Extending the Power of Your SAS® System Applications with Enterprise Reporter ™ Software

Juli Yaguda, SAS Institute Inc., Cary, North Carolina
Jens Dahl Mikkelsen, SAS Institute A/S, Copenhagen, Denmark

Abstract

As a seasoned SAS System applications developer, do you need to implement the best exploitation end to your data warehousing strategy? Do you create and maintain customized SAS System applications, but still need to augment its power with a flexible reporting environment? With the evolution of the SAS Enterprise Reporter, users can quickly generate and administer presentation style reports to specified employees or departments, customize access to corporate data and manage the information for business reporting needs. Enterprise Reporter can accomplish these tasks either as a supplement to exploit back-end data or as an integrated component of varied SAS System applications.

Some examples that can benefit from this integration include:

- Compliment your existing SAS/AF® software or SAS/EIS® software applications by allowing the users to perform ad-hoc reporting on the data fed from your application in an environment familiar to the users.
- Let users of your SAS software applications print nicely formatted reports through the power of Enterprise Reporter.
- Exploit the data warehouse metadata from SAS/Warehouse™ Administrator software or your own application for ad hoc reporting using Enterprise Reporter.
- Close the information gap for information consumers by distributing reports from Enterprise Reporter to the web.

Each of the technology functions mentioned above can be accomplished with Enterprise Reporter to help you extend your company's enterprise reporting environment and each will be discussed in this paper.

Introduction to the Technology

Enterprise Reporter is a component of SAS Institute Inc.'s end-to-end Data Warehousing solution, thus exposing a variety of technical avenues to communicate with other SAS System data warehousing product components.

Common scenarios of integrating Enterprise Reporter with other components of the SAS System are indicated below. They illustrate integration options to assist in creating a complete knowledge-based solution for your organization.

1. *Access to informational data sources in the business terms your users understand.* To access the data, you can use the SAS/Warehouse Administrator to automatically define the reporting metadata of the division's data mart. You also have the option of accessing your data warehouse or data store separate from SAS/Warehouse Administrator.

2. *Automatically organize and process the data prior to reporting in accordance with parameters specified interactively.* To pre and post-process data for reporting, you can use a feature that allows you to execute SAS System code and resolve SAS macro variables during the generation of the dynamic report.

3. *Access and print reports directly from the Windows Explorer or from a customized SAS/AF software application.* To launch

a pre-defined report from the Windows Explorer your users need only have novice experience with the Windows 95 or Windows NT interface. You also have the option of direct calls to Enterprise Reporter from your customized SAS/AF and SAS/EIS software application development tools. These tools can be used by your users to specify customized report parameters and as your venue of integration with Enterprise Reporter software.

4. *Use graphs created with SAS/GRAPH® software in your Enterprise Reporter reports.* You can incorporate SAS/GRAPH output into your Enterprise Reporter report either as a static image or as a dynamic set of images.

5. *Create an ad-hoc report after drilling-down to the data of interest from your application.* To create an ad-hoc report after drilling-down to desired subset of data, you can use SAS/EIS software by overriding the methods to pass in the current data and launch the Enterprise Reporter.

6. *Automatically generate reports for later access through your corporate Intranet and through Adobe® Acrobat® software.* To complete the end-to-end reporting solution, the Web Gallery of Enterprise Reporter will be utilized for web publishing of the pre-defined reports.

7. *Learn more about the future of Enterprise Reporter Software.* The direction of Enterprise Reporter software offers continued improvement for providing your users with the information they need to make knowledgeable, well-managed decisions for your organization in the market place.

Each of these common integration scenarios is embellished upon in the seven sections that follow. Additionally, extended detailed information can be found in the appendices.

Scenario 1: Accessing your Data Store with Enterprise Reporter and SAS/Warehouse Administrator Software

One of the most crucial steps in creating an end-to-end reporting solution is to have a method for easily accessing and managing your data store. With Enterprise Reporter, this can be accomplished through *InfoFolders*. InfoFolders contain data items that can be selected for a report. The data items represent various forms of data that you load from local or remote databases and may be customized to fit your users' business terms. InfoFolders contain the reporting metadata. This metadata can be created and managed explicitly from within Enterprise Reporter or it can be exported from the SAS/Warehouse Administrator Software directly into an InfoFolder.

If your data store is managed through SAS/Warehouse Administrator software, you can export the metadata from SAS/Warehouse Administrator to a specific location. The resulting exported data contains two tables utilized by Enterprise Reporter. Specifically, the two table names are COLUMNS and LOCATION.

