ABSTRACT

This paper will introduce the basic idea, usage, management and maintenance of central macro library. There are many reasons for a user or a group of users to build a central macro library. The essential reason could be that it can bring the important value of proposition to the user and customer.

A central macro library can be used to establish the productive environment across the business units or functionalities, to share the common programs, to set up the defined paths in production, to standardize the process or output, such as ODS template environment, to improve the quality and efficiency, and to meet the regulatory requirement.

Creation of a macro library could be easy. But, development of an efficient macro library and then management of the library could be very challenging. A good central macro library will bring you the power and flexibility. However, a bad macro library may bring you the duplication, confusion, inefficiency, or even error. This paper will discuss the best practice to develop an efficient central macro library.

We will discuss what the central macro library is, why the central library is needed, what value of customer proposition it can bring to the users, available types of macro in the library, and what the best practice is to build a central macro library.

In addition, we will discuss that after you build a central library, how to manage and maintain it. When you build a central library, or when you have to make changes to a built library, what problems or inefficiencies you would like to avoid? So, the problems will be avoided while the power of this Hercules will be maintained by all users.

INTRODUCTION

A macro library is a group of macros and programs that are stored and used by the user for the purpose of the convenient and collective programming. A central macro library is an authorized, approved and validated macro library within an organization that can be commonly used by multiple user groups to achieve standard process and consistent outputs in high quality and efficiency.

As the every company is trying to achieve the best outcome of quality and productivity from their programming regimen, for example all pharmaceutical institutions always want to present the best clinical data and get the approval from FDA, and to shorten the time from data collection to the final submission, the SAS® programmers at work wish they could as much the power as Hercules has to get everything correctly and quickly.

The establishment of a central macro library should contain the following essential parts: team building of a central macro library comprised of experienced programmers, identify the needs of macros or programs in the central library, design the program or macro, write the codes to accomplish the designed goal, validation of the program or macro, documentation of programs or macros in the library. This will be a good programming practice to ensure the creation of a desired macro library that meets the business needs.

A good central macro library will provide the programmers the neat tools or formidable weapons so it can equip you with a Hercules’ power. With this power, the programmers may be able to do anything they wish in the most efficient and effective way.

WHAT IS A CENTRAL MACRO LIBRARY USED FOR?

There are many benefits to have a macro library. A programmer may use a macro library to store any program or...
A macro library can save time and energy to avoid repeated recurrent programming efforts. Among those usages, the most appealing reasons are:

- **Standardize the programming process.** Many institutions, particularly in highly regulated business like pharmaceuticals, are looking for standardized process to meet internal and external regulation requirement. A central macro library can certainly help standardize the process.

- **Make program maintenances convenient.** With a central macro library, you only have to take care of one place which is shared by all other users in your organization. If you need to check a procedure, you need only validate once in the library and the accuracy will be applied everywhere else.

- **Improve the quality of programming.** With a validated and tested central library, the quality of work will be maintained at same high level across the programmers and projects.

- **Enhance the efficiency of programming.** A well-established central macro library can also enhance the efficiency. A central macro library usually contains many useful programs or macros that are available to perform the same routine functions, and to achieve the common goals for the users. So, the individual programmers do not have to write the codes individually.

- **Keep the outcome of programming consistent.** An important benefit of central macro library is that the consistency generated from programming. For instance, suppose you find an error or imperfect calculation in the programs, and you have 20 projects that are involved in the same situation. With a central macro library, you will not have to change 20 times in 20 places, rather, you only need to make one change in the central library, and keep other programs intact, and just run them to get the same correct results.

- **Enjoy the use of Hercules power in programming to make the most difficult calculation or derivation easy.** Many complicated calculation in SAS programming require each programmer’s a great amount of effort and time to finish. With selected and validated programs in a central macro library, the individual effort on such task can be saved.

- **Generalize the program as much as it can in a dynamic environment.** To save the time spent on programming and achieve the effective FTE results, the best generic programs are desired. In the central library, the individual programs are generalized usually in the form of macro. The individual programs are often project/or study specific while the macros are generalized so they can be applied to many different projects/or studies according to design.

- **Transfer the tacit individual programmer’s code into explicit company’s assets.** An extra benefit of central macro library is institutional knowledge retaining, transferring and sharing. For all programmers, the central macro library will make use of the group knowledge and expertise. Along with entire process of building the central macro library, the management works with programmers to identify the business needs and strategy, to convert the individual knowledge into institutional knowledge, and to transfer the past knowledge into new organizational knowledge, to deliver the prompt and quality results, and finally to achieve the best productivity and effectiveness. With an effectiveness-proven central macro library, the management can implement SOP and conduct critical programming training. All programmers of company wise can learn the experience and skill from reading and applying the macros/programs in the library.
WHAT IS A CENTRAL MACRO LIBRARY COMPRISED OF?

A central macro library should have a setup program, a format program, a group of standard process programs, and a group of utility programs. A setup program is created to set up the environment for SAS programming, such as the path/folder of SAS data sets and outputs to access, the general formats to be used, and the common macros to be included.

The standard process programs could be the data extraction, creation, statistics analysis and reporting, and tables, listing or graph generation.

The utility macros could be age calculation, missing data imputation, page counting, data or subjects pooling, converting data type, adding or dropping data columns or rows, etc.

A typical setup program in central macro library will be like this as follows:

```sas
%macro setup(study=);
  libname _all_ clear;
  libname study "J:\PROT\&study";
  libname AD "%sysfunc(pathname(study))\AD"; /*analysis data*/
  libname DATA "%sysfunc(pathname(study))\DATA"; /*raw data*/
  libname EXT "%sysfunc(pathname(study))\external"; /*external data*/
  %inc "%sysfunc(pathname(study))\Prgms\format.sas"/nosource2; /*study format*/
  %inc "%sysfunc(pathname(study))\Prgms\macros\pagebrk.sas"/nosource2; /*page break*/
  %inc "%sysfunc(pathname(study))\Prgms\macros\odson.sas"/nosource2; /*ods macros*/
  /*set out path*/
  %global outpath;
  %let outpath=%sysfunc(pathname(study))\OUTPUT\;
  options nodate nonumber nocenter missing=' ';
  /*output template*/
  proc template;
    define style tStyle/store=work.templ;
      parent=styles.rtf;
      style Table from output /
        just=center cellpadding=1pt
        background=_UNDEF_
        rules=all frame=box;
      style body from document /
        topmargin=.5in
        bottommargin=.2in
        leftmargin=1.in
        rightmargin=1.in;
      style SystemTitle from Titlesandfooters /
        font=('Times Roman',10.pt,Bold) ;
      style SystemFooter from Titlesandfooters /
        font=('Times Roman',10.pt);
      style data from data/
        font=('Times Roman',%sysfunc(sum(10,-1))pt);
      style header from header/
        font=('Times Roman',10.pt,Bold);
    replace headersandfooters from cell;
  end;
```
run;
%mend setup;

This setup macro assigns the LIBNAME to assess SAS data sets and output directories, to include the files under source location, and to create standard output templates.

ATTRIBUTES OF A GOOD CENTRAL LIBRARY

First, you need a stable and standard setup environment. For the SAS programming path or folders, including data sets and output directories, the structure has to be thoroughly elaborated and well established. Once the structure is determined, it should not be changed. With a constantly changing structure, neither of central macro library or individual programming approach will work. The method of %include will suffer the most because the physical file does not exist!

Second, all programs or macros in the central library should be sufficiently tested and validated against the real data. Every single macro or program enters the library only after it is fully validated. You have to make sure the macro in library can perform consistently in all possible situations, and achieve the desired the correct results.

Third, all files or programs in the library are clearly documented. A reference or manual should be created. The purpose, parameters, modification all will be explained in detail. We should also have a good version control of macros being modified and updates.

Fourth, the maintenance and management of a central library are expected to be carefully defined. Although all programmers in the organization are supposed to have read access to the central library, the write access should be assigned to designated personnel or group.

There are three popular methods to refer to the central macro library: AUTOCALL, Compiled Stored Macro, and %INCLUDE. I think %INCLUDE may be the best method in practice because it is explicit and it reflect any updates promptly. The only problem with %INCLUDE is when the structure changed or the files to be included are moved. But, we have already declared the assumption of standard and stable structure.

PROBLEMS OR ERRORS A MACRO LIBRARY SHOULD AVOID

If we do not have the above attributes of a good central library, we may lose the Hercules power of macro library, or the macros and programs may become monsters in practice. We may get unexpected results, or hidden errors that are difficult to detect.

We can list some rules for programmers to follow. For instance, do not create or store any macros with same name, or same macro with different names. This problem will certainly cause confusion and unexpected results that is extremely dangerous.

Do not create any overlapped macro or programs. This type of macro may cause confusion and it is difficult to maintain. If you have to change one but forget to change another, you may be confused by the different results at the end.

For example, two overlapped programs are saved in the central library. One calculates P-values with interaction term, the other calculates P-values without interaction.
The second macro better be separated so to keep output clear and maintenance easy. In a situation you made an adjustment to the model and forgot to make the same adjustment in the other macro, and then you will have a hard-to-detect error in your final report.

CONCLUSIONS

Like Hercules, a central macro library, created by SAS programmers who are passionate and intelligent individual, is capable of doing both great deeds for our work and being a terrible monster who would wreak horrible mess in our programming world. By utilizing its Hercules-like power of a central library, we can make our programmers’ world better and fruitful. With a tested and approved good central macro library, you can do our work with ease, comfort and confidence.
ACKNOWLEDGMENTS

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.

CONTACT INFORMATION

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

Yajiang He
Teva Pharmaceuticals
425 Privet Road
Horsham, PA 19044
Work Phone: 610-747-2695
Fax: 610-747-2980
Email: jhe@barrlabs.com