

Give'm Their Way – They'll Love it!

Sarita Prasad Bedge,
Family Care Inc., Portland, Oregon

ABSTRACT

As many of us know, many users prefer their SAS data in MS[®] Excel spreadsheet. And many of us also may know there are numerous ways to deliver it from SAS to Excel spreadsheet. The various methods depend on your OS and whether you have SAS/Access or some other SAS software licenses and end user's technical ability. Here are few methods...and the list is not comprehensive at all.

- Export/Import Wizards
- Proc export/Import
- Data step
- Proc Print
- Proc Printto
- SAS libname to EXCEL
- ODS CSV /ODS CSVALL /ODS CSBYLINE
- ODS HTML
- ODS Markup
- DDE
- ODBC
- OLE DB
- SAS Intr/Net
- %ds2csv macro
- **SAS Add-in for MS Office (AMO)**

This presentation focuses on using the SAS[®] AMO, one of the client-tool in SAS[®] Enterprise BI bundle, and SAS[®] stored processes. Based on Family Care Inc.'s user responses, they like this method because the SAS reports are easy to locate, easy to execute and easy to work with the data once the SAS data is in their familiar territory –in MS[®] Excel! Lastly and equally important, the user does not need to have any SAS skill at all.

As a developer and SAS administrator, this method is easy to administer because the code for the stored process is maintained on a SAS server and its security is managed by SAS[®] Metadata Server. And because the reports are parameterized, it allows many users to customize the same report to their individual needs, without any additional administration effort.

To understand how SAS to Excel technology works and how the decision to start using AMO at Family Care, I need to explain how few of the SAS® EBI pieces relate to AMO.

KEY SAS® EBI MODULES WHEN USING AMO

SAS® Enterprise BI Server is a bundle of about twenty SAS modules to architect, develop and maintain BI system in a multi-user, multi-client environment. Few of the key modules for this presentation are SAS® Metadata Server, SAS® Metadata Repository, SAS® Stored Process, SAS® Stored Process Server, SAS® Information Delivery Portal, SAS® management console and SAS AMO.

SAS® Metadata Server

A multi-user, centralized server that makes metadata available from SAS® Metadata Repository to all SAS users and SAS enabled applications, including SAS® Add-in for Excel. In another words, it's a SAS process that enables clients such as SAS Add-in for Excel to read metadata from and write metadata to SAS® Metadata repository. A repository is a physical location in which collection of related metadata objects are stored, such as the SAS® Stored Process and its parameter, if the SAS® Stored Process was parameterized.

SAS® Stored Process

A SAS® Stored Process is a SAS program written by a SAS Programmer that are registered in the SAS® Metadata Server and are available to be run from SAS Add-in for Excel, plus few other client tools in SAS® EBI. SAS® Stored Process can also have parameters to customize the SAS® Stored Process for each individual user.

Because stored process is stored on the server and can be executed as required by the requesting applications, it provides an effective method for change control management. For example, instead of embedding the SAS code into the client application, you can centrally maintain and manage this code from the server. This allows the SAS developer/administrator the ability to change the SAS program as needed and at the same time ensures every client that invokes the stored process to execute the latest version of the code.

Stored process code for a report that is run in AMO at Family Care. This paper will not go into the details of creating, explaining or registering a SAS® Stored Process because excellent SAS User Group Presentations and SAS documentation exist on it.
--

```

* Begin EG generated code (do not edit this line);
* Stored process registered by
* Enterprise Guide Stored Process Manager v4.1
*
* =====
* Stored process name: stp_for_MemberByProvider
* History: 25Feb08: Sarita Bedge
* =====
*

*ProcessBody;
%global ProviderID ProductLine;
%STPBEGIN;

* End EG generated code (do not edit this line);

libname FAMTEMP (work); * production libname
*libname FAMTEMP 'E:\SASData\FAMTEMP'; *development libname

/**** This section for testing/developing code only****;
%let _metauser="familycare\saritab";
%let ProviderID = P000-0000;
%let ProductLine = A;
*/

options symbolgen; * To debug macro resolution;

filename pwfile 'e:\SASData\FAMTEMP\sasbippwfile';

/*
proc pwencode in=" out=pwfile;
run;
*/

**** Pass database password into macro variable ****;
data _null_;
  infile pwfile obs=1 length=l;
  input @;
  input @1 line $varying1024. l;
  call symput('dbpass',substr(line,1,l));
run;

data prompt;
*metauser=&_metauser; * development;
metauser=symget('_metauser'); * production
ProviderID = resolve('&ProviderID');
ProductLine = resolve('&ProductLine');
run;

