional detail can be recorded for the "How Did You Hear..." fields: *Newspaper*, *Radio*, and *Realtor*. When the user selects these fields a dialogue box will prompt for further detail. The user may make a new entry or select from the current list.

Information Center Visitor Profile			
Command ==>			
Table Entry - Update			
Home Pr	Command ==>	155522	
0-60		ALI	
60-90	Specify Real Agent		
90-120	Buddy Ebson	90	
120-160			
160-200			
200-300	Senior Living		
300-400	Apartment		
400+	Lease Home		
How Did You Hear...? Household Type: Adult Ages: Children:			
Newspaper Ad	Family	UNDER 19	
Radio	Couple	20-29	
Friend/Relative	Single	30-39	
Realtor		40-49	
Info In MALL		50-59	4+ - - - -
Work Nearby		60-69	
Driving Around		70 up	
Conf. Center			
Direct Main	F2 - Cancel	F4 - Previous Question	
Other	F3 - End	F12- Next Question	

Addition to Tables

Notice that the new entry generates a code number. These responses are stored as two byte fields. The user is actually updating a SAS format control file.

In the case of entering a Realtor, the user is automatically prompted to enter the Agency the agent represents.

Information Center Visitor Profile			
Command ==>			
Table Entry - Update			
Home Pr	Command ==>	155522	
0-60		ALI	
60-90	Specify Realtor		
90-120	Beverly Hills Realty	2b	
120-160			
160-200			
200-300	Senior Living		
300-400	Apartment		
400+	Lease Home		
How Did You Hear...? Household Type: Adult Ages: Children:			
Newspaper Ad	Family	UNDER 19	
Radio	Couple	20-29	
Friend/Relative	Single	30-39	
Realtor		40-49	
Info In MALL		50-59	4+ - - - -
Work Nearby		60-69	
Driving Around		70 up	
Conf. Center			
Direct Main	F2 - Cancel	F4 - Previous Question	
Other	F3 - End	F12- Next Question	

The completed survey will contain the realtor name and agency.

Information Center Visitor Profile
Command >

Home Price		DB: 155522
0-50	New Home (Single)	ALI
50-90	Resale Home	
90-120		
120-150		
150-200		
200-300	Senior Living	
300-400	Apartment	
400+	Lease Home	

How Did You Hear...? Household Type: Adult Ages: Children:

Newspaper Ad	Family	UNDER 19	0
Radio	Couple	20-29	1
Friend/Relative	Single	30-39	2
Realtor	BUDDY EBSCH	40-49	3
Info in MALL	BEVERLY HILLS REALTY	50-59	4+
Work Nearby		60-69	
Driving Around		70 up	
Conf. Center			
Direct Main	F2 - Cancel	F4 - Previous Question	
Other	F3 - End	F12- Next Question	

Sample Survey

Delete/Undelete

The user may Delete or Undelete a visitor by using the push buttons or Function keys.

The screenshot shows a terminal window titled "Marking Deleted Data Utility". The menu includes the following options:

- ADD (F1)
- EDIT (F2)
- DELETE (F3)
- UNDELETE (F4)
- EXIT (F5)
- PRINT (F6)
- RECALL (F7)
- RECALL (F8)
- RECALL (F9)
- RECALL (F10)
- RECALL (F11)
- RECALL (F12)

Below the menu, there is a section for "MORE CHOICES OF MARKING DELETED AND PRINTER STATUS" and a "FUNCTION KEYS:" section.

Locate Base Record

The front-end process to delete a visitor is the same as Adding or Editing. The base file is searched for matches. If multiple matches are found a selection list is presented. When a base record is found, the base record and survey records are copied to backup files and deleted from the master.

This screenshot is identical to the one above, showing the "Marking Deleted Data Utility" menu with options: ADD (F1), EDIT (F2), DELETE (F3), UNDELETE (F4), EXIT (F5), PRINT (F6), RECALL (F7), RECALL (F8), RECALL (F9), RECALL (F10), and RECALL (F11).

Delete Base & Survey Records

To Undelete a visitor the same procedure is used except that the file pointers to the backup and master files are exchanged.

All of the Field Names, Headers, and Selection Values are maintained as formats which the user may edit by pressing the Tables push button on the Registration window.



