

Need to Review or Deliver Outputs on a Rolling Basis? Just Apply the Filter!

Tom Santopoli, Accenture, Berwyn, PA

ABSTRACT

Wouldn't it be nice if all of the outputs in a deliverable passed QC at exactly the same time and could be submitted for final review all at once, while leaving adequate time for the review to be completed prior to a deadline? Unfortunately, outputs often pass QC at different times and must be submitted for final review on a rolling basis in order to meet deadlines. The task of selecting 50 specific outputs to copy from a folder containing 500 outputs can be very tedious and cumbersome for a lead programmer. There are papers that explain how to copy files from one folder to another, but this paper addresses the issue of selectively copying specific files based on criteria set in a project tracking document. A simple macro called %MFILTER is presented to help make life a little easier for lead programmers as deadlines approach.

INTRODUCTION

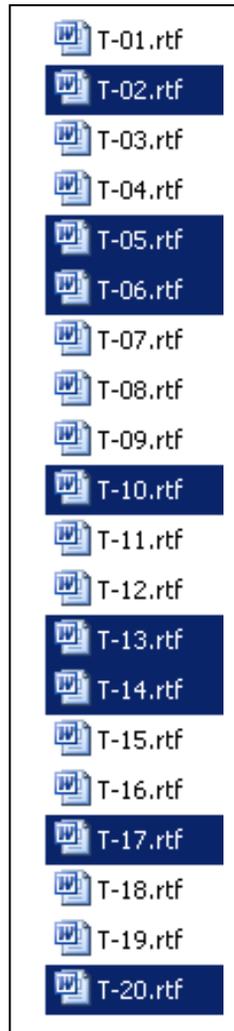
When the programming project in a clinical trial approaches various deadlines, the outputs must typically pass an internal review process prior to delivery. This process may include review by the lead programmer, review by the statistician, etc. Outputs deemed ready for review are often copied to isolated folders to ensure that outputs approved for delivery will not be accidentally overwritten. A project tracking document is a good place to store information on the status of each individual output.

Diagram 1

A Output	B Title	C Topline	D Lead Review	E Stat Review	F Delivery
T-01	Summary of Populations	Y			
T-02	Summary of Populations - Subgroup 1		Y		
T-03	Summary of Populations - Subgroup 2	Y			
T-04	Summary of Populations - Subgroup 3				
T-05	Summary of Adverse Events	Y	Y		
T-06	Summary of Adverse Events - Subgroup 1		Y		
T-07	Summary of Adverse Events - Subgroup 2	Y			
T-08	Summary of Adverse Events - Subgroup 3				
T-09	Summary of Labs	Y			
T-10	Summary of Labs - Subgroup 1		Y	Y	Y
T-11	Summary of Labs - Subgroup 2	Y			
T-12	Summary of Labs - Subgroup 3				
T-13	Summary of Vitals	Y	Y	Y	Y
T-14	Summary of Vitals - Subgroup 1		Y	Y	
T-15	Summary of Vitals - Subgroup 2	Y			
T-16	Summary of Vitals - Subgroup 3				
T-17	Summary of ECG	Y	Y	Y	
T-18	Summary of ECG - Subgroup 1				
T-19	Summary of ECG - Subgroup 2	Y			
T-20	Summary of ECG - Subgroup 3		Y		