```

```

PROC SQL;
delete * from fastbip.sasprompt
      where userkey=symget('_metauser')
      and reportname = 'MemberByProvider';

insert into fastbip.sasprompt
      select metauser as UserKey,
      'MemberByProvider' as ReportName,
      '1' as PageID,
      'ProviderID' as ProviderID_PromptName,
      ProviderID as ProviderID_PromptValue
from prompt;

insert into fastbip.sasprompt
      select metauser as UserKey,
      'MemberByProvider' as ReportName,
      '1' as PageID,
      'ProductLine' as ProductLine_PromptName,
      Productline as ProductLine_PromptValue
from prompt;

connect to odbc as newbip (DSN=FASTBIP UID=sas PWD="&dbpass");

drop table famtemp.MemberByProvider;

create table famtemp.MemberByProvider as
      select MemberID, MemberLastName, MemberFirstName, Language, Age, Gender,
      Street, Street2, City, State, Zip, Phone, ProductLine, PCPFromDate, ProviderLastName,
      ProviderFirstName, ProviderID
from connection to newbip (

EXEC sas.MemberByProvider
      @ProviderID = %unquote(%nrstr(%')&ProviderID%nrstr(%')),
      @ProductLine = &ProductLine,
      @debugFlag = 1
);

select * from famtemp.MemberByProvider;
disconnect from newbip;
quit;

* Begin EG generated code (do not edit this line);
*','*';run;
%STPEND;
* End EG generated code (do not edit this line);

```

SAS[®] Information Delivery Portal is part of SAS[®] EBI web-based, thin-client tool, enabling access to SAS reports created with SAS[®] Web Report Studio and some even embedded with SAS[®] Stored Processes. All the IDP reports have the ability to export its report data into MS[®] Excel. This was our first and only SAS client tool for deploying SAS reports for long time, until realizing many of the users were exporting the data to MS[®] Excel. Once realizing this need, deploying SAS[®] Add-in for Excel was the next client tool for data delivery to the users. The report delivery using AMO has continued to grow and since then and both client tools are being supported. If a user needs the data in MS[®] Excel to do further manipulation, then the report is created in SAS[®] add-in to Excel, otherwise there is choice between SAS Portal and AMO as the client tool.

To maintain consistency between the two deployed client tools, similar (ok, almost similar) folder structure was created to house the various reports based on the various department/job functions that exist at Family Care Inc. Furthermore, same structure is also maintained in SAS[®] BIP Tree and any other structure that is required for SAS[®] EBI.