The Real Estate Agent file contains codes for both the agent and realtor. This allows us to display both the agent and agency when the user selects a realtor from the selection list when the survey data is entered.

Realtor Table

The question heading file is used to produce standard reports. This allows the user to change the heading, order or sort control of responses on the report.

Ques	Question	Answer	Mark
1	1. The first step in the process of creating a new project is to create a new project file.	True	1
2	2. The second step in the process of creating a new project is to create a new project file.	True	1
3	3. The third step in the process of creating a new project is to create a new project file.	True	1
4	4. The fourth step in the process of creating a new project is to create a new project file.	True	1
5	5. The fifth step in the process of creating a new project is to create a new project file.	True	1
6	6. The sixth step in the process of creating a new project is to create a new project file.	True	1
7	7. The seventh step in the process of creating a new project is to create a new project file.	True	1
8	8. The eighth step in the process of creating a new project is to create a new project file.	True	1
9	9. The ninth step in the process of creating a new project is to create a new project file.	True	1
10	10. The tenth step in the process of creating a new project is to create a new project file.	True	1

Question Name Table

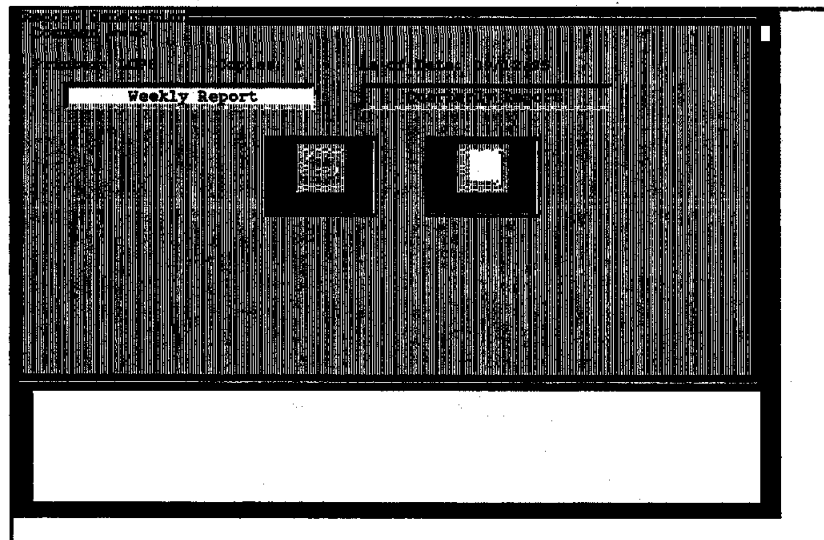
Response values for the screen responses are stored as one file.

Ques	Question	Answer	Mark
1	1. The first step in the process of creating a new project is to create a new project file.	True	1
2	2. The second step in the process of creating a new project is to create a new project file.	True	1
3	3. The third step in the process of creating a new project is to create a new project file.	True	1
4	4. The fourth step in the process of creating a new project is to create a new project file.	True	1
5	5. The fifth step in the process of creating a new project is to create a new project file.	True	1
6	6. The sixth step in the process of creating a new project is to create a new project file.	True	1
7	7. The seventh step in the process of creating a new project is to create a new project file.	True	1
8	8. The eighth step in the process of creating a new project is to create a new project file.	True	1
9	9. The ninth step in the process of creating a new project is to create a new project file.	True	1
10	10. The tenth step in the process of creating a new project is to create a new project file.	True	1