To read this data into the Enterprise Reporter, you create a new InfoFolder for your report metadata. In the Create InfoFolder window, press the **Add** button. In the *Path* field located in the Add window, you specify the location of the exported metadata by

typing it in or by using the **Browse** facility. Click on the **Proper-ties** button to create the InfoFolder. The created InfoFolder will be based on the information contained in the warehouse folder. Enter-prise Reporter will automatically analyze the contents of the SAS/Warehouse Administrator data to create its metadata. **Figure 1** illustrates the communication flow between SAS/Warehouse Administrator and Enterprise Reporter.
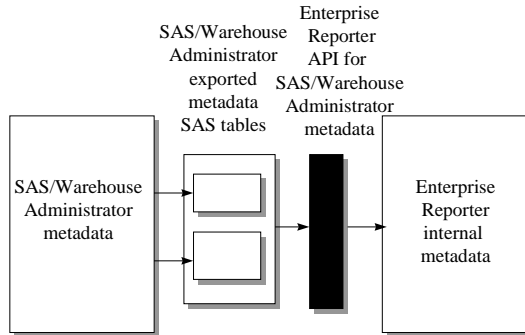

Figure 1: *Communication between Enterprise Reporter software and SAS/Warehouse Administrator software*

It is recommended that SAS/Warehouse Administrator software be used to build the warehouse. However, if you have invested in other warehousing software, Enterprise Reporter software can read any metadata into InfoFolders. You can write a SAS software pro-gram to convert the external data store into a form like the Diction-ary tables found in the SAS System. These tables are TABLES, COLUMNS, and MEMBERS. Additionally, Enterprise Reporter expects a table called USERJOIN that describes how tables are joined with one another. These tables are stored in a location speci-fied in the *Path* field when creating an InfoFolder. In order to do an automatic read of the external data, it is required that it is not mixed with other data sources in the same InfoFolder.

SAS/Warehouse Administrator software can export data to a data store recognized by Enterprise Reporter software. Regardless of how you define Enterprise Reporter InfoFolders, whether through SAS/Warehouse Administrator or by other means, you can decide if you need to further process the data for specific reporting needs.

Scenario 2a: Processing Data Before and After Reporting with SAS System Software

On occasion, you may have reports that require special processing of the data prior to generating and printing them. With Enterprise Reporter, you have the power of the SAS System available by executing any SAS program when an InfoFolder is either selected or is deactivated by the user. Listed below are examples of when you can take advantage of this feature:

- Display a customized window used for sub-setting data
- Perform advanced data manipulation and calculations
- Dynamically generate data dependent SAS formats
- Locate SAS format libraries
- Move data across platforms and join them on a single plat-form
- Delete the temporary data after reporting
- De-allocate resources

You can honor these requests by using the START.SAS and STOP.SAS reserved program names. Place your code in either file

depending upon when you want it to execute. START.SAS exe-cutes when an InfoFolder is selected and STOP.SAS executes when an InfoFolder is deactivated. If you use either of these files, they must be stored in the same folder containing the metadata for the selected InfoFolder. Note these files are executed within an SCL Submit Continue block from Enterprise Reporter. *Appendix C* illustrates an example of using START.SAS to create a Top-10 report.

Scenario 2b: Working with Macro Variables in Enterprise Reporter

Customizing general report information is not an uncommon prac-tice. For example, your users may need to be able to subset infor-mation according to their regional interest and indicate the region in the title of the report. SAS software macro variables can be used to establish these types of customizations. These values, typically defined in Enterprise Reporter Text Objects, are resolved when the report is generated provided they have been previously defined in the SAS system.

A common use of macro variables from within Enterprise Reporter is through its Filter window. For example, you can define a filter for a report to include a range of dates. The date range is specified by a macro variable that can be defined either prior to or during the launch of the report. This can be done by either retrieving one or more values from an existing data source or by allowing the user to fill in a date range using your customized application. An example of how macro variables in the Enterprise Reporter Filter window is indicated in **Figure 2**. Here, &BEGNDTE and &ENDDTE repre-sent the beginning and ending dates to subset in the report.
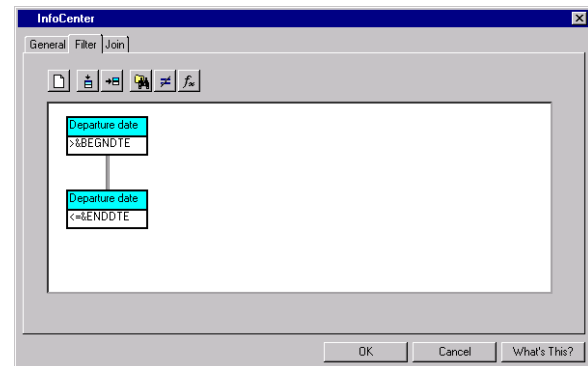

Figure 2: *Filter window with date range of macro variables*

Enterprise Reporter also offers you a set of pre-defined macro variables accessible from the Text Report Object. In Enterprise Reporter report builder, from the object toolbar, select **Text** and draw a region in the report builder. Right-Mouse-Button in the region and select **Insert Field**. The available macro variables that are automatically resolved for you are: *User name, Report name, File name InfoFolder, System time and Author*.