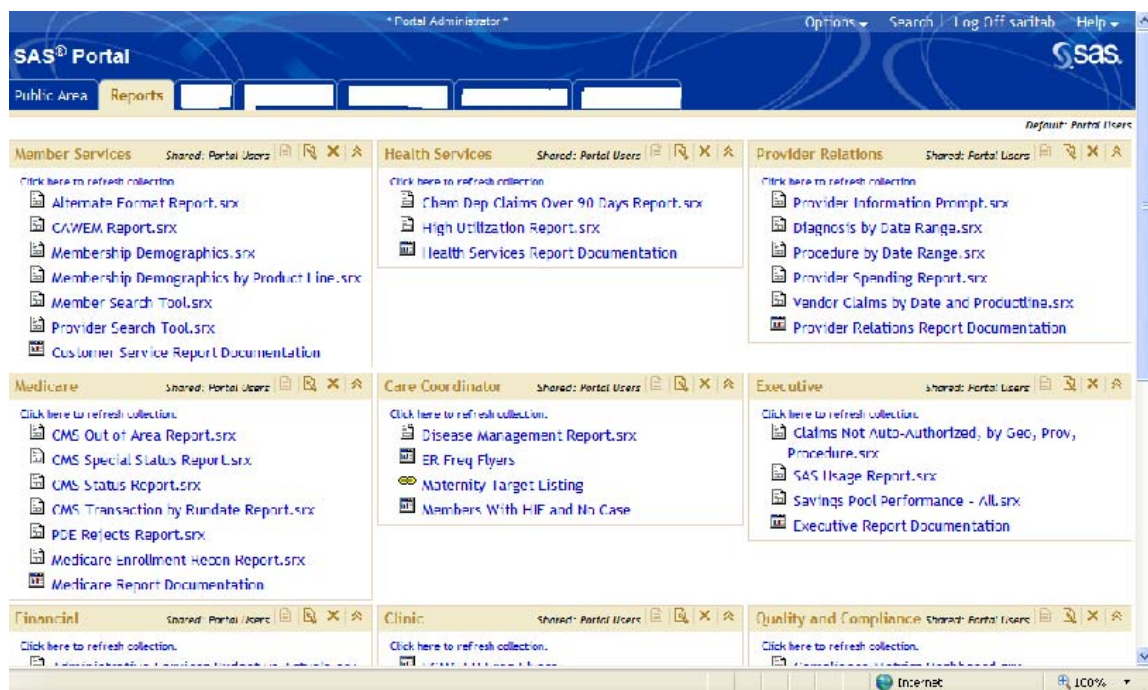


Figure1: Family Care’s SAS[®] Portal report showing various reporting folders. Similar folder structure is maintained in AMO client and other areas.

SAS[®] Management console

It is the graphical interface tool for managing SAS[®] EBI resources, such as SAS servers, library definitions and resource security for SAS[®] Enterprise BI objects. The

SAS® Management Console works by creating and maintaining metadata definitions for each computing resource or control. These metadata definitions are stored in a metadata repository on a SAS® Metadata Serve where they are available for use by other client applications, including SAS® Add-in for Excel.