Response Value Table

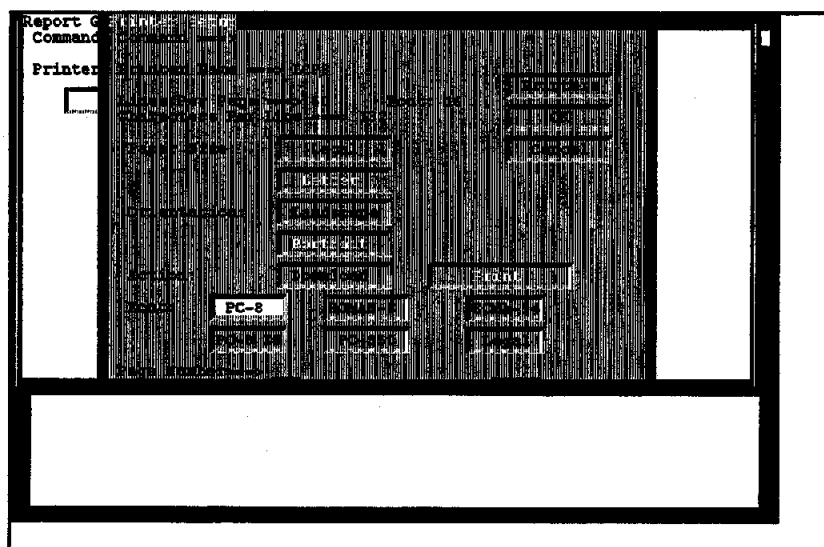
Reports

Standard Weekly and Quarterly reports are produced by using the Reports push button or Function key in the Registration Window.



Report Select Window

Print control characters are generated by a home-grown utility that takes into account the various printer types and communication protocols that are used throughout the company.



General Print Window

A job is submitted to the background to produce the requested report.

Query By Example

To allow the user to produce mailing lists and special reports the survey data AF program was used as a basis for a Query-by-Example program.

The screenshot shows a terminal window titled "Query By Example". It contains several sections for inputting search criteria. On the left, there are two boxes for price ranges: "60-90" and "90-120". In the center, there are two boxes for home type: "New Home (Single)" and "Resale Home". On the right, there are two boxes for time periods: "3 - 6 Months" and "6 - 12 Months". Below these, there are more boxes for "Children" and "Date Range". At the bottom, there are boxes for "City" and "State".

Query By Example

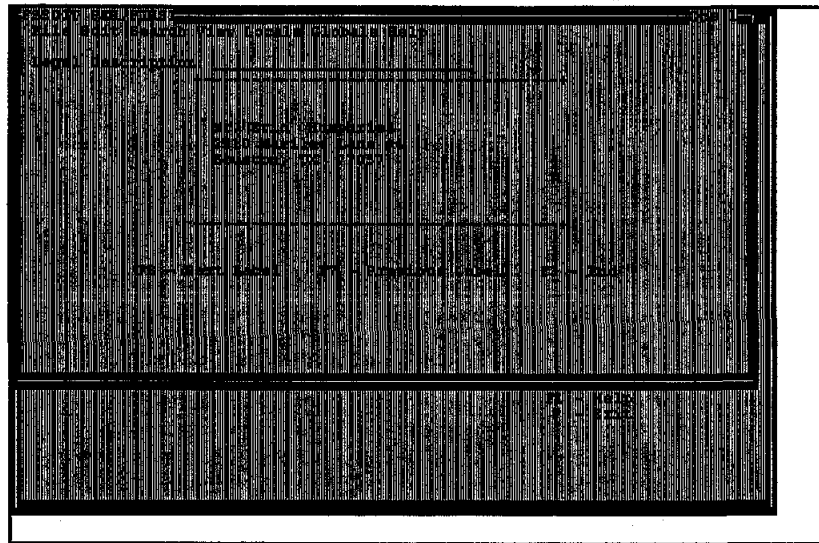
The user is required to enter a date range, but beyond that this window operates like the Survey window. The request shown in the figure above requests those visitors during January who were looking for a new or used home priced between 60-120 thousand dollars who intend to move within the next three to six months.

As the AF program runs legends are displayed to keep the user apprised of the progress. Once the data has been prepared the user can select a sort order for the data.

The screenshot shows a terminal window titled "Test/Sort Labels Window". It contains the following text: "Command ==>", "To Extract Entire Data Set, Leave Number of Observations BLANK", "Number of Observations (For Test): _____ Extracted:", "-----Sort Fields (In order Selected)-----", and a row of buttons: "None", "ZIP Code", "Last Name", "City", "State", and "Street". Below the buttons is a small box for input. At the bottom, it says "After Making Your Selections, Press the ENTER Key" and "F1 - Help", "F - Exit".

Test/Sort Labels Window

The user is then presented with the image of the labels as they will be printed.



Edit Label Images

After the data has been edited, the user has the options of: printing the labels locally; downloading the file to a PC; or, Returning to the Editor.

```
Command ==>
To Extract Entire Data Set, Leave Number of Observations BLANK
Number of Observations (For Test): 26      Extracted: 26
First:           Maximum Lines:           Maximum Length: 29

-----Sort Fields (In order Selected)-----
None  ZIP Code  Last Name  City  State  Street

      [ ]

After Making Your Selections, Press the ENTER Key
Select Output Style and Print Labels
Create Download file for PC
Return to Edit Screen

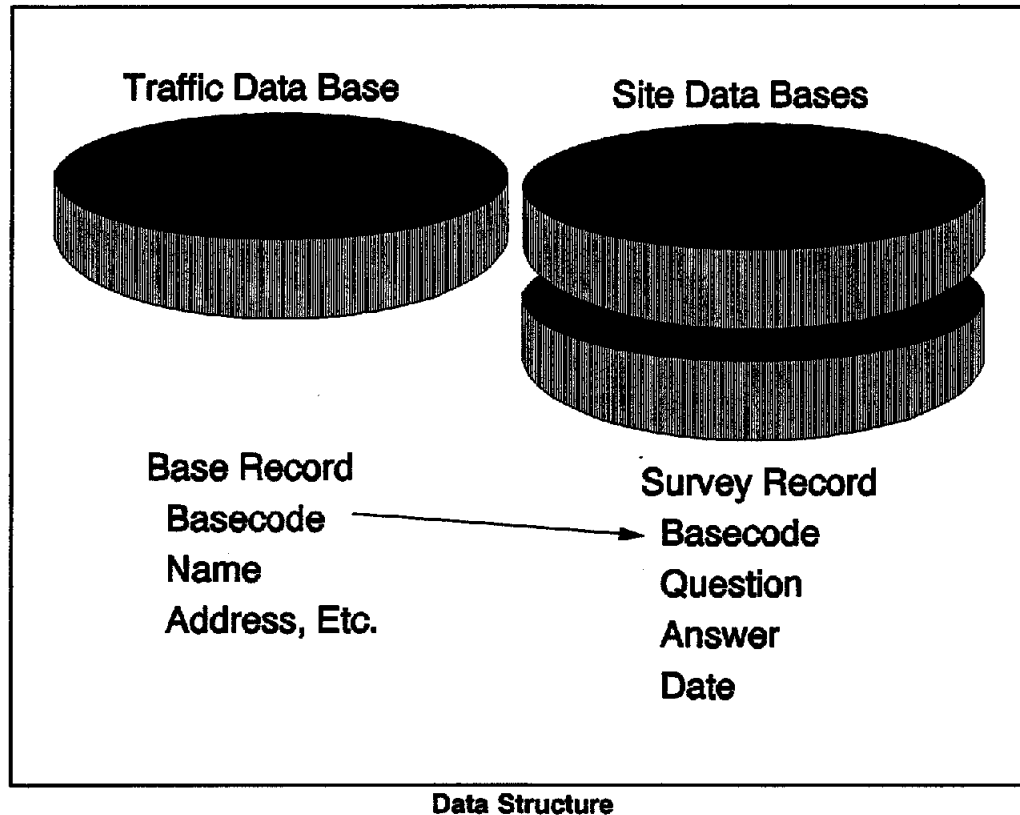
F1 - Help
F  - Exit
```

Sort/Edit Labels Window

Subsequent windows provide the user with the option of storing the data as /s for later use and selecting the desired label stock and printer options. After the user has completed these requests the Registration window is displayed.

Methods

Data Structure



Using SAS/Share® allows all of the information centers update access to the BASE member of the SAS® library "TRAFFIC."

Both the BASE and SURVEY files are indexed. The BASE file has two indices:

- NAMES LASTNAME
 FRSTNAME
 ZIPCODE
- BASE BASECODE
 VISTDATE

The survey file is indexed by BASECODE, VISTDATE and QUESTION.

Unique Basecodes

To allocate unique record ids (BASECODE) the following fragment is used:

```
if basecode=BLANK then do;
  code=open('traffic:basecode','u');
  do until(fetch(code)=0);
  end;
  basecode=getvarn(code,1)+1;
  call putvarn(code,1,basecode);
  rc=update(code);
  rc=unlock(code);
end;
```

Initializing Survey Window

```
array cxx{11} {8,8,1,1,10,1,1,1,10,1,1};
array qall{11,10} n1x1 - n1x8 dummy dummy
      .
      .
      .
array quest{11,10} $6 temporary
      {'n1x1','n1x2','n1x3','n1x4','n1x5','n1x6',
      .
      .
      .

CALL WREGION(8,20,4,50,' ');
CALL PUTLEGEND(1,'Searching the Survey File','RED','REVERSE');
CALL PUTLEGEND(2,'Please Wait','RED','REVERSE');
CALL LEGEND('Market Research System','','RED','BLINKING');
do while(fetch(dsox)=0);
  _ss=_1;
  question=getvarc(dsox,varnum(dsox,'question'));
  do _i=_ss to 11;
    do _j=1 to cxx[_i];
      if question=quest[_i,_j] then do;
        qall[_i,_j]=getvarc(dsox,varnum(dsox,'response'));
        _j=cxx[_i];
      end;
    end;
  end;
```

Three arrays are created:

- QALL - Response fields per Question
- QUEST - Question Names
- CXX - The number of possible responses to each question.

The survey data file contains these variables:

```
BASECODE
VISTDATE
QUESTION
RESPONSE
```

A where statement is used to point to the survey data for the current basecode and visit date. As each of the visitor's records are read, the question variable is compared with the QUEST array elements. When a match

occurs the QALL array (screen fields) are updated.

Cursor Controls

Since our users are communicating with the mainframe through either Token Ring, Coax, or via Dial-In modems, it is best to offer more than one method of navigation about the screen. Therefore, we have included two Function key definitions to move the cursor from one question block to another.

```
array colmap{7} 2 (1,14,44,2,26,43,56);
array colmap{7} 2 (13,43,55,24,41,55,76);
array rowlap{7} 2 (1,1,1,11,11,11,11);
array rowmap{7} 2 (9,9,9,20,20,20,20);

main:
if _status_="E" & _status_="C" then do;
do _i = 1 to 7;
  if rowlap[_i]<=winfo('cursorrow')<=rowmap[_i]
    & colmap[_i]<=winfo('cursorcol')<=colmap[_i] then do;
    curcur=_i;
    _i=7;
  end;
end;
if upcase(word(1))='NEXTFIELD' then do;
select curcur;
when(1) cursor reghn1;
when(2) cursor ym;
when(3) cursor paper;
when(4) cursor family;
when(5) cursor teens;
when(6) cursor z;
when(7) do;
  cursor h1;
  curcur=1;
end;
otherwise;
end;
```

Two function keys are defined: **NEXTFIELD** and **PREVFIELD**. The four arrays are the starting and ending locations in terms of rows and columns. The first step in the main program is to determine the location of the cursor. In the event that the Function key has been pressed then the cursor is relocated.

Multiple Responses

In most cases the value assigned to selected options are controlled by the field attributes.

```

BUILD: DISPLAY WISURDOC.PROGRAM (F)
Command ==>
Command ==>

Use the scroll commands or function keys to review the fields.

Field name: H1      Frame: 1  Row: 2  Col: 1  Length: 7

Alias: H1      Choice group: NLX1  NO  Pad:
Type: CHAR      Protect: YES
Format:          Just: RIGHT  CENTER NONE
Inform:
Error color: RED      attr: REVERSE  Help: .....

List: A

Initial: 0-60
Replace:
Options: CURSOR REQUIRED  NOPROMPT NON-DISPLAY

00020  SMOREMO      F3 - End      F12- Next Question
00021
00022
00023
00024
00025
00026
00027
00028

```

Field Attributes Screen

In the case of three information sources (Newspaper, Radio and Realtor) choices may vary and the user may make additions to the choice tables. To handle these situations a method was developed.

```

if modified(n5x2) then
call method('other.scl','select',
n5x2,n5r2,n5h2,'B','info','radios','Radio Station');

set:
method invar holdvar hvalue subval $4;
holdvar=invar;
hvalue=invar;
invar=subval;
endmethod;
select:
method invar holdvar hvalue subval $4 library member $8 prompt
$12;
if (invar=_BLANK_ & holdvar=_BLANK_) then do;
holdvar=_BLANK_;
hvalue=holdvar;
end;
else if (invar=_BLANK_ & holdvar=_BLANK_) then do;
if hvalue='' then
call
display('info.set.program',holdvar,library,member,prompt);
if holdvar='' then holdvar=subval;
hvalue=holdvar;
end;
endmethod;