The power of the SAS System is utilized by Enterprise Reporter and as described previously, that same power can be utilized by you as the administrator. Similarly, you can take advantage of this power by creating your own customized applications using SAS Software application development tool set to allow your users to gain access to the reports they need for making intelligent business decisions.

Scenario 3: Integrating Customized Applications with SAS/AF Software

Your SAS/AF software application can benefit from using Enterprise Reporter as an ad-hoc reporting solution for data that your application creates. This functionality allows your users to design their own reports as it relates to data store attributes of interest. You can make this very easy to use since you can customize the Enterprise Reporter environment. For example, you can prompt the user to pre-select the data source in your application and programmatically pass this information to Enterprise Reporter.

Additionally, you can produce a number of standard reports automatically by creating the reports beforehand and allowing your users to select the reports to generate. You can also provide your users with the ability to pass parameters, such as their own customized titles or their criteria to report on subsets of the data.
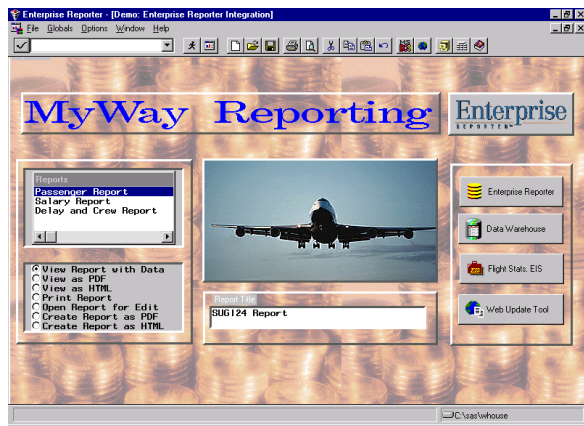

Figure 3: *Example of Customized SAS/AF software application integrating with Enterprise Reporter.*

Figure 3 displays a customized SAS/AF software menu with options to access Enterprise Reporter, SAS/Warehouse Administrator, SAS/EIS multidimensional report object and the Enterprise Reporter Web Gallery tool.

In the above application, users can additionally select a report, subset the data, define a title for the report and launch Enterprise Reporter. Enterprise Reporter, in turn, surfaces the users selections and report criteria and displays the customized report.

The techniques you can use to directly launch Enterprise Reporter from a SAS software application or from the SAS System command line is with an AF or AFA command. With Screen Control Language (SCL) you can use the CALL DISPLAY routine.

To invoke Enterprise Reporter from a command line, enter the following command:

AF C= sashelp.business.start.scl

To invoke Enterprise Reporter from an SCL statement, use the following call in your SCL program:

CALL DISPLAY('sashelp.business.start.scl');

Note that in order for this to work, Enterprise Reporter needs to be included in the search paths of the CONFIG.SAS file used to run the SAS System and Enterprise Reporter.

Enterprise Reporter's executable, ERSAS.EXE, contains the necessary instructions to set up the Enterprise Reporter environment. After it finishes with its instructions, it invokes SAS.EXE from the location specified in your CONFIG.SAS file.

Enterprise Reporter additionally relies on this SAS System configuration file to establish reference to the location of the application and to provide system environment options conducive to the needs of the Enterprise Reporter's application environment.

During the installation process of Enterprise Reporter, a CONFIG.SAS file is created for you. It is a copy of your SAS System configuration with necessary modifications to run Enterprise Reporter. By default, this file is located in the !SASROOT\Business directory.

When you integrate the SAS System with the Enterprise Reporter, whether through a customized application, a SAS command or the SAS Desktop, you will need to do <u>one</u> of two things:

- Modify your existing CONFIG.SAS file
- Modify the Enterprise Reporter CONFIG.SAS file.

If you choose to modify the Enterprise Reporter's CONFIG.SAS, the only modification you need to make is to remove the –ICON option. The –ICON option indicates whether or not to iconize the SAS System DMS. Then, instead of launching ERSAS.EXE, launch SAS.EXE.

If, instead, you choose to modify your existing CONFIG.SAS file, rather than the Enterprise Reporter CONFIG.SAS file, you will need to make the following changes:

- Define !BUSROOT to point to the Enterprise Reporter location. In the following example, Enterprise Reporter was installed in C:\SAS\Business.

    -SET BUSROOT "C:\SAS\Business"

- Add !busroot to the concatenation for SASHELP, SASMSG and PATH:

    -SASHELP(!BUSROOT\SASHELP
            !SASROOT\CORE\SASHELP
            !SASROOT\SASCFG
            ......