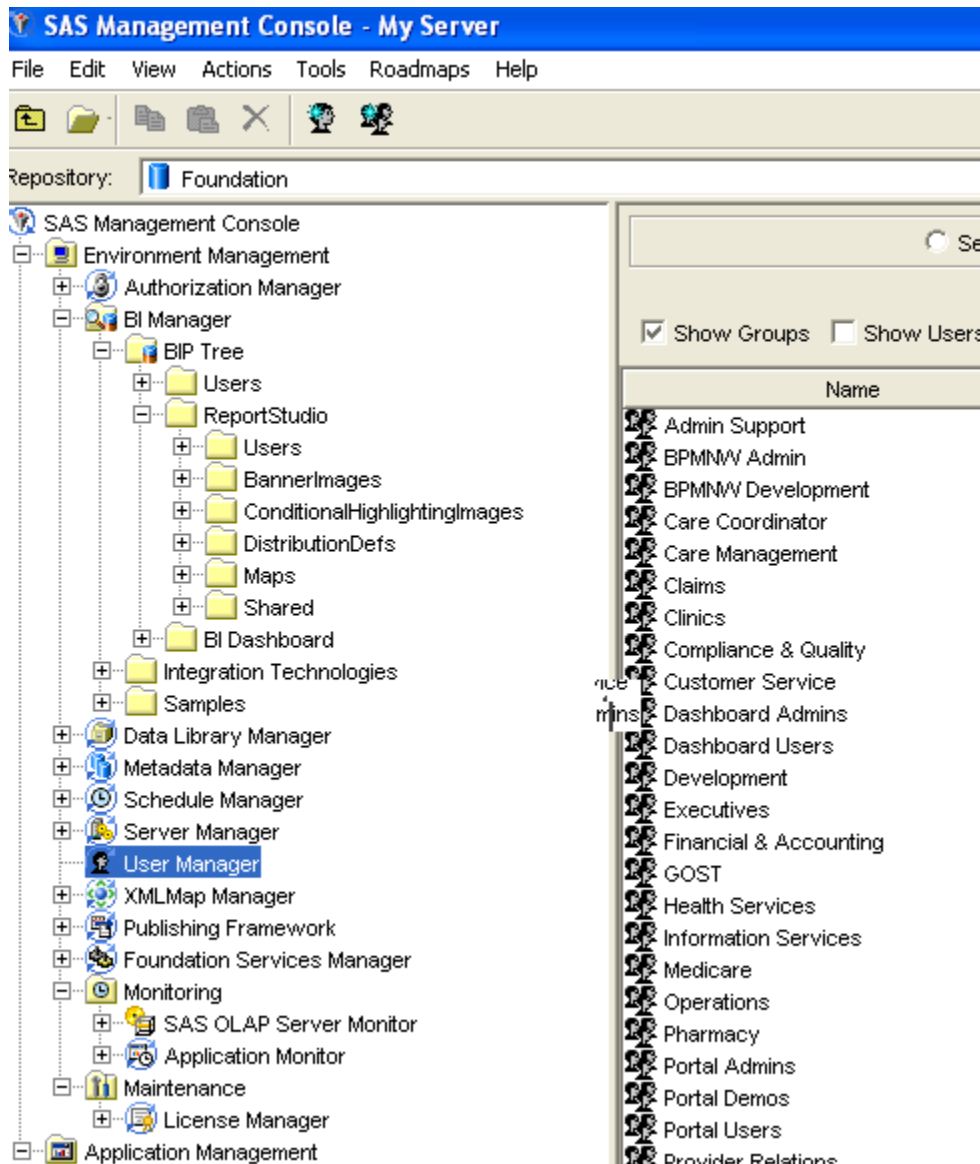


Figure 2: SAS® Management Console showing the contents that can be managed using it.

With this very brief overview of various pieces of the SAS® EBI and how they relate to AMO as client tool to deliver SAS data into MS® Excel, It is time to discuss the details of AMO and how it is implement at Family Care Inc.

SAS[®] ADD-IN FOR MS[®] OFFICE

SAS[®] Add-in for MS Excel is a Component Object Model (COM) add-in to Microsoft[®] Excel. According to Microsoft[®], COM add-in is a program that extends the capabilities of MS[®] Office applications such as MS[®] Excel. The SAS[®] Add-in for Microsoft[®] Office extends the functionality of Microsoft[®] Excel by enabling you to access SAS analytics and other SAS tasks such as reporting function from within MS[®] Excel.

The SAS[®] Add-in for Excel is a client tool within SAS[®] EBI, geared towards users who are familiar with Microsoft[®] Excel but not SAS software. It is available only as part of SAS[®] EBI or SAS[®] BI Server. It is not available as a stand alone SAS product.

Once SAS[®] Add-in for MS Office is installed, it automatically integrates SAS menu and SAS tool bars as part of the MS[®] Office application. Using these menus and toolbars you can harness much of the power of SAS without knowing SAS, including running SAS[®] stored process.

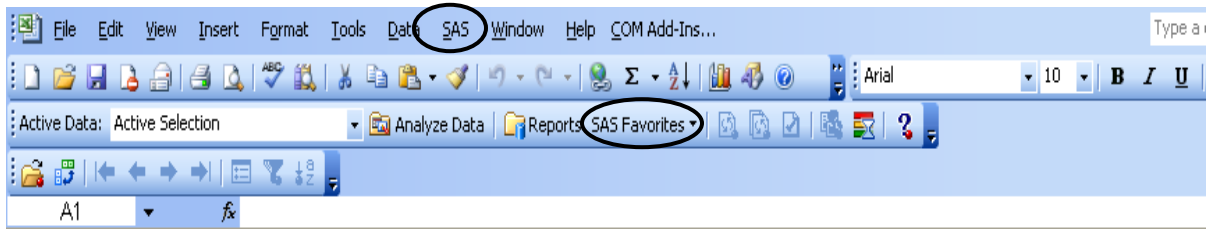


Figure 3: MS[®] Excel menu and toolbar after installing AMO. Notice the addition of SAS specific menus and SAS[®] Favorites.

After AMO was installed, the next step was to figure out the SAS[®] Favorites folders. For us the structure was obvious and as result it is structured same as the SAS Portal reports. I highly recommend you create and use SAS Favorites otherwise each a user needs to know the report path each time to run their report. Here is what a user would have to do if SAS[®] Favorites was not setup.

- Click on Reports button in Excel
 - > SAS Folders
 - > BIP Tree
 - > ReportStudio
 - > Shared
 - > Reports
 - > Members folder
 - > MemberByProvider report.

As you can see the path can be cumbersome and would increase the likelihood of user not being as receptive to using the tool or the report. Here is the same report being accessed using SAS® Favorites!

- Click on SAS Favorites in Excel
 - > Member folder
 - > MemberByProvider report

The figure 4 below shows the SAS® Favorites setup at Family Care. The SAS® Favorites is easy to use and hardly anything to remember when you need to run a report. The “Test” folder, as the name indicates, is the folder to house reports that are in development and test mode. Once it is ready for production, it is moved to a different folder. Furthermore, if same report is used by two or more departments, a favorite link is created in each of the folders.

