

Using SAS/AF® Software and SAS® Screen Control Language to Create an EIS

Nancy Carley, The Principal Financial Group®

Introduction

To meet the challenge of effectively and efficiently managing a field force of approximately 200 sales reps, the Group & Pension Sales department at The Principal Financial Group® developed an Executive Information System. The system delivers relevant and timely information to corporate office management in the form of on-line graphs. The graphs serve as a management tool by illustrating various measures and comparisons of sales for all of the sales reps and offices located across the country.

System Needs

A customized information system was developed by combining the power of Base SAS® and SAS/Graph® software to manipulate and present data with the application development capabilities of SAS/AF® software and SAS® Screen Control Language (SCL). This system meets the following needs:

User-friendly interface: Since the users of the system are sales managers with little time for computer training, the system must be easy to use.

Smooth screen navigation: The users need to access the information quickly without having to maneuver through several screens to get to it.

Batch printing: The system currently does not extend to the field offices, so it is necessary to have the capability to print the graphs to be distributed to the offices. This allows the sales reps to evaluate their own performance based on the information on the graphs.

Easy modification: The flexibility to add new graphs to the system without making a lot of program changes is also necessary.

Maintenance free menus: The information on the menus for selecting sales reps and offices changes every time a new rep is hired or an office is transferred to another manager. Since this happens frequently, it is essential to have these menus updated automatically by linking them to other data files.

Interactive generation of graphs: The graphs need to be created on demand versus displaying preformatted graphic images on request. This is imperative as the initial set of 8 basic graphs yields over 2000 different graphs when a set is created for each sales rep and office. This approach eliminates the initial amount of time required to create all of the possible graphs. It also reduces the amount of storage space the system requires as the graph data can be stored instead of the graphic images.

System Specifications

The PC based system operates on a Novell® network and was developed in OS/2 SAS. The file server and workstations contain 486 processors and the workstations have at least 8 meg of RAM.

System Overview

The system data is updated monthly, processed, and distributed to the appropriate people within an hour after being received. This process is explained in the following steps.

1. On the first day of the month, summarized sales data is downloaded from the mainframe to a PC.
2. This data and some general information from a PC database such as name and title are read into SAS.
3. The data is merged together and manipulated into useful information in SAS. By preprocessing the data at this point, the amount of time to generate a graph on demand is greatly reduced.
4. Data sets for the graphs and menus are output for the users to access via the network. Each user only receives the information that pertains to the user's own sales reps and offices.

The menu data sets contain the updated information for the selection lists on the rep and office menus. By linking to these data sets, the menus are updated automatically.

5. Once the current data is distributed, the user interface designed in SAS/AF software and SAS Screen Control Language accesses the updated data sets and allows the users to generate graphs by choosing selections from a menu.

It is important to note that the interface is a generic shell that contains exactly the same code for all users. The information in the data sets that the users receive is the key to making the application unique to them. This technique makes it practical to add new graphs or make other enhancements to the system as the changes can be made once and implemented by distributing a revised copy of the shell to all users.

User Interface

The remainder of the sections will focus on the user interface that was designed in SAS/AF software and SAS Screen Control Language. For this application, SAS/AF software was used to design the screens and SAS Screen Control Language was used to develop the programs that drive the screens. The key features of these two products for the system are described below.

Block Menus: Display choices as icons in a user-friendly environment where the user simply clicks with the mouse on the desired option.

Extended Tables: Display a custom selection list based on the values in a SAS data set, an array in a SCL program, or an external file.

Choice Groups: Allow users to make one selection from a choice group. Then the appropriate SCL code for that choice is executed.

Submit Blocks: Enable your applications to generate SAS statements that are submitted to the SAS System for processing.

SAS/AF Environment

SAS/AF applications are collections of entries stored in one or more SAS catalogs. A permanent SAS catalog can be created by executing the following statements.

```
libname eis "m:\sasdata\eis";
proc build catalog=eis.menus;
run;
```

All of the entries are stored in the catalog and are listed in the catalog directory as shown below.

Name	Type	Description	Updated
graphs	program	graphs.program	2/18/92
main	program	main.program	2/18/92
office	program	office.program	2/18/92
region	program	region.program	2/18/92
rep	program	rep.program	2/18/92

There are different types of entries such as CBT, HELP, MENU, and PROGRAM entries. This application only uses PROGRAM entries. PROGRAM entries are the entries that contain SCL source programs.

By selecting an entry from the catalog directory, the screen layout can be designed, attributes can be assigned, and SCL code can be entered. There are two different types of attributes that can be assigned: general attributes and field attributes.

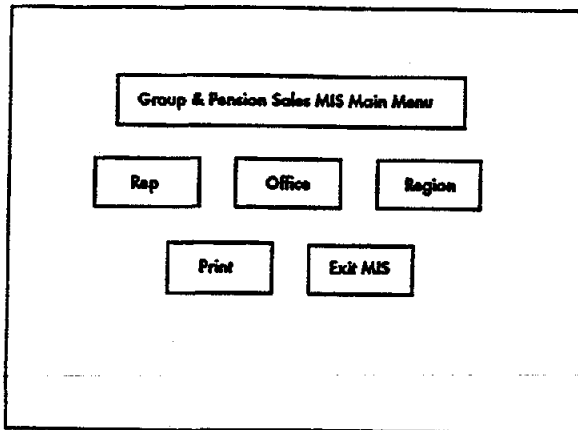
General Attributes: General attributes apply to the entire entry. This includes features such as window size and position, function key information, and window type.

Field Attributes: Field attributes are assigned for each field in the entry. Some examples are field type, format, and justification.

SAS Screen Control Language Code Examples

This section contains examples of some screens and the corresponding SCL code. The first example illustrates a block menu along with the code that is used to create it. The next sample shows the code that is used with the extended tables feature to automatically update the selections on a menu based on the values in a SAS data set. The last example illustrates how choice groups can be used with submit blocks so the appropriate SAS code can be submitted to the SAS System based on a user's response.

Example 1. Main Menu



INIT:

return;

MAIN:

```

choice=1;
choice=block('MIS','Gp & Pen Sales MIS Main
Menu',17,
  'Region','Office','Rep','',
  'Print','Exit','',
  '','','');

```

```

select(choice);
  when (1) call display('region.program');
  when (2) call display('office.program');
  when (3) call display('rep.program');
  when (5) call display('print.program');
  when (6) call display('exit.program');
  otherwise;
end;

```

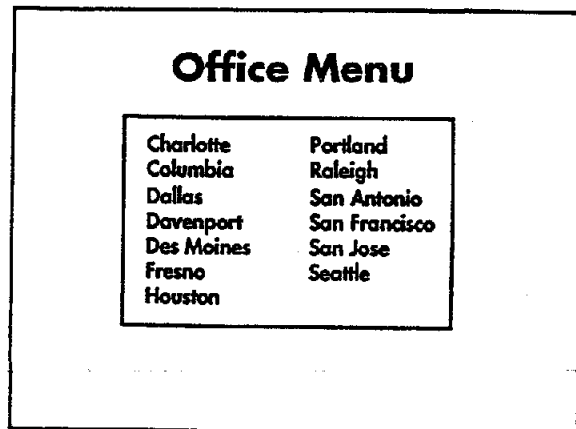
```

call endblock();
return;

```

TERM:
return;

Example 2. Office Menu



INIT:

```

offid=open('eis.offmenu');
call set(offid);
numobs=attrn(offid,'nobs');
call setrow(numobs,1);

```

return;

MAIN:

return;

TERM:

```

if offid then rc=close(offid);
return;

```

GETROW:

```

rc=fetchobs(offid,_currow_);
return;

```

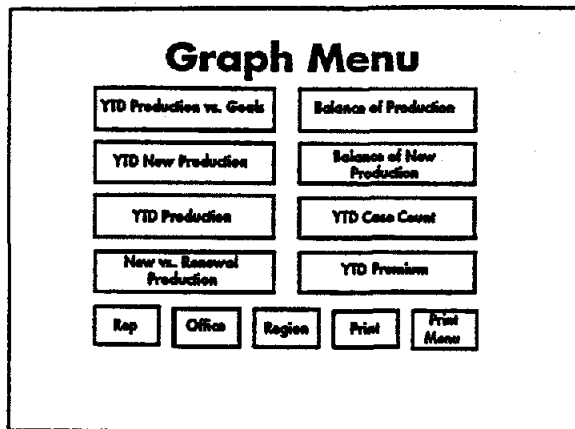
PUTROW:

```

rc=fetchobs(offid,_currow_);
wheresel="grpoff="||grpoff;
dataset="off";
rc=unselect(_currow_);
if offid then rc=close(offid);
call display('graphs.program,wheresel,dataset);
_status_='H';
return;

```

Example 3. Graph Menu



entry wheresel dataset \$ 20;

INIT:

return;

MAIN:

select(graphsel);

when('PYD') do;

submit continue;

Title1 height=1 "Total YTD PC's &
Goals by Product";

Legend label=(height=1)

value=(height=1);

proc gchart data=eis.pyd&dataset;

where &wheresel;

axis1 label=(height=1 "PC's")

value=(height=1);

axis2 label=none value=none;

axis3 label=none value=(height=1)

nobrackets;

vbar yr /sumvar=pcs

group=product

subgroup=year

sum

frame

legend=legend

raxis=axis1

maxis=axis2

gaxis=axis3;

run;

quit;

endsubmit;

end;

otherwise;

end;

return;

TERM:

return;

Future Plans

Future plans for the system include adding ad hoc graphing and reporting capabilities as well as exception reporting, linking to mainframe data, and extending the system to the field offices. SAS/EIS[®] software will also be explored with the possibility of integrating it with the current system.

SAS, SAS/Graph, SAS/AF, and SAS/EIS are registered trademarks of SAS Institute Inc. in the USA and other countries. [®] indicates USA registration.

Contact Information

Nancy Carley

The Principal Financial Group

Des Moines, IA 50392-3900

Phone: (515) 248-8773

Fax: (515) 248-3519