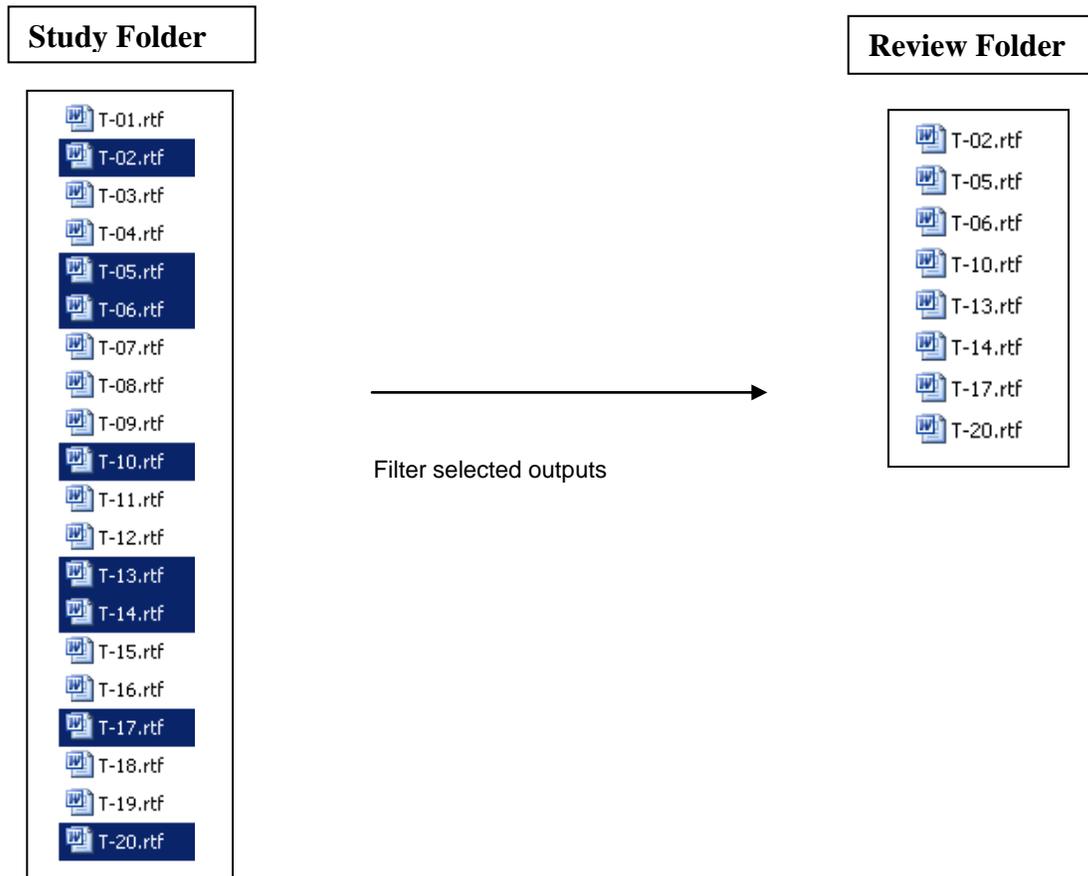
In the sample tracking document above, the last three columns indicate whether or not the output is ready for lead review, statistical review, and delivery, respectively. There is also a column to indicate topline outputs of high priority. The diagram of the folder shown below lists the twenty outputs in the study with the eight outputs that are currently ready for lead review highlighted:

Diagram 2



An actual study folder may contain 400 outputs, 80 of which are currently ready for lead review. The statistician cannot be expected to review all 400 outputs in one day. So, the review will have take place on a rolling basis. The lead programmer would copy the 80 outputs that are currently ready for review to the lead review folder. As illustrated, the outputs currently ready for review are in random sections of the folder. The lead programmer would have to check the tracking document, visually locate each output in the folder, and then click copy and paste every single time. What is needed is a macro that can apply a filter to copy the appropriate outputs to the review folder, as shown below:

Diagram 3



HOW %MFILTER WORKS

%MFILTER accepts three parameters:

1. INPATH - specifies the folder that contains the outputs to be filtered
2. OUTPATH - specifies the folder to which the filtered outputs are to be copied
3. CONDITION - specifies the filter criteria

The following call to %MFILTER will select the outputs indicated as ready for lead review in the tracking document above and copy them from the folder labeled 'Production' to the folder labeled 'Lead Review':

```
%mfilter(inpath=F:\Study Folder\Production,  
         outpath=F:\Study Folder\Lead Review,  
         condition=%str(lead_review='Y'));
```

The tracking document in this example is an excel spreadsheet where the column labeled 'Lead Review' indicates that an output is ready for lead review. Any criteria desired may be specified in the CONDITION parameter. For example, only outputs that are both topline and ready for review may be filtered. The application of %MFILTER may be customized to the needs of a particular study or the specific internal review process at a given organization.

