Paper 25-25

Empowering Your End-Users' Reporting Potential from SAS/AF® Applications with SAS Enterprise Reporter™

Gary Mason, Qualex Consulting Services, Inc., Apex, North Carolina

ABSTRACT

Empower the end-user, by making SAS Institute's, Enterprise Reporter™ (ER), a reporting tool option in your SAS/AF® application. This gives the user the opportunity to produce presentation quality reports. For ease of access to SAS ER in your SAS/AF® application, add it as an additional toolbar icon. This paper will illustrate that as the icon invokes ER, the layout allows the user to create an ad-hoc report. The SAS/AF® application passes the SAS dataset into the ER Design View which allows the end-user to immediately begin producing their own high quality reports.

INTRODUCTION

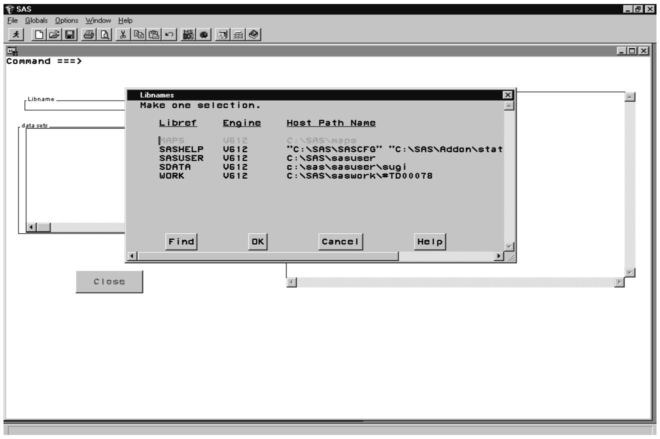
For nearly 25 years, SAS Institute has provided developers and end-users with many tools to present their data as output. From tools like PUT statements, PROC PRINT, PROC REPORT, and now Enterprise Reporter (ER). SAS ER is a full-featured report generator. It is capable of being used from ad-hoc reporting to creating Standard Annual Reports. The graphical features of ER makes it much more robust than Proc Report and much easier to use.

As a developer, you can put the reporting power in your end-user hands. With ER, the end-user can easily create a table, graph and/or listing report. The end-user has the tools necessary to easily add many customizations to the reports, which would otherwise have required hours requesting and explaining to a programmer. If a end-user knows how to point and click, drag and drop, ER is the tool for them. Features of ER that are available for the end-user include a cover page, a variety of graphics, tables, page headers, logos, titles and many more. Consult the Enterprise Reporter documentation or visit the Web site for all the latest features. This paper will concentrate on the interface needed to integrate this reporting tool into an existing or new SAS/AF® application.

GETTING RIGHT TO IT

This paper includes the SAS/SCL code needed to insert the toolbar icon. After a simple click, on the toolbar icon, ER starts and passes the SAS dataset the user has selected from this application.

In this sample SAS/AF® program, the end-user selects a SAS Libname from the Libname selection box by clicking the Down arrow and selecting the SDATA libname. (Figure 1)This sample library contains datasets for a fictitious company that has stores in several North Carolina counties. These datasets contain sales information for several months. The data has been summarized to the County and Store ID level. With this level of data the Number of Customers and Sales in Dollar amount can be analyzed. In this example, an end-user could analyze the sales for these six counties quickly. But, for a moment, imagine 60,000 rows of data, and an end-user looking for trends using a PROC REPORT table.