```

Assigning Unique Codes to Response Items

When a new Radio Station, Newspaper, Real Estate Agent or Realtor is added by the user the following code is executed to add a unique code to represent the new label. A Table is built containing the alpha and numeric character locations in the standard collating sequence. We find the number of records in the format file and the system then generates a unique id in sequence using the collate function.

```
lastcode=attrn(list,'nobs');
labx=label;
do until(used=-1);
  if lastcode<37 then do;
    sx=lastcode+1;
    start=collate(table(sx),,1);
    code=start;
  end;
  else do;
    if lastcode<62 then lastcode=62;
    _dig1=int(lastcode/61)+1;
    sx=lastcode-(int(lastcode/61)*61);
    if 1<=sx<=62 then
      start=collate(table(_dig1),,1) || collate(table(sx),,1);
    end;
    code=start;
    wx=start || trim(start) || ' ';
    rc=where(list,wx);
    used=fetch(list);
    lastcode+1;
  end;
label=labx;
rc=append(list);
rc=unlock(list);
submit continue;
proc format ddname=library cntlin=&memname;
run;
endsubmit;
```

After the code is generated the file is updated and PROC FORMAT is executed to update the Format in question.

Storing Data in the Survey File

The Procedure for retrieving data is reversed to store data in the Survey data member. Notice that each *live* window field is checked and if a record exists in the survey file but is blank, it is deleted.

```
CALL WREGION(8,20,4,50,' ');
CALL PUTLEGEND(1,'Updating the Survey File','RED','REVERSE');
CALL PUTLEGEND(2,'Please Wait','RED','REVERSE');
CALL LEGEND('Market Research System','','RED','BLINKING');
do _i = 1 to 11;
  do _j = 1 to cxx(_i);
    wx2='and question="' || trim(quest(_i,_j)) || '"';
    rc=where(dsox,wx,wx1,wx2);
    if rc=0 | rc=-580015 then do;
      if qall(_i,_j)= BLANK_ then do;
        if fetch(dsox)=0 then rc=delobs(dsox);
      end;
    else do;
      if fetch(dsox)=0 then do;
        response=qall(_i,_j);
        question=quest(_i,_j);
        rc=update(dsox);
      end;
      else do;
        question=quest(_i,_j);
        response=qall(_i,_j);
        rc=append(dsox);
      end;
    end;
  end;
end;
CALL ENDLEGEND();
```

Query-By-Example

The main section of the Query-By-Example AF program is identical to the survey recording window with the exception that an Array WX is created to hold up to 4 Where declarations to allow subsetting of the survey file based upon buttons pressed on the screen.

```
poi=0;
do i = 1 to 11;
  subst=1;
  do j = 1 to cxx{i};
    if qall{i,j}=_BLANK_ then do;
      poi+subst;
      subr{poi}=j;
      subq{poi}=i;
      subst=0;
      if poi=5 then do;
        _msg_='Only Four field selections can be made.';
        _status_='R';
        return;
      end;
    else wx{poi}='';
  end;
end;
```

Once all of the choices have been made a temporary dataset "STAT" is created to hold the survey data that falls within the date range given.

```
survx=trim(site) || '.survey';
wy='vistdate between ' || put(strdate,7.)
  || ' and ' || put(enddate,7.);
stat=open('stat','n');
newt=newvar(stat,'basecode','n',8);
newt=newvar(stat,'vistdate','n',8,'Vist Date','mddyy8.',
  'mddyy8.');
```

```
do i=1 to 11;
  do j=1 to cxx{i};
    if quest{i,j}='n3' | quest{i,j}='n4' then do;
      newt=newvar(stat,substr(quest{i,j},1,2),'c',4);
      j=cxx{i};
    end;
    else newt=newvar(stat,quest{i,j},'c',4);
  end;
end;
stat=close(stat);
```

The stat dataset contains one observation for every visit record.

```

Libref: WORK
Dataset: STAT
Variable Length Format      Informat
BASECODE 8
VISTDATE 8 MMDDYY8. MMDDYY8.
N1X1 $4
N1X2 $4
N1X3 $4
N1X4 $4
N1X5 $4
N1X6 $4
N1X7 $4
N1X8 $4
N2X1 $4
N2X2 $4
N2X3 $4

```

The stat file is then updated with the survey data.

```

dsox=open(survx,'i');
rc=where(dsox,wy);
do while(fetch(dsox)=0);
basecode=getvarn(dsox,varnum(dsox,'basecode'));
if basecode=baxcd then do;
if baxcd then do;
call putvarn(stat,varnum(stat,'vistdate'),vistdate);
call putvarn(stat,varnum(stat,'basecode'),baxcd);
if append(stat) then do;
msg=sysmsg();
status='R';
return;
end;
else cr+1;
end;
baxcd=basecode;
end;

question=getvarc(dsox,varnum(dsox,'question'));
vistdate=getvarn(dsox,varnum(dsox,'vistdate'));
response=getvarc(dsox,varnum(dsox,'response'));
call putvarc(stat,varnum(stat,question),response);
end;
legend='Number of Surveys within Date Range==> ' ||

```

Where statements are constructed based on the choices made in the Survey Window.

```

cox='';
cx=cr;
if poi then do;
  do i = 1 to poi;
    if -ix then ix=i;
    do j = 1 to subq{i};
      if qall{subq{i},j}= _BLANK_ then do ;
        if 4<=subq{i}<=5 or subq{i}=8 then
          quest{subq{i},j}=substr(quest{subq{i},1},1,2);
        wx{ix}=trim(wx{ix}) || cox || trim(quest{subq{i},j})
          || '="' || trim(qall{subq{i},j}) || '"';
        cox= ' OR ' ;
      end;
    end;
    if subq{i}=4 & subq{i+1}=5 then do;
      cox= ' or ' ;
      ix=i;
    end;
    else do;
      wx{ix}=trim(wx{ix}) || ' ' ;
      cox= ' and ' ;
      ix=0;
    end;
  end;
end;

```

The STAT dataset is subset based on the where statements.

```

if wx1='' then do;
  rc=delete('temp');
  call new('work.temp','work.stat',0,'N');
  dsnx=open('temp','u');
  call set(dsnx);
  dsox=open('stat','i');
  call set(dsox);
  rc=where(dsox,wx1,wx2,wx3,wx4);
  if rc>0 then do;
    _msg_=sysmsg();
    _status_='R';
    return;
  end;
cx=0;
do while(fetch(dsox)=0);
  cx+1;
  rc=append(dsnx);
end;
legend='Number of Surveys Meeting Criteria==> ' || put(cx,4.);
CALL PUTLEGEND(4,legend,'RED','REVERSE');
call legend('Market Research System','','red','blinking');

```

Using Tables to Create Print Controls

In the Question Name Table (INFOQUE), in addition to the normal format elements, there is a field named ORDER, which is used as a parameter in the field definition in PROC REPORT, or prior to the execution of PROC REPORT as in the example shown.

```
options source source2 mprint;
%macro prtweek(date,pc,file);

data null_;
retain cnt 0;
do until(eof);
set traffic.&file.que end=eof;
cnt+1;
var='var' || left(put(cnt,2.));
call symput(var,start);
ordx='ord' || left(put(cnt,2.));
call symput(ordx,order);
call symput('last',put(cnt,2.));
end;
cnt+1;
var='var' || left(put(cnt,2.));
call symput(var,'99');
ordx='ord' || left(put(cnt,2.));
call symput(ordx,'internal');
call symput('last',put(cnt,2.));
stop;
run;
%info.week;
data rep;
length totsuv $67 ord2 8;
merge &file..wtd
      &file..mtd
      &file..ytd;
by variable vx;
order=input(scan(variable,1,'.abcdefg'),2.);
if v1=int(v1) then order=order+(v1-int(v1));
order=order*100 ;
ord2=0;
%do i = 1 %to &last;
  if v1=&&var&i and "%trim(&&ord&i)" = "FREQ" then
    ord2=year;
%end;
if v1=99 then do;
totsuv= 'Total Number of Surveys' ||
        put(week,6.) ||
        put(month,6.) ||
        put(year,6.);
call symput('tots',totsuv);
end;
else output;
run;
proc sort;by order descending ord2;run;
```

The macro is passed the date for reporting, the Info Center initials (PC) and the file name containing the particular center's data, e.g. INFO for the Woodland's Info center. The first data step reads the question label table (INFOQUE) and creates a macro variable for each question establishing the variable name and the type of order for printing the results. The macro &file.week (INFOWEEK) updates three files which contain cumulative statistics for: months (INFOMTD); year (INFOYTD); and weeks (INFOWTD). These are merged together and the print order is set.