The following is the SAS® code for %MFILTER:

Need to Review or Deliver Outputs on a Rolling Basis? Just Apply the Filter!, continued

```

%macro mfilter (inpath=,outpath=,condition=);

* bring in tracking document ;

proc import
  dbms=xls
  datafile="F:\Documentation\Track.xls"
  out=work.track replace;
  getnames=yes;
  sheet="TASKS";
run;

* apply filter and copy outputs ;

data _null_;
  set track;
  where &condition;
  call execute(cat("systask command 'copy ",'"',
                  "&inpath\",strip(output),'.rtf" "',
                  "&outpath\",strip(output),'.rtf"',
                  "';"));

run;

%mend mfilter;

```

The first step is to bring the project tracking document into a SAS Data Set. Whatever the application used, the project tracking document may be brought into a SAS Data Set. In this example, the project tracking document is in an excel spreadsheet, so a standard PROC IMPORT is used to import the document into the SAS Data Set TRACK.

**Table 1:
SAS Data Set TRACK**

	Output	Title	Topline	Lead_Review	Stat_Review	Delivery
1	T-01	Summary of Populations	Y			
2	T-02	Summary of Populations - Subgroup 1		Y		
3	T-03	Summary of Populations - Subgroup 2	Y			
4	T-04	Summary of Populations - Subgroup 3				
5	T-05	Summary of Adverse Events	Y	Y		
6	T-06	Summary of Adverse Events - Subgroup 1		Y		
7	T-07	Summary of Adverse Events - Subgroup 2	Y			
8	T-08	Summary of Adverse Events - Subgroup 3				
9	T-09	Summary of Labs	Y			
10	T-10	Summary of Labs - Subgroup 1		Y	Y	Y
11	T-11	Summary of Labs - Subgroup 2	Y			
12	T-12	Summary of Labs - Subgroup 3				
13	T-13	Summary of Vitals	Y	Y	Y	Y
14	T-14	Summary of Vitals - Subgroup 1		Y	Y	
15	T-15	Summary of Vitals - Subgroup 2	Y			
16	T-16	Summary of Vitals - Subgroup 3				
17	T-17	Summary of ECG	Y	Y	Y	
18	T-18	Summary of ECG - Subgroup 1				
19	T-19	Summary of ECG - Subgroup 2	Y			
20	T-20	Summary of ECG - Subgroup 3		Y		

The second step is a Data Step to apply the filter. The CONDITION parameter is inserted into the WHERE statement so that only the observations representing the outputs to be copied are read. CALL EXECUTE is a powerful function that can be used to execute specific code for each observation in a Data Set. In this case, CALL EXECUTE is used to execute the SYSTASK command for each output that was selected by the WHERE statement.

Need to Review or Deliver Outputs on a Rolling Basis? Just Apply the Filter!, continued

The SYSTASK command is applied to copy the outputs from the input path to the output path. A simple example of a SYSTASK command that would copy T-02.RTF from H:\Lead to H:\Stat is as follows:

```
systask command "copy H:\Lead\T-02.RTF H:\Stat\T-02.RTF";
```

A blank space is the delimiter between the input path and the output path. In study folders, however, there are often blank spaces in path names. For example, suppose the input path is the following location:

```
H:\Company A\Panacea\Lead\T-02.RTF
```

The blank space between 'Company' and 'A' would be interpreted as the delimiter between the input path and the output path, which would present a problem. The syntax to include blank spaces in path names is as follows:

```
systask command 'copy "H:\Company A\Drug Cureall\Lead\T-02.RTF"  
"H:\Company A\Drug Cureall\Stat\T-02.RTF"';
```

Here, the input path and the output path are each enclosed in double quotes. The entire SYSTASK command must also be enclosed in quotes, so single quotes are used to enclose the double quotes. This creates another complication:

```
systask command 'copy "&INPATH\T-02.RTF" "&OUTPATH\T-02.RTF"';
```

The macro variables &INPATH and &OUTPATH are specified by parameters in the call to %MFILTER and must be resolved. Double quotes are required to resolve macro variables. However, &INPATH and &OUTPATH would not resolve in this example because the double quotes are enclosed in single quotes. To get around this, the following code would have to be applied:

```
%let T=%str(%) copy "&INPATH\T-02.RTF" "&OUTPATH\T-02.RTF"  
%str(%) ;  
  
systask command %unquote (&T) ;
```

%STR(%) masks the single quotes, allowing the macro variables inside the double quotes to resolve. The complete statement after &INPATH and &OUTPATH have resolved is stored in the macro variable &T. Then, %UNQUOTE un masks the single quotes so that the SYSTASK command in the second line can execute.

The good news is that the CALL EXECUTE in %MFILTER circumvents the need to apply macro quoting functions. The statement to execute the SYSTASK command with CALL EXECUTE is assembled using the CAT function:

```
call execute(cat("systask command 'copy ", "'",  
                "&inpath\"", strip(output), '.rtf" "',  
                "&outpath\"", strip(output), '.rtf"',  
                "';"));
```

The CAT function concatenates the individual components that are separated by commas. The macro variables resolve as the components are being concatenated. By the time the SYSTASK command is ready to execute, the macro variables have already resolved. The RTF extension is concatenated to the end of the output name to correctly specify the appropriate outputs in the folder. CALL EXECUTE can then fulfill the purpose of executing the SYSTASK command to copy the outputs from the input path to the output path. Each output whose name is contained in the values of OUTPUT that were selected by the WHERE statement will be copied from the input path to the output path at their respective observations in the Data Set.

Note that the quoting in the CAT function is tricky. Components that represent literal values must be enclosed in quotes. However, some components contain single and double quotes that are part of literal values. Components containing single quotes that are part of literal values must be enclosed in double

Need to Review or Deliver Outputs on a Rolling Basis? Just Apply the Filter!, continued

quotes, and components containing double quotes that are part of literal values must be enclosed in single quotes. For example, the first component to be concatenated is the following:

```
systask command 'copy
```

This component contains a single quote that is part of a literal value. So, this component must be enclosed in double quotes:

```
"systask command 'copy "
```

The next component to be concatenated is the first double quote that encloses the input path. This double quote is a literal value and must be enclosed in single quotes. Of course, the macro variables &INPATH and &OUTPATH must be enclosed in double quotes in order to resolve. The variable OUTPUT is not a literal value and is not enclosed in any quotes.

CONCLUSION

The %MFILTER macro is a powerful snippet of code that can help save significant time when preparing a project deliverable. Specific outputs in a study folder can be conveniently filtered with the click of a button, which dramatically simplifies the task of juggling the many moving parts associated with a large review. The functionality of %MFILTER may also be expanded to include the filtering of specific programs in a study folder. In addition, %MFILTER may be used in conjunction with automated tracking to establish the project tracking document as the centralized location to obtain accurate, reliable information regarding the status of a project (see Santopoli, Zhong (2013)).

REFERENCES

1. Tom Santopoli, Wayne Zhong. 2013. Let SAS Set Up and Track Your Project. PharmaSUG 2013.
2. Irina Walsh. 2009. Pros and Cons of X Command vs. Systask Command. WUSS 2009.

ACKNOWLEDGMENTS

I would like to thank Vivek Mohan and Wayne Zhong for providing valuable feedback on the testing of %MFILTER.

CONTACT INFORMATION:

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

Tom Santopoli
Accenture
1160 W. Swedesford Rd. Bldg. One, Berwyn, PA
(610) 407-7583
thomas.r.santopoli@accenture.com
www.accenture.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 registered trademarks or trademarks of their respective companies.