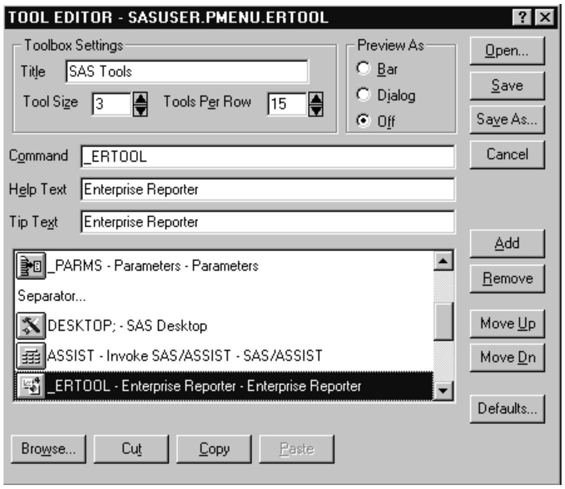


```
Make one selection.

    Libref      Engine    Host Path Name

    MAPS        V612      C:\SAS\maps
    SASHELP     V612      "C:\SAS\SASCFG" "C:\SAS\Addon\stat
    SASUSER     V612      C:\SAS\sasuser
    SDATA       V612      c:\sas\sasuser\sugi
    WORK        V612      C:\SAS\saswork\#TD00078
```

Figure 1.

This is where the link to SAS Enterprise Reporter is invaluable for you and the end-user. When ER is added to the SAS reporting toolset, PROC PRINT and PROC REPORT, a developer can greatly enhance the end-users ability to efficiently analyze their data.

First, in order to add the ER icon into the toolbar, the entries in the PMENU Catalog, PMENU and TOOLBAR need to be modified. In Figure 2, after the ASSIST toolbar icon the ER icon is added. The _ERTOOL command is executed, when the end-user clicks the icon, with the ENTERPRISE REPORTER label.

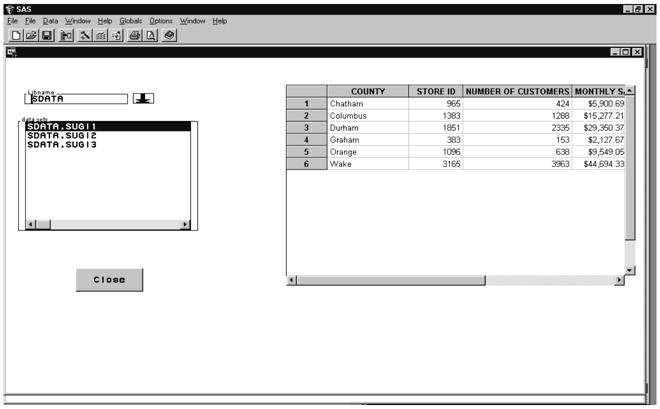


Title: SAS Tools
Tool Size: 3
Tools Per Row: 15
Preview As: ● Off
Command: _ERTOOL
Help Text: Enterprise Reporter
Tip Text: Enterprise Reporter

_PARMS - Parameters - Parameters
Separator...
DESKTOP; - SAS Desktop
ASSIST - Invoke SAS/ASSIST - SAS/ASSIST
_ERTOOL - Enterprise Reporter - Enterprise Reporter

Figure 2.

For the purpose of this paper, there are three entries in the SUGI catalog that drive this example: MAIN.FRAME, MAIN.SCL, and EREPORTR.SCL. MAIN.FRAME contains the visual objects that control the selection and data table view of the chosen SAS dataset. MAIN.SCL is where the SCL code that controls these objects and handles the TOOLBAR selection for the _ERTOOL command. The code for the two SCL objects is listed at the end of this paper.



Libname: SDATA

data sets:
SDATA.SUGI1
SDATA.SUGI2
SDATA.SUGI3

	COUNTY	STORE ID	NUMBER OF CUSTOMERS	MONTHLY S
1	Chatham	965	424	$5,900.69
2	Columbus	1383	1288	$15,277.21
3	Durham	1851	2335	$29,350.37
4	Graham	383	153	$2,127.67
5	Orange	1096	638	$9,549.05
6	Wake	3165	3963	$44,694.33

Figure 3.

The label in MAIN.SCL named SEL1 handles the selection of the SAS library from the allocated libraries for this SAS session. By clicking the DOWN Control object SCL runs the SEL1 label which pops a LIBLST window allowing the end-user to select a SAS library from the frame. Figure 3 shows how choosing the SDATA Library the code instantiates the SAS Library object and initializes it with information on the library. The SDATA library with members of type DATA populate the SCL list TLST and the DO loop formats and loads the DSETS List Box with the members of the SDATA library.

The DSETS label executes when the end-user selects the SAS dataset from the List Box. The object gets all the path information and sends the DATA TABLE object, TABLE, the dataset that was selected. It also sets the PMENUS and toolbars with the ER toolbar icon.

With the data table populated, the end-user can either choose another SAS dataset or select the Enterprise Reporter icon from the toolbar. The MAIN label traps the _ERTOOL command and links to the ER label that calls a method to the EREPORTR.SCL passing it information on the SAS dataset selected.

Control is passed to the EREPORTR.SCL method that assigns default values to several variables. It is assumed that Enterprise Reporter is installed locally with the C:\Enterprise Reporter folder present. ER is invoked with the
`call display('SASHELP.BUSINESS.START.SCL');.`

When the ER Design View comes up, as in Figure 4, the user has their dataset passed to ER's Report Data window with all the variables in that dataset. This is where the user can now begin to build their report. The left side, of the Enterprise Reporter window, contains the toolbars for the different objects the user can place upon the report. From the top, the first three objects are Master, List, and Table. This paper only briefly discusses the Table and the Bar Chart objects.

Objects are placed upon the report by clicking one of the tools to drag and draw the outline onto the report. The Report, once this outline is placed on the report, the user can select, from the Report Data window(Figure 4), the columns to populate the Table object with. Once this is done, the user can press the Data View icon in the bottom left and ER fills the Table object with the data from the selected dataset. The labels and formats are retained and summary data is also presented. Adding totals and subtotals is just a point and click away. Select the column(s) to add totals to, and click the “Grand Totals at the Bottom” icon in the top toolbar. Totals are added to the bottom of the report.

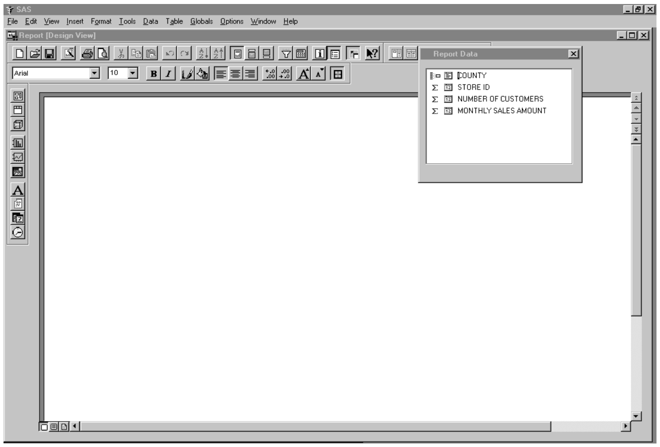


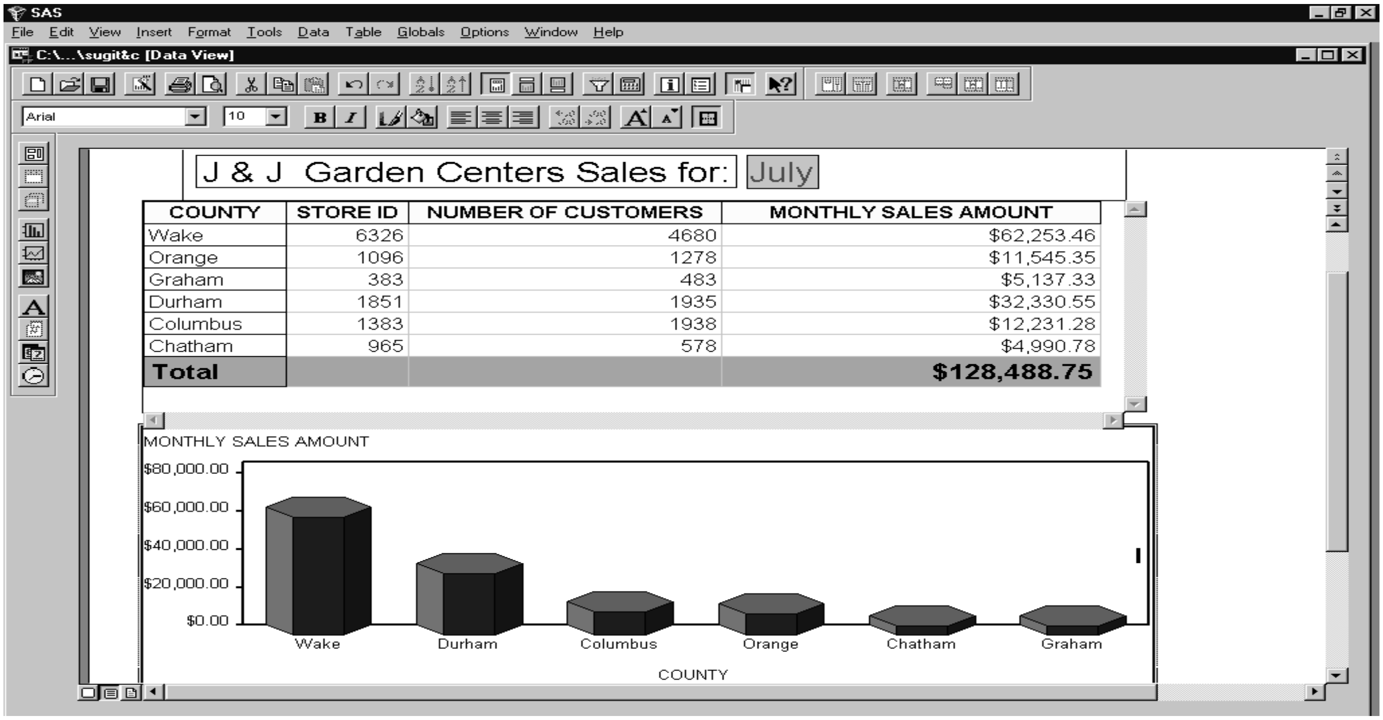
Figure 4.

Adding a Bar Chart is just as simple. Click the Bar Chart Icon on the left toolbar and Drag and Draw the area. The end-user can then select the x-axis and y-axis variables by selecting the variables from the Report Data window and dragging it to the location on the Bar Chart Object. One possible finished report may look like the frame in Figure 5.

See the Enterprise Reporter Documentation for the numerous other objects available for enhancing the report or creating HTML also.

CONCLUSION

SAS Enterprise Reporter is a great enhancement to the set of reporting tools already available from the SAS System. ER is a superior choice for developers wanting to enable the end-users to perform customization and output high-end reports. Easily incorporating ER as a point and click option for the end-users, is a timesaving to both developers and the end-users. Increased productivity is achieved with the ability to customize the invocation of Enterprise Reporter tailored to the needs of the end-user and their data.



J & J Garden Centers Sales for: July

COUNTY	STORE ID	NUMBER OF CUSTOMERS	MONTHLY SALES AMOUNT
Wake	6326	4680	$62,253.46
Orange	1096	1278	$11,545.35
Graham	383	483	$5,137.33
Durham	1851	1935	$32,330.55
Columbus	1383	1938	$12,231.28
Chatham	965	578	$4,990.78
Total			**$128,488.75**

Figure 5.

SCL CODE

```
/*************************************************
************/
/***   MAIN.SCL
***/
/***   SUGI Mock up to demonstrate ER with the
toolbar    ***/
/*************************************************
************/

length sel         $8
       dset        $8
       libs        $100
       libpth      $100
       libname     $22
       ;


rc=rc;
init:
 control always;
 CALL EXECCMDI('COMMAND CLOSE;');
return;


LIB:

 link sel1;
return;

SEL1:
if tlst>0 then tlst=dellist(tlst);

libs= liblist();
```

```
CALL NOTIFY ('lib', '_SET_TEXT_',libs );

tlst=makelist();
libclass=loadclass('sashelp.fsp.library.class');
libobj=instance(libclass);

call send(libobj, '_setup_', libs);
CALL SEND(libobj, '_GET_MEMBERS_', tlst,
'MEMTYPE ="DATA"' );

CALL NOTIFY ('dsets', '_DELETE_ALL_');
i=listlen(tlst);

do n=1 to i;                     /** loop
through the list **/
  dsname= getnitemc(getiteml(tlst,n),'MEMNAME');
  libname=
getnitemc(getiteml(tlst,n),'LIBNAME');
  CALL NOTIFY ('dsets',
'_ADD_',trim(libname)||'.'||dsname , -1 );
end;

return;/* SEL1 */

DSETS:
CALL NOTIFY ('DSETS',
'_GET_LAST_SEL_',row,is,libpth);


CALL NOTIFY('TABLE', '_SET_DATASET_', libpth);
CALL NOTIFY ('TABLE', '_DISPLAY_COLUMN_LABEL_',
'_ALL_');

 /***  set the toolbar and Pmenus for the Data
table  ***/

 call
notify('.','_SET_PMENU_','SASUSER.PMENU.ERTOOL.P
MENU');
 toolload='SASUSER.PMENU.ERTOOL';
 call execcmdi('TOOLLOAD WINDOW
'||toolload||';');

return; /* DSETS */

END:                                  /****
close window ****/

link TERM;
return;

TERM:

if tlst>0 then tlst=dellist(tlst);
return;/* TERM */


MAIN:       /**** GET _ERTOOL from pmenu ****/

 cmd=word(1,'U');

 select(cmd);

    when('_ERTOOL') link ER;

 otherwise;

 end; /* end select */

 cmd='';

return; /* MAIN */


ER: method;


 /***     call method that will bring up ER
***/

 call method('EREPORTR.SCL','DISPLAY',' ',
libpth);

endmethod; /* ER */


/************************************************
***************
 EREPORTR.SCL

 Call this to bring up ER on a specific report.

************************************************
**************/
rc=rc;

DISPLAY: method
 optional=
 path $200
 outdsn $30
 user $10
 ;

 if length(path) le 8 then
     path='c:\enterprise\my
reports\'||path||'.erp';

 if user='' then user='ADMIN';

 loc_list = envlist('L');
 cmd_list = getniteml(loc_list,'_CMDLIST_');

 /***     do a savelist before clear
***/
 /***     then restore cmdlist after returning
from ER  ***/

 rc=clearlist(cmd_list);

 /***     Heres the CMD_LIST getting populated
with     ***/
 /***     what will be passed to ER at
invokation       ***/

 cmd_list = setnitemc(cmd_list,outdsn,'DATA',-
1);
 cmd_list = setnitemc(cmd_list,'NO','EXITICON',-
1);
 cmd_list = setnitemc(cmd_list,user,'USER',-1);

 /***     Heres how ER gets invoked with SCL
***/

 call display('SASHELP.BUSINESS.START.SCL');

 /***     ER returns control to SCL
***/

 loc_list = envlist('L');
 CALL EXECCMDI('TOOLload');

endmethod;
```

REFERENCES

SAS Enterprise Reporter™ Software: User and Data Administration, Course Notes, 06Aug98

SAS Enterprise Reporter™ Software: Creating Business Reports, Course Notes, 31Oct97

Getting Started with Enterprise Reporter™ Software ,1998

Yaguda J., Mikkelsen J.D. ,"Extending the Power of Your SAS® System Applications with Enterprise Reporter™ Software" *Proceedings of the Twenty-Fourth Annual SAS Users Group International Conference, Paper 32, page 196*

ACKNOWLEDGMENTS

The author would like to express appreciation to the
The SAS® SAS Consulting Services, SAS Institute Inc.
With special thanks to:
Dave Berger, Senior Applications Developer,
Brian Bennet, Associate Applications Developer

CONTACT INFORMATION

Contact the author at:
Gary Mason
235 Indian Creek Ln
Apex, NC 27502
919 271-7074
gary.mason@qlx.com
Web: www.qlx.com