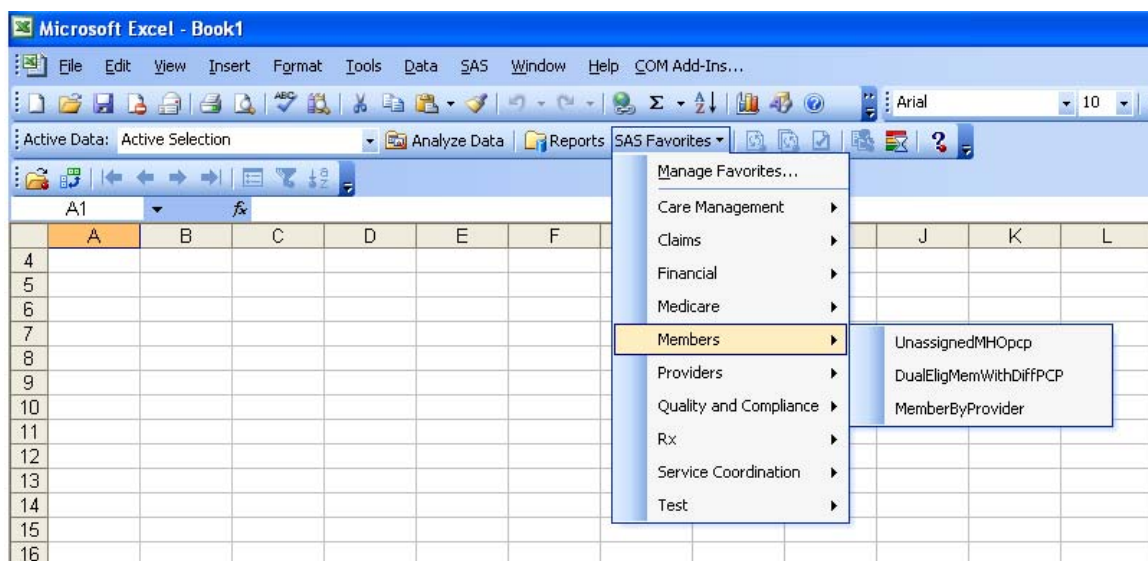


Figure 4: SAS® Favorites folder structure at Family Care Inc.

CONCLUSION

Often there are many ways to accomplish a task in SAS, including many ways to deliver SAS data into MS® Excel. At Family Care, we are using SAS Add-in for Excel because it is easy for the users and the developer to create and maintain. In order to use this method, you need either SAS® EBI or SAS® BI server license. Furthermore, to create and maintain the SAS Stored Process, you also need to understand and be able to use few of the key modules that come with the SAS® EBI/SAS® BI bundle.

SAS® ADD-IN FOR MS OFFICE TECHNICAL REQUIREMENTS

To install and implement SAS® Add-in for MS® Office here are the client, Server and software requirements. As always check with you SAS Account representative for the latest software and hardware requirements.

Client Environment:

- Windows® (x86-32); Windows® 2000 Professional or higher
- Internet Explorer® 5.5 and 6

Server Environment:

- SAS® Add-In for Microsoft® Office must connect to an environment running SAS® BI Server or SAS® Enterprise BI Server

Required Software:

- SAS® Add-In for Microsoft is delivered as part of SAS® BI Server and SAS® Enterprise BI Server.
- Microsoft® Office 2000/XP/2003
- Microsoft® .NET framework 1.1. SAS® includes MDAC 2.7 and the required Visual C++ components.

REFERENCES

SAS® Add-In 2.1 for Microsoft Office: Getting Started with Data Analysis.

SAS® Add-In for Microsoft Office – Leveraging SAS® Throughout the

Organization from Microsoft® Office, Jennifer Clegg, SAS Institute Inc., Cary, NC, Stephen McDaniel, SAS Institute Inc., Cary, NC, Paper 223-29

SAS® Enterprise Guide® and Stored Processes, Frederick E. Pratter, Eastern Oregon University, La Grande, Oregon, Paper 218-2007

SAS® 9.1.3 Intelligence Platform, Desktop Application Administration Guide

http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/index.html

ACKNOWLEDGEMENTS AND CONTACT INFORMATION

The author thanks Family Care Inc., Pacific Northwest SAS User Group and Jeff Livermore for making this presentation possible and Jason Snyder for reviewing the presentation.

Sarita Prasad Bedge
Family Care Inc.
2121 SW Broadway, Suite 300
Portland, OR 97201
503-345-5789
saritabedge@gmail.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies