Building an Executive Information System (EIS) without SAS/EIS™ Software

Stephen Davidson, Dartmouth Medical School

Gregory A. Smith, SAS Consulting Services Inc.

The SAS System is a powerful program that has been used for years to develop applications of all types and sizes. At SUGI 16, Dr. Goodnight announced a new and exciting addition to the wide range of products already available, SAS/EIS software.

An EIS or Executive Information System is meant to provide an intuitive interface to a company's key information and make that information easily understood. Furthermore, access to the EIS should not be limited to upper management, but should be available to anyone within the company who needs access to corporate data in order to perform their job better and more efficiently. A shop foreman should be able to access data from a part defect database as easily as a vice president can access sensitive financial data. An EIS should provide each with access to these different yet complimentary types of data.

The SAS/EIS product provides a set of tools that allow the developer to create an EIS, complete with many of the most popular EIS features. However, by using the existing SAS/AF product, the developer is able to simulate some of these same features, without the SAS/EIS product. Important EIS features like graphical drill-down, CSFs (Critical Success Factors) and hot-spotting can be accomplished with SAS/AF and SAS/GRAPH software. This paper will discuss the topic of providing EIS functionality without the SAS/EIS product by utilizing these EIS simulation techniques.

Graphical Drill-Down

Drill-down is a way of presenting information to the user and allowing them to narrow their focus from a general or higher level of aggregation to a more specific or lower level of detail. Graphical drill-down provides this functionality by presenting the information in the form of a graph or chart. A user is able to examine a graph and, using this technique, drill-down to examine the underlying data.

Selection List

To demonstrate how to accomplish graphical drill-down* using the SAS/AF product, we have developed an example based on maps of the United States. In our example, we performed a graphical drill-down from the highest aggregation, the U.S. map, to a specific state in the Northeast U.S. Our maps were taken from the US and STATES maps from the SAS/GRAPH map database. The user's possible drill-down choices are a limited list of regions in the U.S. Figure 1 shows our example of graphical drill-down. At the top of the figure is the top level graph and is accompanying selection list. The graph is displayed a graph window that has been resized to allow room for a selection list to be displayed next to it. The selection list is an extended table, built using the SAS/AF product, displaying a list of regions of the United States stored in the SYSTEM.FAMILY SAS dataset. Based on the user's selection, made by highlighting and selecting the desired row in the extended table, another graph and selection list are displayed. By reading the FAMILY database, the system determines both the appropriate graph to display and the possible drill-down values to be displayed in the selection list. This is an iterative process, each time the user makes a selection a new graph and selection list are displayed. At the "bottom" of the drill-down, the point at which lower-level graphs do not exist, the system only allows the user to see a table of the underlying data for the last graph, i.e. "the numbers behind the picture". As shown in our example in figure 1, the user has chosen the region named nor-eas and then Vermont from the selection lists. After each selection, a new graph and selection list are displayed. Finally, at the bottom of the drill-down, the user selects the display report option and is shown the underlying data. This is done using the FSVIEW procedure in the SAS/AF product.

1. This technique of graphical drill-down only works on graphic devices that support the graph manager or graph windows.
Figure 1

1457
InH:

Open Control DataSet

Parent/Child Relationships

diskopen("system.family");
call set(disk);
call fill(disk);
call rewind(disk);

Draw associated Parent Graph

call exec(cmd("graph1 system.maps.""llparlgph") ;

List Children in Extended Table

call setrow(0,1,'N','V');

Return;

Getrow:
if (fetchobs(disk, _currow) ne 0 then
call endtable();

Return;

Putrow:
Return;

Main:
if modified(rpt) then call review("system.family");
if _status_ not in('E','e') and child ne _blank_ then do;
call display(at program,child); 

ReDraw associated Parent Graph

call exec(cmd("graph1 system.maps.""llparlgph") ;
end;

Return;

Figure 2

FSP® product in our example, but it could be useful information displayed in any format. In designing our drill-down, we could have chosen to select the option to show us the "numbers" at each level if desired. Had we chosen to do this at the non-eas level, relevant data for that aggregation of states could have been displayed. Figure 2 shows the SAS/AF SCL program for the iterative drill-down part of our example.

Hot Spotting

A key requirement for an Executive Information System is ease of use. An EIS not only needs to provide access to the most important data, but must allow the user to get to the data quickly, without having to think about how. To meet this requirement, most EIS systems provide some sort of "point and click" interface. In PROGRAM entries, developers can use choice groups and push-buttons to provide this type of an interface. However, in addition to the more commonly used PROGRAM entries, SAS/AF software provides another type of entry that can be useful to the EIS developer. Although the CBT entry was originally developed for Computer Based Training, it can be used very effectively as an EIS tool.

Like PROGRAM entries, the CBT entry allows the developer to "paint" a screen and write code to handle user interaction. However, the CBT entry uses its own, limited, programming statements which do not provide the wide range of functionality Screen Control Language can provide. One of the most useful features of the CBT entry is the ability to define portions of the display as "hot spots" and branch to other CBT or PROGRAM entries when a user "clicks" on the "hot spot" with a mouse or moves the cursor to area and hits the ENTER key. This feature can be used to create complex menus or hypertext-like documents. Another example of how to use this feature is to display a diagram of a process and allow the user to select the stage of the process that is of interest. Consider an EIS being developed for the ACME CORPORATION. Their Manufacturing Accident Incident Management System, or MAIMS, allows executives to monitor accidents on the shop floor. Figure 3 shows the CBT entry screen that serves as the main menu for MAIMS system. It is a diagram, drawn with text characters and reverse video, of the shop floor of the ACME manufacturing plant. Each stage of the manufacturing process is shown in its approximate location on the plant floor. ACME executives can move their cursor to any area on the diagram, select it and review the accident statistics for that part of the plant. Figure 4 shows the CBT programming statements needed to make the menu work. Each SELECT statement defines a rectangular region on the display and indicates what action to take when the user selects it. By overlaying these regions it is possible to create shapes more complicated than a rectangle. In this example, each "hot spot" region branches to a PROGRAM entry that handles the display of accident data. If ACME's manufacturing process were more complicated, the MAIMS system could use several CBT entries to...
Critical Success Factors (CSFs)

In addition to providing executives with user-friendly access to an organization's most important data, an EIS should present the data in an easily digestible form. One popular form is used to show the current value of a variable within a range. Looking much like a car's speedometer, these pictograms allow executives to check the level of important indicators. Grouping several of these pictograms on a screen, a developer can create an "Executive Dashboard", displaying all of a company's Critical Success Factors (CSFs) and allowing a busy executive to quickly scan the screen and determine the current situation. The executive should also be able to "click" on a CSF and explore the underlying data.

Once again, SAS/AF software provides this functionality with the CBT entries. In addition to the ability to define "hot spots", CBT entries allow a developer to display a graph on a CBT's display. Combining these two features, the developer can easily produce an "Executive Dashboard". Developing the actual CSF pictograms can be done using SAS/GRAPH software. Although ANNOTATE data sets could be used, the Data Step Graphics Interface, DSGI, is particularly well suited for this task. A CSF is usually depicted as a half circle divided into three sections. The leftmost section, usually colored red, indicates values at a "dangerous" or "unacceptable" level. The middle section, usually colored yellow, indicates values that, although not presently in the "danger zone", bear close monitoring. The rightmost section, usually colored green, indicates values at "acceptable" or "good" levels. The variables current value is shown as a needle, positioned at the appropriate point within the range. Figure 5 shows an example of a CSF and

Figure 6 shows the SAS Data Step using the DSGI that was used

data _null_

redline = 6240;
yellowline = 11250;
maxvalue = 17300;
value = 14450;
redline = (redline/maxvalue)*180;
yellowline = ((yellowline/maxvalue)*180); 
nend = (value/maxvalue)*180 + 1;
ncatalog , 'SYSTEM', 'WINDOWS');
rc=graph('CLEAR', 'ONE');
rc=gsel('FILTYPE', 'SOLID');
rc=set('FILCOLOR', 4);
rndraw('PIE', 110,50,35,yellowline);
rc=set('FILCOLOR', 6);
rndraw('PIE',110,50,35,redline);
rc=gsel('FILCOLOR', 2);
rndraw('PIE',70,50,35,redline);
rc=set('FILCOLOR', 1);
rc=gdraw('PIE',110,50,40,nstart,nend);
rc=gsel('TEXFONT', 'OP');
rc=gsel('TEXALIGN', 'CENTER', 'OP');
rndraw('TEXT', 110,45, '12450');
rc=gsel('TEXFONT', 'SIMPLEX');
rc=gsel('TEXHEIGHT', 'OP');
rc=gdraw('TEXT', 70,45, 'in thousands of dollars');
rc=gsel('TEXALIGN', 'LEFT', 'OP');
rc=gdraw('TEXT', 70,50, 0); 
rc=gset('TEXFONT', 'SYSTEM', 'OP');
rc=gdraw('TEXT', 150,50, '15,000');
nend;
run;

Figure 6
to create it. Figure 7 shows how several of CSFs are combined in

![Executive Dashboard](image)

Figure 7

a CBT entry to create an "Executive Dashboard" and Figure 8 shows the CBT programming statements that display the CSFs

![CBT Programming Statements](image)

Figure 8

and define the "hot spots" that will allow the user to "click" on a specific CSF and see the underlying data. Once again, all of our hot spots branch to PROGRAM statements, although they could branch to other CBT entries; perhaps to CBT's that display BAR CHARTS, PIE CHARTS or some other SAS/GRAPH output. Since an EIS is expected to allow executives to access the most recent data, any graphs used in the EIS should be recreated periodically. This could be easily accomplished in a batch job that is run each night and regenerates the EIS's graphs using the most current data.

Summary
The SAS System allows developers to create complex and impressive applications. The power of the SAS System has been enhanced with the introduction of SAS/EIS software. However, for those developers who are not able to have access to SAS/EIS software, SAS/AF and SAS/GRAPH software provides a means to simulate some of that EIS functionality. This paper has presented several techniques that can be used by SAS applications developers to create impressive SAS solutions for potential EIS users.

SAS, SAS/AF, SAS/EIS, SAS/FSF, and SAS/GRAPH are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration.