1. Introduction

You have just been told that the data has changed for one dataset and you need to know which program use it so they can be run. You have finished all your programs and would like to do a final check on the log files to make sure that there is nothing in them that should be run.

Both these are situations where you need to be able to quickly search some files— you may have a search facility to do this easily if not not then how? You already have SAS so is there not a way you can use that?

The following is one way of doing it: written as a macro. using the file handling functions within SAS, you just pass in the directory to be searched, the file type to be looked at and optionally what to look for and the type of report you want produced at the end.

3. Read the directory

This is the directory that you will search through and find out how many files are in there. Loop through the full of files, checking if they are of type you want to check. If the file extension matches the type that you want to search then output the filename.

```
data data_dir;
  set list_dir;
  if ext eq &file_ext then output;
  run;
```

5. What to search for

In order to decide if the text you are looking for in the file or not each line of the needs to be checked to see if the text is in the line. The Index function is used on the variable that contains the line of the file. If the text is found then the line is output. Before you can do this you need to know what text you are searching for. If no text has been passed in then the code will look for the default, in the example given here this is things that you do not want in the log files.

If you have passed in what you are going to search for then you need to know how many lines of text you are looking for. This is found by if the parameter string is greater than one then replacing the ’ ’ with , then using do while loop and the scan function to add to one on the counter every time a new word is found. The final count is put into a macro variable which is then used in the next dataset.

```
%let &numwrd = 0;
%do %while ((lengthn(wrd_lst) > 0) && (i < &numwrd));
  ... the code to find the text is here ...
  ... outputting the result ...
  i = i + 1;
%end;
```

11. Calling the Macro

Now you have the macro just pass in the path, what you are looking for and run the code. So to list all the SAS programs in asset/ bbble/bcll/programmd that use the SSV/SAV database use the following call:

```	klist dir = asset/bbble/cc/or/com/programmd; ext=sas, scan_ext = sas, scan_str = asset/bbble/cc/or/com/programmd; report = short;
```

To search through the log files in a given directory for WARNING and ERROR, producing both reports use:

```	klist dir = asset/bbble/cc/or/com/programmd; ext=log; scan_ext = sas, scan_str = asset/bbble/cc/or/com/programmd; report = short, error = error, warning = warning, report = short;
```

4. Read in the files

Use the data set of files you created above read in the contents of the file to a dataset. Each line is read in an observation. You will need to add the directory name to the file and check the return codes to make sure that it is all going plan.

```
data read_file;
  set list_dir;
  length f_line 200;
  file_name = dread(did,i);
  if fileexist(file_name) then output;
  run;
```

8. The Long Report

Use PROC REPORT to create a report. There are only three columns to put out, the file that something was found in, what was found and the line it was found in. Use an order option on the define statement for the file name so it is easy to see which files have something found in them.

```
title "The following have been found in &dir";
task = data zupeł; nfound = count;
```

9. Setting the Short Report

The short report is very similar to the long report, the difference is that when sorting the data sort it based on the file name and what was found otherwise duplicated bases on this sort selection.

Check how many observations are in the resulting dataset, putting the number into a macro variable and as before if no observations were found create a blank observation.

```
%if nobs = 0 then output; %end;
```

10. The Short Report

The PROC REPORT this time only has two variables, the file name that something was found in and what was found. Again put an option order on the define statement for the file name, and use a compute block to put out a message if nothing was found.

```
title "The following have been found in &dir";
task = data seule; nfound = count;
```

12. And finally

This code was developed using SAS Enterprise Guide on a PC submitting the code to SAS on a Solaris server.

The program, along with several that done similar things, is stored in an Enterprise Guide project, so if the question is asked then I can just start the project, open the file and pass in the parameters.

Equally the code can also be called into any other session and ran that way.

Contact details: Helen Nicholson, Cmed Clinical Research Services Ltd, Holmwood, Broadbands Business Campus, Langhurstwood Road, Horsham, West Sussex, RH12 4QP, UK
Email: hnicholson@cmedresearch.com