

Paper 1199-2017

Using SASv9.cfg, autoexec.sas, SAS Registry, and Options to Set Up Base SAS®

Peter Eberhardt, Fernwood Consulting Group Inc.

Mengting Wang, Qualcomm

ABSTRACT

Are you frustrated with manually setting options to control your SAS® Display Manager sessions but become daunted every time you look at all the places you can set options and window layouts? In this paper, we look at various files SAS accesses when starting, what can (and cannot) go into them, and what takes precedence after all are executed. We also look at the SAS Registry and how to programmatically change settings. By the end of the paper, you will be comfortable in knowing where to make the changes that best fit your needs.

INTRODUCTION

Are you frustrated with manually setting options to control your SAS® Display Manager sessions but become daunted every time you look at all the places you can set options and window layouts? Would you like to have custom environments for different projects? Would you like to run your SAS programs right from a click on a desktop icon? All of this is possible, if you know all the pieces that SAS accesses when starting and running; however, many of us do not fully understand the way SAS sets up its environment. In this paper we will look at the three main places SAS accesses configuration settings:

-) sasv9.cfg
-) autoexec.sas
-) SAS registry

In addition, we will examine command line arguments. Although this paper will focus on desk top SAS 9.4 for Windows, the concepts are applicable to other environments. The paper assumes a basic understanding of SAS processing and well as Windows experience.

STEP 1 – THE COMMAND LINE

Since we start SAS by running an operating system command, let's start with the command line (Figure 1). In a default Windows installation, the Start menu will have a SAS folder in which there is a SAS icon to launch SAS. If you check the properties of this icon (right click on the icon) you will see something like Figure 2 (for Windows 7). In Windows 10 (Figure 3) from the start menu, right click... More...Open File location; In the Windows Explorer, right click on SAS 9.4 and select properties. The properties sheet for both versions is essentially the same (Figure 2, right side)

```
"C:\Program Files\SASHome\SASFoundation\9.4\sas.exe"  
-CONFIG "C:\Program Files\SASHome\SASFoundation\9.4\nls\en\sasv9.cfg"
```

Figure 1. The default command line.

From the command line (Figure 1) we can see when we start the SAS system, we are also specifying the configuration file to use with the –CONFIG switch (also called option); the –CONFIG switch specifies the name of the configuration file to use.

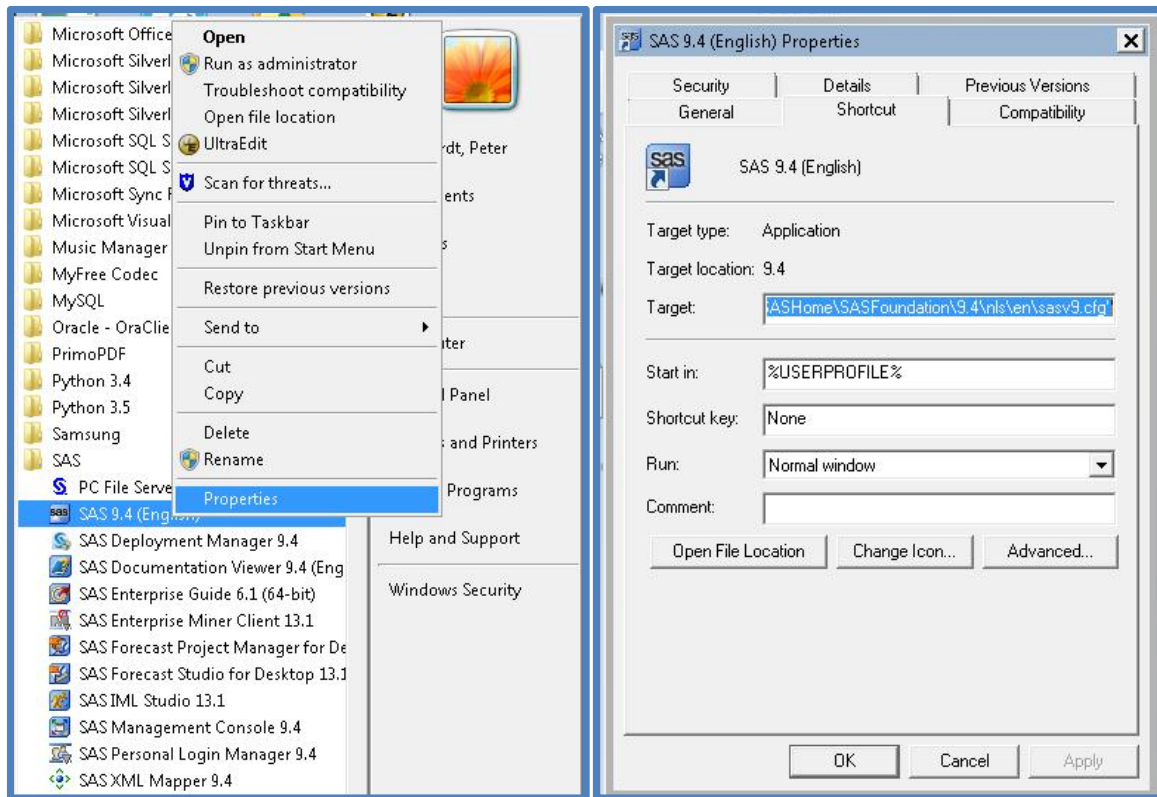


Figure 2. Windows 7 Start Menu (left) and Property Sheet (right)

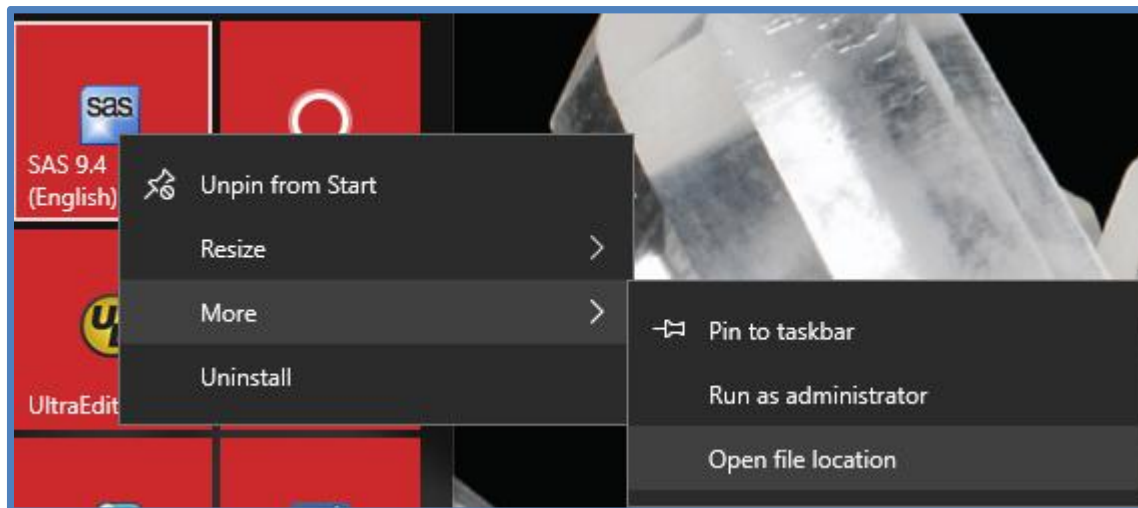


Figure 3 Windows 10 Start Menu.

In general, you should avoid making changes to the default configuration file. We will look at the SASv9.cfg file later, but for now, let's look at using the command line to make changes.

SASINITIALFOLDER

One option that many people want to change is SASINITIALFOLDER; this option tells SAS where to look when you do a File... Open Program (or File... Save). This option must be set at SAS invocation, that is, when SAS starts. To set it in the command line (Figure 4); the command line must be all on the same

```
"C:\Program Files\SASHome\SASFoundation\9.4\sas.exe"  
-CONFIG "C:\Program Files\SASHome\SASFoundation\9.4\nls\en\sasv9.cfg"  
-SASINITIALFOLDER "E:\projectX\SASpgms"
```

Figure 4. Setting an option on the command line.

line, it is split here for clarity.

To set the option on the command line, set the option switch, in this case –SASINITIALFOLDER, followed by the value, in this case the path to the programs - E:\projectX\SASpgms. The important thing to note here is that all the file paths must be enclosed in double quotes.

SAS WORK

In addition to SASINITIALFOLDER, setting a new location of the SAS WORK folder is a common.

```
"C:\Program Files\SASHome\SASFoundation\9.4\sas.exe"  
-CONFIG "C:\Program Files\SASHome\SASFoundation\9.4\nls\en\sasv9.cfg"  
-SASINITIALFOLDER "E:\projectX\SASpgms"  
-WORK "D:\SSD\SASWORK"
```

Figure 5. Setting SAS WORK.

As should be clear, setting different options using command line switches is easy, however, it would soon lead to an unwieldy command line. Let’s look at the configuration file in more detail.

STEP 2 – THE CONFIGURATION FILE

The SAS configuration file, SASv9.cfg, has many entries the user does not have to worry about, indeed should not even change; however, there are many SAS options which only be set during SAS invocation, meaning on the command line or on the configuration file. As we saw above, adding options to the command line can result in an unwieldy command, so the better way to set these options is in the configuration file. If you look in the SASv9.cfg file you will see the following warning line (Figure 6):

```
/* DO NOT EDIT BELOW THIS LINE - INSTALL Application edits below this line */
```

Figure 6. SASv9.cfg warning line.

yet many of the options you want to change are already defined below this line, for example TEMP. The simplest solution is to create a second configuration file, and use this one in the command line.

“WAIT” you say, “how does this help?”. This second configuration file first sets the options from the default configuration file, then sets the options you want. See Figure 7 for an example:

```
-CONFIG "C:\Program Files\SASHome\SASFoundation\9.4\nls\en\sasv9.cfg" ①  
-SASINITIALFOLDER "E:\projectX\SASpgms" ②  
-WORK "D:\SSD\SASWORK" ③
```

Figure 7. A custom configuration file.

Looking at Figure 7 we see the following:

–CONFIG tells SAS to process the contents of the configuration file C:\Program Files\SASHome\SASFoundation\9.4\nls\en\sasv9.cfg.

–SASINITIALFOLDER is set. If it was set in step 1, this would reset the value.

–WORK is set. If it was set in step 1, this would reset the value.

In fact, this looks like the command line we built (Figure 5), except all the entries in the file are on a separate line. By creating this new configuration file, we do not change the default file yet we can easily override any settings in that file plus add our own settings.

STEP 3 – THE AUTOEXEC FILE

The autoexec file is executed after SAS has started. Like any other .SAS file, it must contain valid SAS commands or errors will occur. Among the most common uses of the autoexec file is to set up libraries and views. Note that any valid filename can be used, it does not have to be called autoexec.sas.

Since SAS is now ‘up and running’ we cannot change any options that can only be set at invocation, however all other options can be set. In addition, since we now have access to all the programming facilities of SAS, including SAS macro, we can do custom, user specific customization as the following code shows:

```
%macro setLibraries;
%local whoAmI
    prefix
    ;
%let whoAMI = %upcase(&sysuserid);           ❶
%if &whoAMI = PETERPAN %then                ❷
%do;
    libname HR "C:\HoW\Yao\Data" access= readonly;
%end;
%else %if &whoAMI = WENDY                    ❷
%then
%do;
    libname HR "C:\HoW\Yao\Data";
%end;
%else %if &whoAMI = TINKERBELL               ❷
%then
%do;
    libname HR CLEAR;
%end;
%else ENDSAS                                ❸

%mend;
%setlibraries;
```

In this example we are checking the user ID of the person starting SAS (assuming the person starting SAS is using his/her own account). The program:

- 1) Capitalizes the user ID for comparisons
- 2) Checks each of the three users who are validated to use the system, PeterPan, Wendy, and TinklerBell. Each gets the appropriate LIBNAME statement.
- 3) If the user is not one of the validated users, SAS will close.

In this simple example we are only assigning a LIBNAME, however we could do much more. Later we will see an example with more customizations.

INITSTMT

Closely related to the autoexec file is the system option INITSTMT; this option has to be set at invocation. INIYSTMT specifies any valid statement you want executed **after** the autoexec completes. Since %INCLUDE is a valid statement for the INITSTMT option, it can also be used for extensive customization.

The autoexec file can be specified on the command line as in Figure 1, or as an –AUTOEXEC option in a configuration file. In addition, if an autoexec file is not specified (on the command line or in a configuration file), SAS will search for and AUTOEXEC.SAS file in the following locations:

1. The current folder.
2. Checks folders in the Windows PATH environment variable.
3. Checks the root folder of the current drive.
4. Checks the folder that contains the SAS.EXE file.

In all of the above, the search is only for AUTOEXEC.SAS and no –AUTOEXEC option was specified. In addition, if –NOAUTOEXEC is specified, the search for an AUTOEXEC.SAS is not performed.

STEP 4 - THE SAS REGISTRY

The SAS registry is probably the least understood part of SAS environment. The registry is the main storage location for the session configuration. To view the SAS registry contents, from the SAS menu select Solutions... Accessories... Registry Editor (

Figure 8)

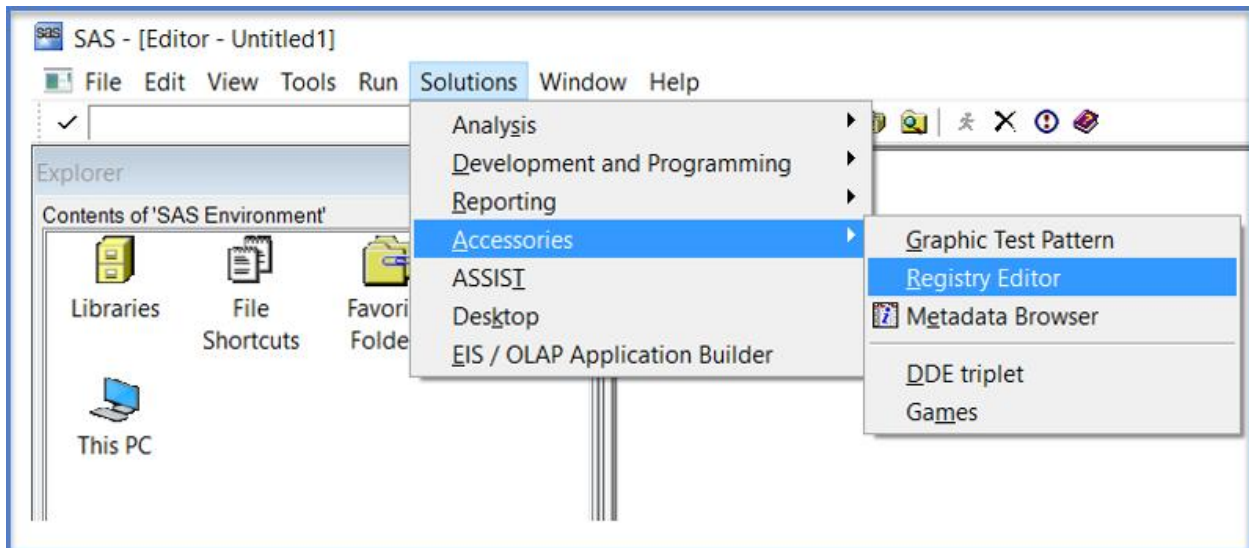


Figure 8. Accessing the SAS registry

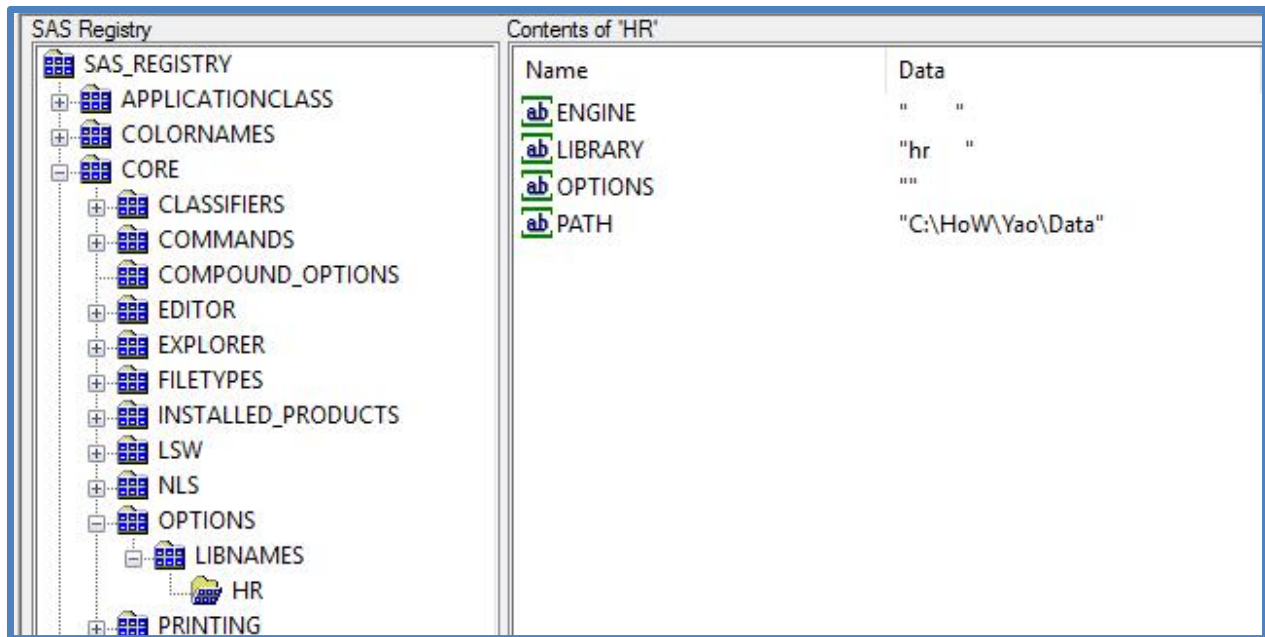


Figure 9. Viewing a library assigned at startup.

All of the entries in the Registry can be set through the Display Manager interface or through option settings so there should be no need to edit the Registry. In fact, the SAS documentation implies you may not want to edit the Registry Figure 10:

CAUTION:
If you make a mistake when you edit the registry, your system might become unstable or unusable.

CAUTION:
If you use the Registry Editor to change values, you are not warned if any entry is incorrect.

Figure 10. Warning messages from SAS documentation.

One type of item stored in the Registry you may want to check/change is the list of library assignments that are assigned at start-up; these are the libraries that are assigned through the New Library dialogue when the 'Enable at startup' box is checked. You can access through the "New Library" option under "Tools" (Figure 11):

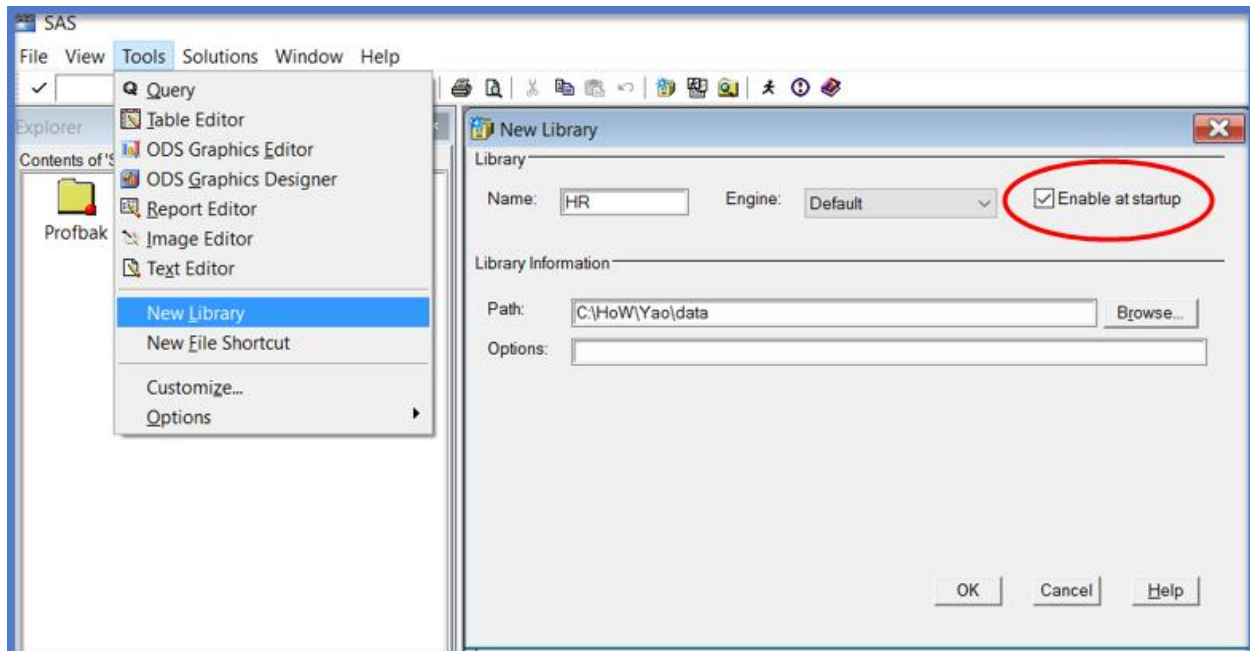


Figure 11. Assigning a library

Once these are set, the only place to check/change the values is in the Registry. As we saw in Figure 9, we can find these librefs in the **Core... Options...Libraries** tree in the Registry. To change entry, the path for example, in the right panel double click on the name (e.g. PATH). This will bring up a dialogue box that will allow you to change the value (Figure). Remember, when you save the entry, no validations will be performed; as a result you could end up with an unstable SAS session.

BACKING UP THE REGISTRY

If you plan to manually make changes to the Registry, you should first back it up. There are two fundamental ways to backup:

1. Use Windows to copy/backup the file

You can save a copy of the Sasuser registry file named registry.sas7bitm. Note that the result of an exact copy of the registry is at the moment that the copy was made. Any changes to the registry after the time that the copy is occurred will be lost.¹

To backup SAS registry using Explorer, the first step is to go to **View** and select **Explorer** (Figure 12), make sure that the explorer window is left open.

¹ Chapter 14 The SAS Registry, SAS 9.4 Language Reference: Concepts, Third Edition

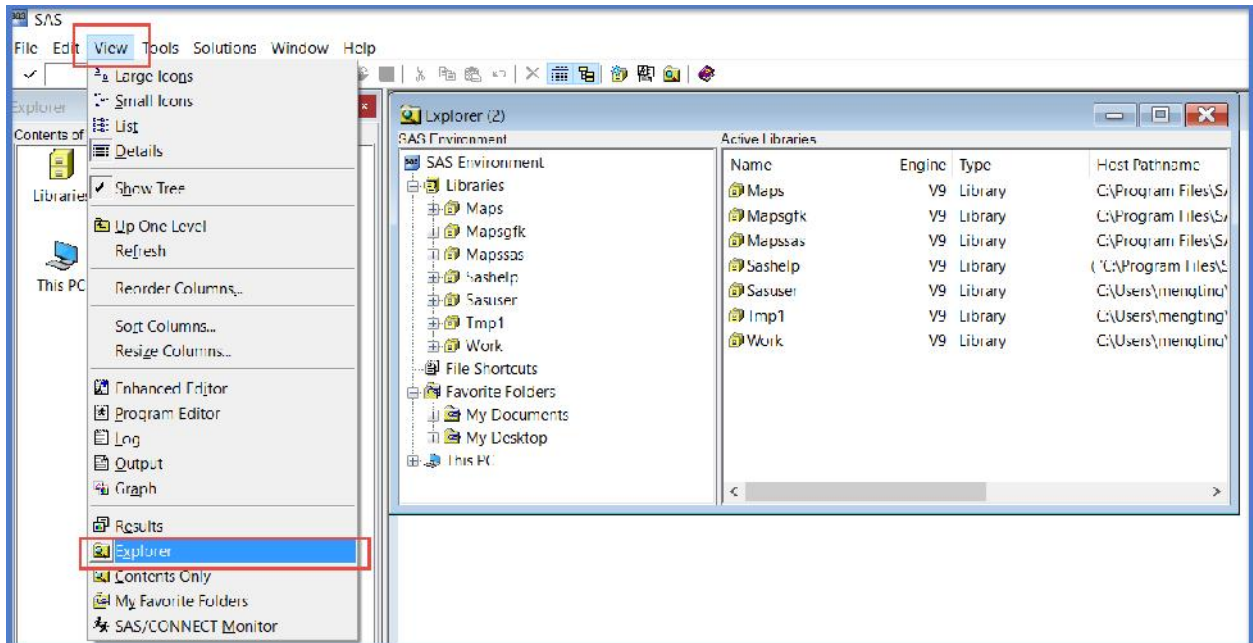


Figure 12. Opening Explorer window from View.

Second step is to go to **Tools**, and select **Options > Explorer** (Figure 13). (Note that without the first step, the explorer option under tools in this step will not pop up.)

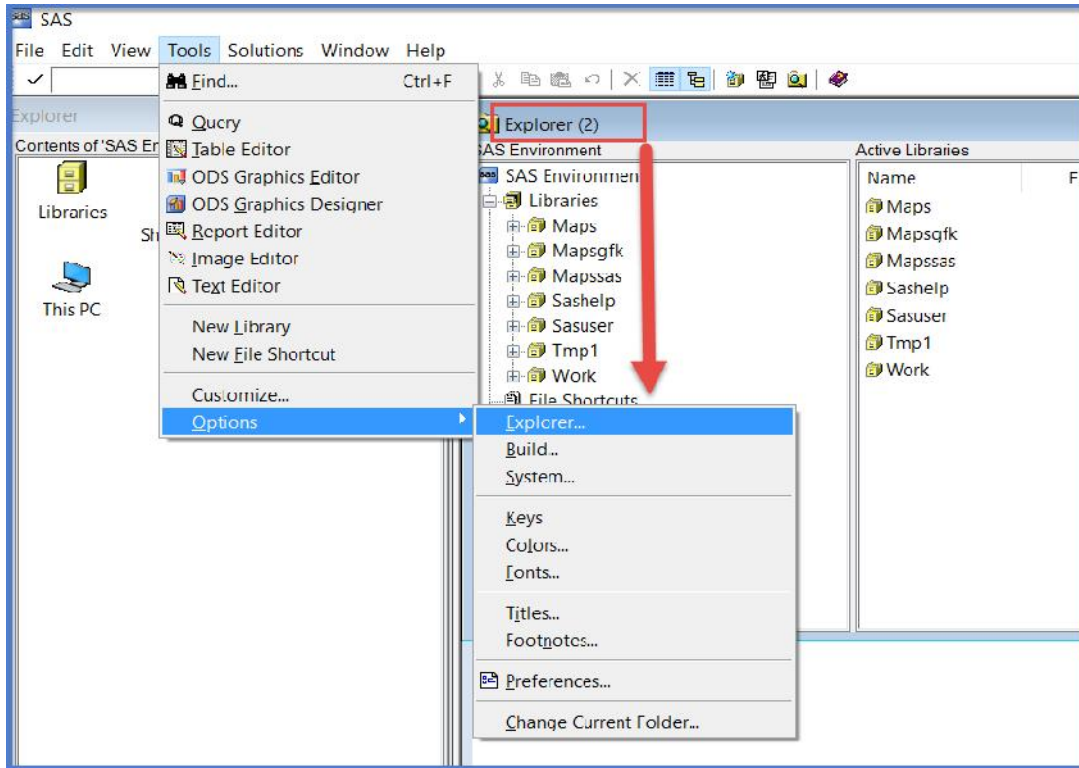


Figure 13. Opening Explorer option window from Tools.

Once the prompt window is open, go to **Members** tab and find **ITEMSTOR**, click **Unhide** at the bottom right corner (Figure 14). You may select *any items* that is present in the window to activate the unhide feature (Figure 15). Select **OK**.

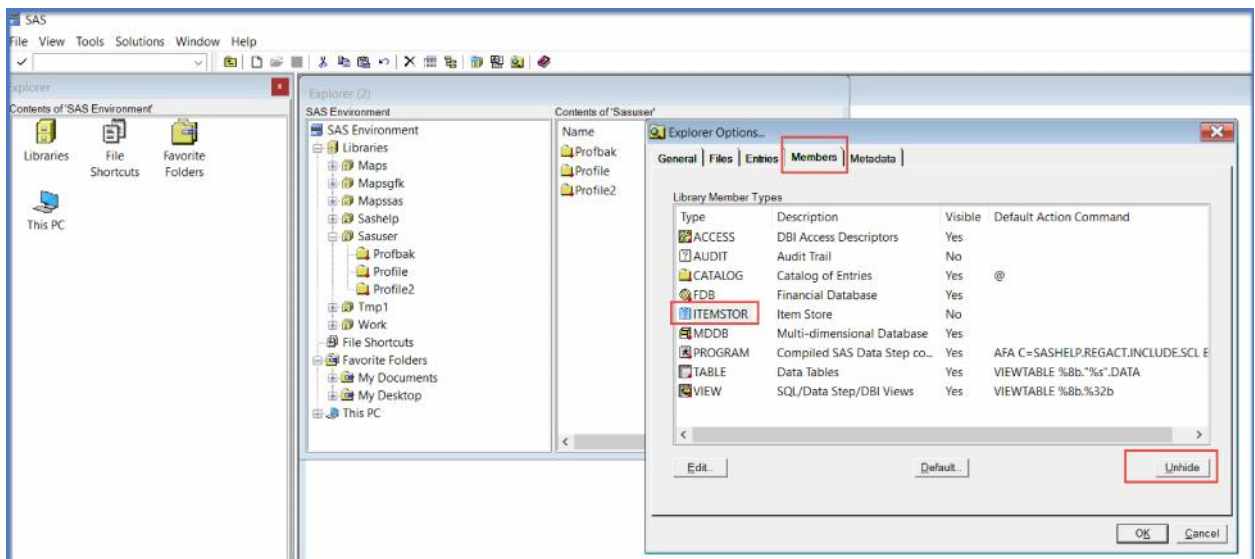


Figure 14. Edit Explorer Options.

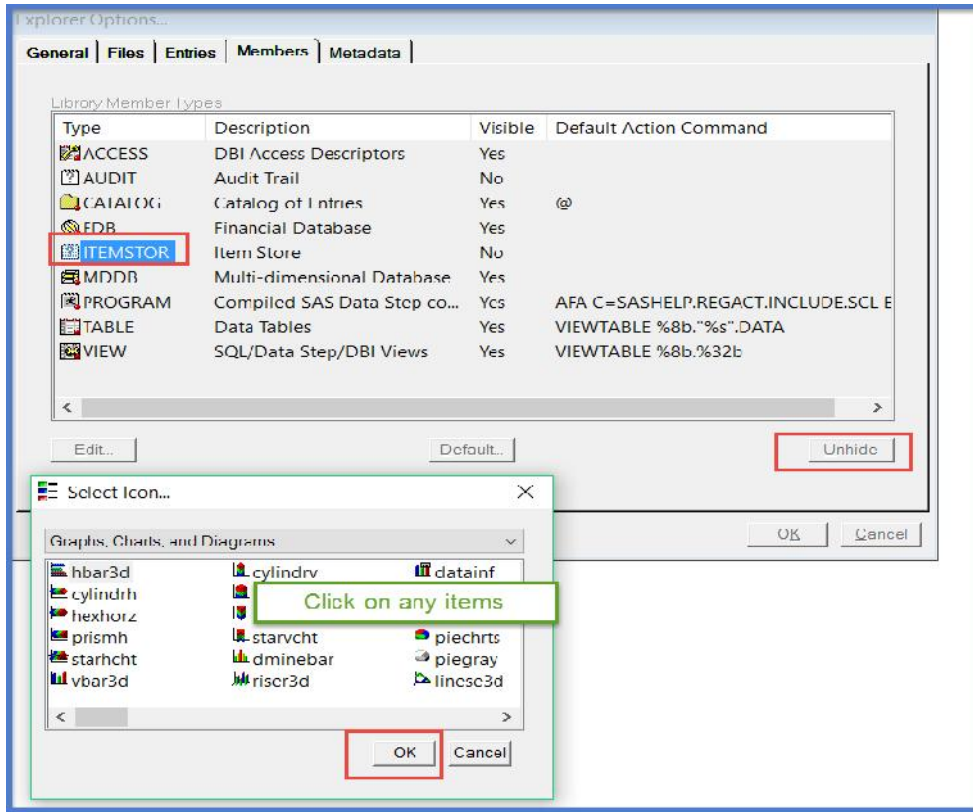


Figure 15. Select any item in ITEMSTOR to make it unhidden.

Now return back to the Explorer window and **double-click** on **Sasuser library**, you will see **Registry itemstor** file present in Sasuser library content window (Figure 16). Right-click and select **Copy**. (Figure 17)

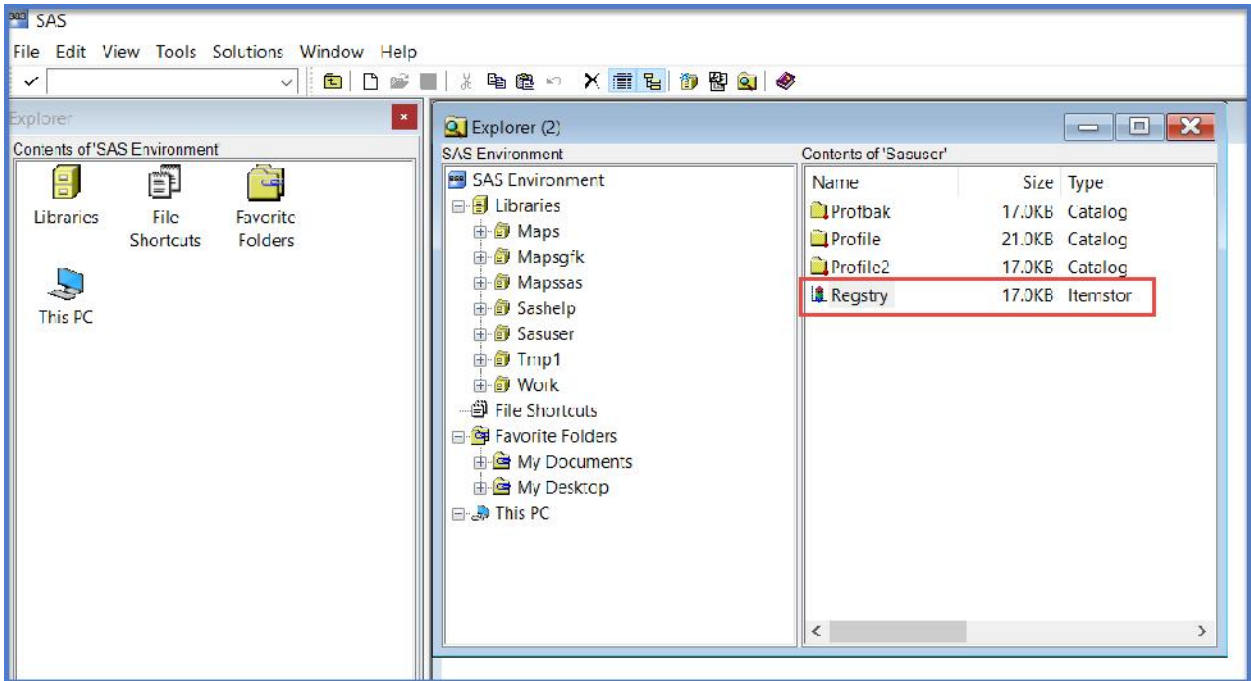


Figure 16. Registry Itemstor file is now showing in Explorer.

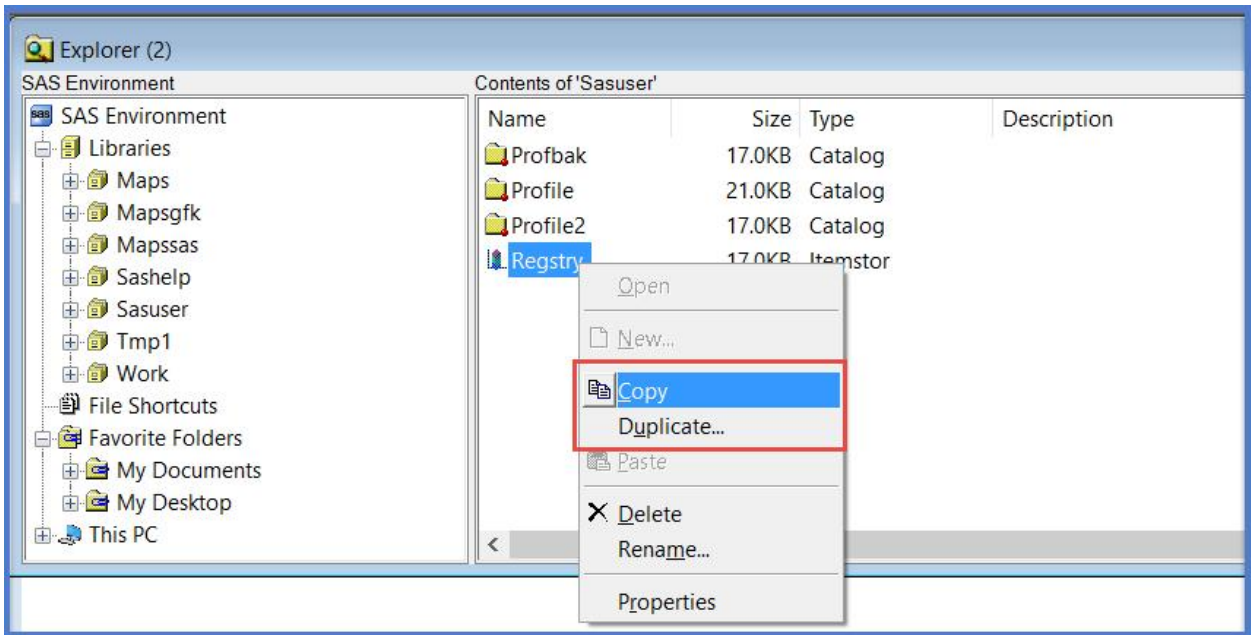


Figure 17. Make a copy of the Registry Itemstor file.

Once the registry file is copied, **right-click** again and hit **Duplicate** (this option is located right underneath copy in the previous step), you will see that SAS by default named this file as "Registry_Copy" (Figure 18). Click OK and the Registry_Copy will be saved in Sasuser library by default.

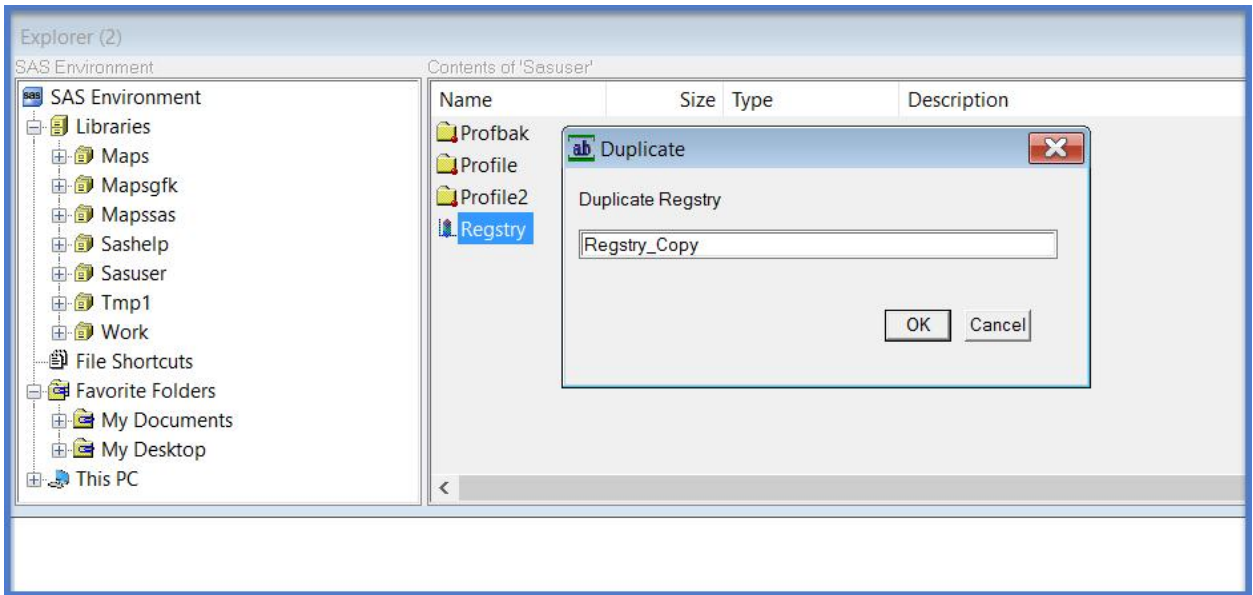


Figure 18. Duplicate the registry file.

2. Use SAS Registry Editor or PROC Registry to back up parts of the Sasuser registry that have changed
 - a. PROC Registry
 - b. Registry Editor

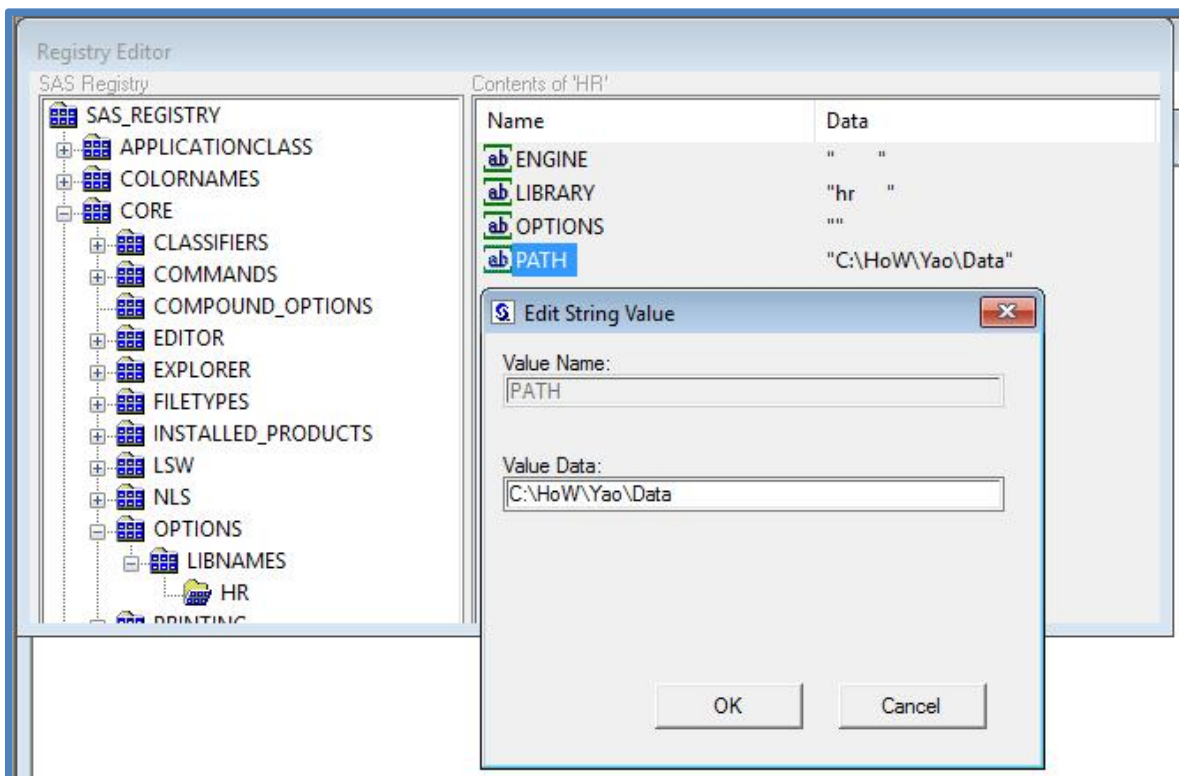


Figure 19. Changing a value in the Registry.

The simplest way to back up the registry is with PROC Registry ():

```
proc registry export='E:\backup\sas\registry20170301.txt';  
run;
```

By running this command, SAS will save any portions of changes made to the registry.

To restore the backup:

```
proc registry import='E:\backup\sas\registry20170301.txt';  
run;
```

When the restore process happens, the backup file is concatenated with the current SAS registry that incorporates any existing or new (if any) keys, subkeys or values that were added or changed to the SAS registry in the backup file values after the restore. ²

You can also Export/Import using the Registry Editor (Figure 20); using the Registry Editor to export will export the complete registry whereas PROC Registry exports only changes. Files exported through the Registry Editor can be imported through PROC Registry. This makes it useful to use one computer to set up a SAS 'look and feel', then export the settings to be imported to other computers.

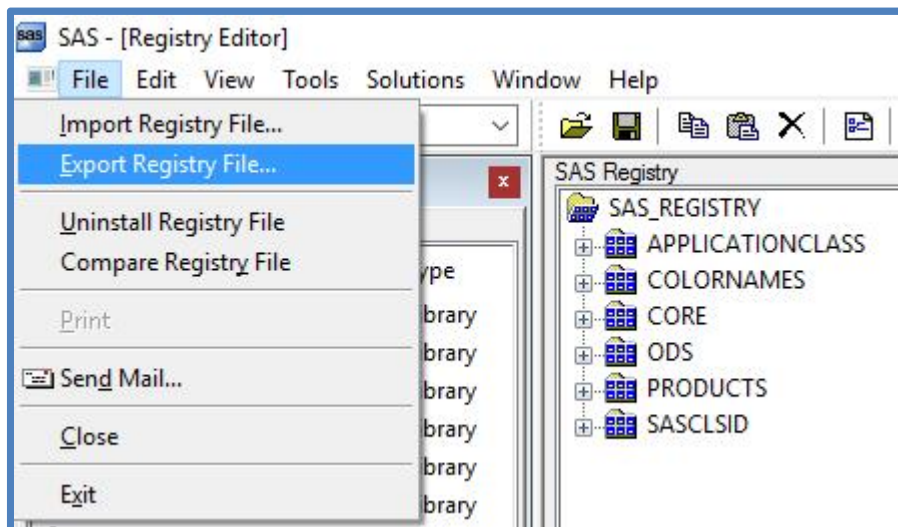


Figure 20. Export using Registry Editor

The SAS Registry is designed to be managed through the Display Manager interface, that is by making changes to settings in Display Manager rather than to the Registry directly. As a result, although it is possible to manually change the Registry, you should only do so when necessary.

² Chapter 14 The SAS Registry, SAS 9.4 Language Reference: Concepts, Third Edition

STEP 5 – THE COMMAND LINE REVISITED

As we saw in Figure 1, the command line specifies a configuration file to process at startup. What if no configuration file is specified? In what order does SAS process the startup files? SAS automatically looks in many places for a configuration file at startup. All steps in the following order are executed:

1. Check SAS_SYS_CONFIG Windows environment variable. If this variable is set, it must resolve to a valid SAS configuration file.
2. Files specified by -CONFIG command line switch. If you want to process multiple configuration files, each must have its own -CONFIG switch.
3. The SASV9.CFG in the folder where SAS.EXE resides, only if you have not specified a -CONFIG switch.
4. The SASV9.CFG in the Windows user profile folder, only if you have not specified a -CONFIG switch.
5. The SASV9.CFG in the current folder, only if you have not specified a -CONFIG switch.
6. SAS_USER_CONFIG Windows environment variable. As with SAS_SYS_CONFIG environment variable, it must resolve to a valid configuration file.
7. SAS_OPTIONS Windows environment variable. This variable can contain a list of valid SAS option, for example **-nodate -fullstimer**,
8. Options on the command line

As you can see, there are many ways to include a configuration file, and in some cases many different configuration files to be included. Because a SAS option can be set more than once, care must be taken to understand the order of processing noted above.

SYSIN

The SYSIN option takes a valid SAS program as its value. This option, which can only be set at invocation, tells SAS to execute the program in batch mode. This could be useful when you have programs to run that require no user intervention. When a program is run in batch mode, the .LOG (the SAS log file) and .LST (the SAS listing file) are located in the same folder as the .SAS program; all have the same name with different extensions.

CONCLUSION

This paper provided an overview of the files SAS uses to when starting, as well as the order of processing. For SAS options in particular it is important to note when they can be set; some options can only be set at invocation, which means the options must be set in a configuration file or on the command line. By setting different Windows desktop icons with different combinations of configuration and autoexec files, custom environments can set for different projects.

REFERENCES

SAS 9.4 Language Reference Concepts, Third Edition

Peter Eberhardt, Solving the 1,001-Piece Puzzle in 10 (or Fewer) Easy Steps: Using SASv9.cfg, autoexec.sas, SAS Registry, and Options to Set Up Base SAS®

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Peter Eberhardt
Fernwood Consulting Group Inc.
peter@fernwod.ca

twitter: @rkinRobin
WeChat: peterOnDroid
www.fernwood.ca

Mengting Wang
Qualcomm Inc.
mengting@qualcomm.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.