    -SASMSG (!BUSROOT\SASMSG
            !SASEXT0\AF\SASMSG
            ......

    -PATH   (!BUSROOT\SASEXE
            !SASROOT\AF\SASEXE
            ....

You can choose to make further modifications to customize your reporting environment. These modifications include specifying menu and font control, SAS/CONNECT® software script locations and help file inclusion. *See Appendix A for details.*

Once you have modified the CONFIG.SAS file appropriately, you can successfully invoke Enterprise Reporter from your SAS System environment. You can pass a number of parameters to control Enterprise Reporter actions. For the AF (or AFA) command, these parameters are specified as options:

AF C=sashelp.business.start.scl *options*

The following options apply:

open = "<path\>*report*.**erf"**!"*report*.**erp"**!"*report*.**ert"**
print = "<path\>*report*.**erf"**!"*report*.**erp"**
html = "<path\>*report*.**erf"**!"*report*.**erp"**
pdf = **"<**path\>*report*.**erf**"!"*report*.**erp**"

These options are used to respectively load a report in design view, print a report, load a report and create the output in HTML format or in Portable Document Format (PDF). *Note:* To create a PDF file you need the Adobe Acrobat software version that includes a PDF writer driver.

If you want to redirect your HTML or PDF output to another destination, for example to your web server, you can specify this with the **out** option:

out = "<path\>*nam*e.**htm"**!"*name*.**pdf**"

While loading a report, you can switch the data source. For example, if the report is associated with a 1998 data set and you want to reuse the report specification for a 1999 data set with the same columns as the 1998 data set, you can specify one of the following options in addition to the ones mentioned above:

data = **"***libref.dataset"*
file = **"<**path\>*sasfile*.**SD2"**!"*sasfile*.**SV2"**

To prevent the user Login window from displaying, you can specify the userid and password as options as well:

user = "*userid"*
pw = "*password"*

When Enterprise Reporter is invoked, the default view is Design View. You are given the opportunity to change the default behavior. For example, you can display a selected report in Data View or specify to open other windows in Enterprise Reporter by using this option:

command = **"***command <; command>"*

Commands include:

- AGENT   - opens the InfoCenter window
- LOADAGEN - opens the Select InfoFolder window
- FILTER   - opens the Filter window
- DESIGNM   - displays the report in Design View
- SCREEN   - displays the report in Data View
- SPREVIEW   - displays the report in Print Preview
- NEW   - clears the current report
- OPEN   - opens the Open Report dialog window
- GALLERY   - opens the Report Gallery window
- SAVE   - saves the current report
- SAVEAS   - opens the Save As dialog window
- SPRINT   - prints the current report

Finally, you can avoid the default behavior to icon the SAS session upon return from Enterprise Reporter by issuing

exiticon = "NO".

Example: Invoke Enterprise Reporter with a DM statement and display the report c:\myreport.erp in data view. Also, avoid the SAS session to be iconnized when returning from Enterprise Reporter:

DM 'AFA c=sashelp.business.start.scl
        Open="c:\myreport.erp"
        Command=SCREEN
        Exiticon="NO" AFA';

When used from SCL in connection with the CALL DISPLAY routine, all the options mentioned above are passed through the local environment list. *Refer to Appendix B for an example.*

You are provided with many integration options when you use Enterprise Reporter with your SAS/AF software applications. The techniques above also apply to integration with the SAS Display Manager (for example, adding a command to the toolbar that launches Enterprise Reporter), the SAS desktop or SAS macros.

You can also benefit from the SAS System integration components by using SAS/GRAPH software to extend report graphics.

Scenario 4: Incorporating SAS/GRAPH Software Output into your Enterprise Reporter Reports

Enterprise Reporter allows you to include both static and dynamic graphs in their reports. Dynamic graphs allow you the ability to include graphs pertinent to specified groupings. You can use the standard features of the Enterprise Reporter Graph report object to run against live data or you can include images. You can therefore include such images as GIFs, BMPs, and JPGs. to incorporate images you have already created in your application using SAS/GRAPH software into Enterprise Reporter.

You do this by writing your own SAS program to convert the SAS/GRAPH GRSEG catalog entries you have created into standard Microsoft Windows compliant images. *See Appendix D for a snippet of source code to do this.* Keep in mind, this conversion can be done either through a batch job or during the generation of the Enterprise Reporter report.

Once the images are created, you can include them in your report just as you would any other image. You do this by updating both the report and the InfoFolder to recognize the image(s).

SAS/GRAPH software provides you with the ability to extend Enterprise Reporter software's graphic capabilities. You can further extend the reporter's default behavior by incorporating SAS/EIS software's information delivery application objects.

Scenario 5: Integrating Customized Applications with SAS/EIS Software

A frequently requested function for users of SAS/EIS software applications is to create nicely formatted reports after drilling down into their data. An example is through the SAS/EIS multidimensional report object. You can provide this functionality for them by overriding SAS/EIS report object methods.

Many of the SAS/EIS software objects have a _WRITE_ method. This is used to create a SAS table containing the current subset of data. Thereafter, just invoke Enterprise Reporter specifying this table as the input data source.

The user, for example, can invoke Enterprise Reporter from the pop-up menu of the multidimensional report object at any given time of the exploration of the data cube. You can make this possi-

ble by overwriting the _POSTINIT_ method of the multidimensional report object. *Please see the SCL code in Appendix B for details on how to do this.*

Once reports are created, whether static or dynamic, whether through the use of customized application extensions or solely by use of Enterprise Reporter, there are a variety of ways to distribute the information. You can use Enterprise Reporter Report Gallery, a Windows Explorer, customized applications or through a web browser.

Scenario 6: Distributing Reports with Enterprise Reporter Web Gallery

One method for distributing company-wide reports is through Enterprise Reporter Web Gallery.
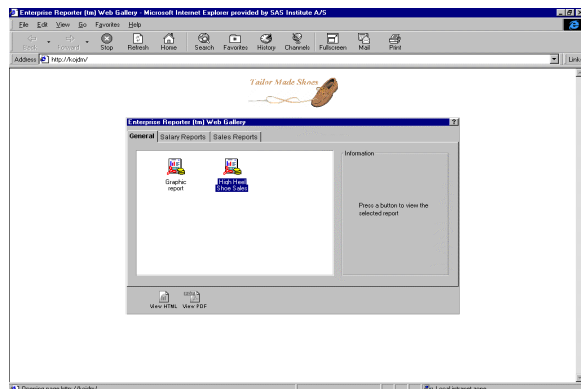

Figure 4: *Enterprise Reporter Web Gallery*

The user interface in the Web browser is similar to the Report Gallery in Enterprise Reporter. Enterprise Reporter Web Gallery users can:

- Browse reports as HTML (with or without HTML frames)
- Print reports as they would have been printed from Enterprise Reporter
- Download and print reports in PDF format (for example, through Adobe Acrobat)

And you, as an Enterprise Reporter *Administrator* can:

- Specify which reports the users will see (.ERP, .ERF, .HTML)
- Specify which folders or tabs users will see
- Specify the background layout (HTML) of the Web Gallery
- Specify which HTML templates to use for generating framed HTML
- Decide whether or not to generate reports in PDF
- Direct the web pages to the desired location (web server)
- Run the creation/update of the web reports interactively or in batch.

The display in **Figure 4** above contains the Enterprise Reporter Web Gallery interface. In this case, the two tabs *Sales Reports* and *Salary Reports* are available for users to select. Additionally, reports in the Web Gallery can be in the form of PDF or any HTML generated output. That is, you can use Enterprise Reporter along with other sources to generate HTML. See **Figure 5** for an explanation on how these options are specified.

To use Enterprise Reporter Web Gallery, you need a web server. Web Gallery is a Java applet that is downloaded to the user's browser. The users need a web browser that supports Java, graphics and preferably HTML frames. To use the PDF facility, Adobe Acrobat Writer is needed on the server. Adobe Acrobat Reader or other software that can print PDF files is needed on the client.

To establish this distribution method for users, you first create the desired reports in Enterprise Reporter and save them as .ERP or .ERF files in organized folders. The Enterprise Reporter Web Gallery is installed from the Enterprise Reporter CD-ROM. You invoke the Web Gallery Update Tool from the Windows Start prompt. Then, you fill in the window indicated in **Figure 5** and click on **Start** to generate the Web Gallery.
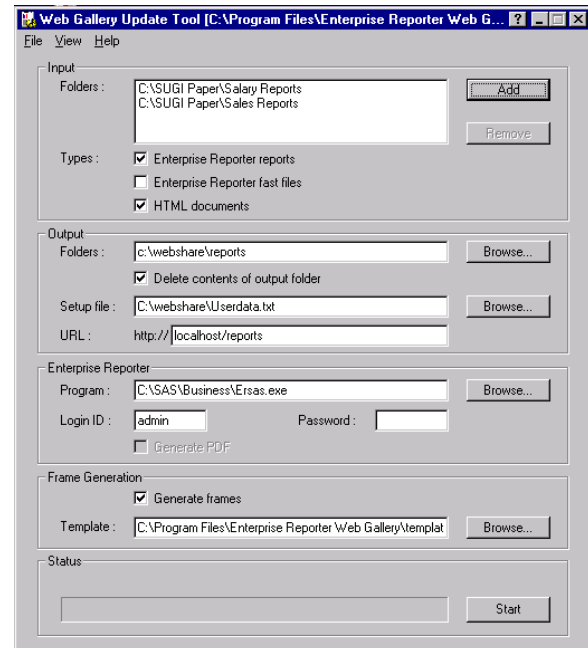

Figure 5 *Web Gallery Update Tool*

Each of the parameter specifications for the Enterprise Reporter Web Gallery set up tool is described below.

Segment	Label	Description
Input	Folders	The distribution folder where Enterprise Reports are stored
	Types	The type of report to include in Web Gallery (.ERP, .ERF or HTML)
Output	Folders	The location of the generated out files in the form of HTML. (It is recommended to initially specify an empty folder)
	Delete contents of folder	Controls whether the output folder contents should be deleted every time the Update Tool is used.
	Setup File	Initially generates the file to use as a basis for setting up the Web Gallery applet that users see. (Specify a non-existing file available to your web browser)
	URL	The logical web address that points to the physical address in the Folder field. The logical-physical relation is specified

		during the installation of the Web Gallery in the Gallery.htm file.
Enter-prise Reporter	Program	Location of the Enterprise Re-porter executable file (ERSAS.EXE)
	Login ID	Your Enterprise Reporter identifi-cation. You must at least have read access to the InfoFolders that the reports use.
	PDF	Controls whether the reports should additionally be created in PDF format. Note this requires Adobe Acrobat Writer on the server.
Frame Genera-tion	Generate frames	Controls whether to generate framed HTML.
	Template	Location of frame template to use if Generate frames is selected.
Status	Start	Launches Enterprise Reporter and generates the reports and its out-put.

Note in **Figure 5,** Enterprise Reporter reports and HTML are pulled from C:\SUGI Paper\Salary Reports and C:\SUGI Pa-per\Sales reports respectively. The output is generated and placed in C:\webshare\reports and the URL, http//localhost/reports. Typi-cally, the output will be placed on your local area network for all of your users to access from your corporate Intranet. C:\SAS\Business\ersas.exe is used to generate the reports and place them in these designated locations. Additionally, a template using HTML frames is used to display the report. Sample templates are provided with the software, but you can also specify templates unique to your organization. Changing the logo, for example, is quite simple to do.

With Release 1.5 of Enterprise Reporter software, the new Web Gallery allows you to distribute reports to many people through a web browser. Users, therefore, only need a thin client interface to access reports. This may lead you to ask, what other important reporting capabilities will be available as each new release makes its way to market?

Scenario 7: Future Directions of Enterprise Re-porter Software

As an integrated part of the SAS end-to-end data warehousing solution, future versions of Enterprise Reporter will provide in-creased flexibility and additional seamless integration features.

From within SAS/Warehouse Administrator software, you will be able to select a certain area of your data warehouse, and select to create an Enterprise Reporter InfoFolder based on this. The only information required is a name for the InfoFolder. The InfoFolder will then be created automatically. If you want to customize the InfoFolder, or distribute to Enterprise Reporter user groups, you can be brought into the relevant Enterprise Reporter window di-rectly to perform this.

Currently work is underway to allow you as a SAS/AF software or SAS/EIS software developer to invoke and perform further actions in Enterprise Reporter programmatically from your application. This will be possible through a scripting feature, where you can instantiate a scripter class that communicates the actions you want to perform in Enterprise Reporter from an SCL program, and also parses information on events in Enterprise Reporter back to your SCL program.

The Enterprise Reporter Web Gallery will also be enhanced to provide you and your users with a more dynamic environment. The first step in this direction will be to allow your users to identify themselves using their userid and password when they invoke the Web Gallery. This will in turn call the user administration compo-nent of Enterprise Reporter in order to have the user only see the reports for the user groups that he or she is a member.

Conclusion

Enterprise Reporter Software delivers an important component of SAS Institute's integrated Data Warehouse strategy. Use of indus-try-standard user interfaces increases productivity for business users while administrative tools reduce the cost of managing the reporting process. You have the opportunity to focus on a com-plete, integrated reporting solution so your users can concentrate on report design and bottom-line business decisions.

Appendix A: Optional settings for customizing your standard CONFIG.SAS file

The following points provide information on optional customiza-tions to your SAS System CONFIG.SAS file.

- Include Enterprise Reporter settings in your configuration file If you have coexisting applications you can choose whether or not to include these settings.

  -noawsmenumerge
  -awstitle 'Enterprise Reporter'
  -font 'Sasfont' 8
  -usericon C:\SAS\Business\sasexe\wutil32.dll 2

  For Enterprise Reporter settings, it is best to specify –noaws-menumerge. When you specify this option, the AWS does not contain the Window and Options menus. If your other applications require these menus, then omit this option from your configuration file and specify it instead in the OPTIONS statement prior to calling Enterprise Reporter.

  Note: Some of these options, noawsmenumerge, icon, and font can also be specified in an OPTIONS statement prior to calling Enterprise Reporter.

- Incorporate Enterprise Reporter Help files by Appending Help Files to SAS System files.

  SAS Help files are directed by the –**Helploc** option in your configuration file. *This Helploc option does not support con-catenation*. Therefore, simply copy the Enterprise Reporter Help files over to the SAS System Help folder.

  Change the –helploc option

  From:-Helploc !busroot\winhelp
  To: -Helploc !sasroot\core\winhelp

  Then, copy the files business.hlp and buisiness.cnt

  From:!busroot\winhelp
  To: !sasroot\core\winhelp

- Note the following statement in the configuration file for SAS/CONNECT software script files. In this example, Enterprise Reporter's server folder and definitions were installed in C:\SAS\Business\Servdef:

  -SASSCRIPT "C:\SAS\Business\Servdef"

  Enterprise Reporter takes advantage of standard SAS/CONNECT software scripts to access remote data. During installation of the Enterprise Reporter, the standard script files are copied to the location indicated above. If you are referencing SAS/CONNECT software scripts for other purposes (e.g. The SQL Query Window), you need to point to a single script file location. In an environment where you are calling Enterprise Reporter from the SAS System, you will want to combine your scripts in one location for the purposes of integration.

  For example, by default the SAS System uses:

  -SASSCRIPT !sasext0\connect\saslink

  If you want to continue using the SAS/CONNECT script files provided with the SAS System, modify the scripts in this directory as it relates to Enterprise Reporter. Alternatively, you can copy over the script files from !BUSROOT\Servdef. Note that if you have previously created server definitions from the Server Definition window in the Enterprise Reporter, you will need to modify any existing Servers you may have defined to Enterprise Reporter to point to the specified script file folder.

Appendix B: Customized SAS Software Application Integration Examples

SAS/AF Software SCL and Enterprise Reporter Integration

Example: Invoke Enterprise Reporter from SCL and pass parameters to open a report called c:\er\report.erp, but use the table WORK.SUBSET as the input data. Reset the SAS session when returning:

```
init:
  rc = rc;
  loc_list = envlist('L');
  cmd_list = getniteml(loc_list,'_CMDLIST_');
  loc_list = delnitem(loc_list,'_cmdlist_');
  new_cmd  = makelist();
  new_cmd  = setnitemc(new_cmd, 'work.subset',
          'DATA', -1);
  new_cmd  = setnitemc(new_cmd, 'NO','EXITICON', -1 );
  new_cmd  = setnitemc(new_cmd, 'c:\er\report.erp',
          'OPEN', -1 );
  loc_list = insertl(loc_list,new_cmd,1,'_CMDLIST_');
  call display('sashelp.business.start.scl');
  loc_list = delnitem(loc_list,'_cmdlist_');
  rc = dellist(new_cmd,'Y');
  loc_list = insertl(loc_list,cmd_list,1,'_CMDLIST_');
  call execcmdi('awsrestore; toolload');
  return;
```

SAS/EIS Software Multidimensional Report and Enterprise Reporter Integration

SCL code for overwriting the _POSTINIT_ method of the multi-dimensional report object in SAS/EIS software to call Enterprise Reporter and load the current subset of data:

```
length _method_ $40;
/* Pop-up menu. Overrides _POSTINIT_*/
postinit:  method;
  call super(_self_,_method_);

  /* Add an item to the popup menu */
  select_l = getniteml(appllist,'SELECT_L');
  popup_list = getniteml(select_l,'POPUP');
  popup_list = getniteml(popup_list,'_ANYWHERE_');
  actionsl=getniteml(popup_list,'POPUP_L');
  mapl = getniteml(popup_list,'MAP_L');
  rc=insertc(actionsl,'Export to Enterprise Reporter',-1);
  rc=insertc(mapl, 'MODEL',-1);
  rc=insertc(mapl, '_EXPORT_SER_',-1);
  call send(modelid,'_SET_INSTANCE_METHOD_',
          '_EXPORT_SER_', screenname(),'EX_SER');
  endmethod;

/*  Method to invoke Enterprise Reporter */
EX_SER:
  method;
  call send(_SELF_,'_write_','work.out','Y','Y');
  loc_list = envlist('L');
  cmd_list = getniteml(loc_list, '_CMDLIST_');
  loc_list = delnitem(loc_list,'_CMDLIST_');
  new_cmd  = makelist();
  new_cmd  = setnitemc(new_cmd, 'work.out', 'DATA', -1 );
  new_cmd  = setnitemc(new_cmd, 'NO','exiticon', -1 );
  loc_list = insertl(loc_list,new_cmd,1,'_CMDLIST_');
  rc = search('sashelp.bvgraph');
  rc = search('sashelp.bvpmenu');
  rc = search('sashelp.bvtooll');
  rc = search('sashelp.tools');
  rc = search('sashelp.useradm');
  rc = search('sashelp.hmddb');
  rc = search('sashelp.rmddb');
  rc = search('sashelp.mddb');
  rc = search('sashelp.printcon');
  rc = search('sashelp.business');
  call display('sashelp.business.start.scl');
  rc = search('-POPALL');
  loc_lis = delnitem(loc_list,'_CMDLIST_');
  rc = dellist(new_cmd,'Y');
  loc_list = insertl(loc_list,cmd_list,1,'_CMDLIST_');
  call execcmdi('awsrestore; toolload');
  endmethod;
```

```
/* Suppress warnings */
_self_   = _self_;
_method_ = _method_;
rc = rc;
modelid  = modelid;
appllist = appllist;
```

Appendix C: START.SAS Program to Pre-processing data to create a "Top-10" report

This example illustrates a SAS software program that extracts the 10 flight numbers with the highest number of boarded passengers on each Monday into a SAS table called SASUSER.TOP10.

```
libname airline '!busroot\demodata\airline';
proc sort data = airline.march (keep= date flight boarded)
          out  = sorted;
  by date descending boarded;
  where weekday(date) = 2;
  run;


data sasuser.top10;
  set sorted end = no_more;
  by date;
  if first.date then
    rank = 0;
  rank + 1;
  if rank <= 10 then
    output;
  if no_more then
    call symput('subsetdt', put(datetime(), datetime16.));
  run;
```

Appendix D: Example of SAS/GRAPH Software Integration

This example uses the sample data provided with the software representing *Tailor Made Shoes*. It generates a graph for each country, showing the sum of sold shoes per product. The result of this example is similar to creating an Enterprise Reporter report by placing a Master object and a Vertical Bar chart below it. The Master object includes the variable Country and the Vertical Bar Chart contains the variable Product Name on the X-axis and Sold Shoes on the Y-axis.

(Remember to copy the sample Tailor Made Shoes InfoFolder from the Sample Tab to the User Tab). Below is the source code to submit either as part of the InfoFolder (START.SAS) or through a batch job.

```
/* Set up the location of the data */
%let dataloc=
C:\Enterprise Reporter\Business\Demodata\Shoes;
libname shoes "&dataloc";
proc sql;
  create table _tmp
  as select a.*, b.prodname
  from shoes.amount
  as a left join shoes.name
```

```
  as b on a.prodid = b.prodid
  order by c_id;
  quit;


data shoes.graph(keep= c_id graphloc);
  set _tmp end=end;
  by c_id;
  length graphloc $ 200;
  label graphloc =
          "Sold shoes, grouped by country";
  if first.c_id then do;
    count + 1;
    call symput('CC_' !!
            left(put(count, 5.)), c_id);
    graphloc = "&dataloc\Graphs\" !!
            trim(c_id) !! ".gif";
    output;
    end;
  if end then
    call symput('count', put(count, 5.));
  run;


goptions reset =(axis, legend, pattern, symbol, title, footnote)
        norotate hpos =0  vpos=0
        htext= ftext= ctext= target=
        gaccess= gsfmode= gsflen= interpol= hsize=
        vsize=;


goptions dev=gif  ftext=swissl
        gsfname=grafout gsfmode=replace  gsflen=80
        interpol=join hsize=7 in  vsize=6 in;


pattern  color=blue value=solid;
axis      color=black width=1.0
        label=(font=swissb height=0.8)
        value=(font=swissl height=0.7);

/*  The macro variable CC&I=the name of the GRSEG cre-
ated by the SAS/GRAPH output source code  */
%macro dographs;
  %do i=1 %to &count;
    %let cname=%trim(&&CC_&I..);
    %let fname=&dataloc\Graphs\&cname..gif;
    filename grafout "&fname";
    proc gchart data=_tmp;
      vbar prodname / maxis=axis
              raxis=axis sumvar=sold;
      where c_id="&cname";
      run;
    quit;
  %end;
%mend;
```

```
proc sql;
   drop table _tmp;
   quit;
libname shoes clear;
```

References:

Enterprise Reporter™ Software: User and Data Administration,
Course Notes, 1998

Enterprise Reporter Software: http://www.sas.com

Adobe Systems Inc. http://www.adobe.com

Contact Information

Your comments and questions are valued and encouraged. Contact
authors at:

Juli Yaguda
SAS Institute Inc., SAS Business Solutions
SAS Campus Drive
Cary, North Carolina 27513
Telephone: 919.677.8000
Fax: 919.677.4444
Email: snojes@wnt.sas.com

Jens Dahl Mikkelsen
SAS Institute A/S
Koebmagergade 9A
DK-1150 Copenhagen K, Denmark
Telephone: +45 33 96 27 02
Email: sdkjdm@sdk.sas.com