

A Batch Processing Companion, how to write Windows *.bat and *.cmd files for my-program.sas

Ronald J. Fehd, senior maverick, theoretical programmer,
Fragile-Free Software Institute

Abstract **Description :** This paper review issues in writing Windows batch and command files for processing SAS® software *.sas programs.

purpose : The purpose of this paper is to provide programmers and users information about Windows operating system environment variables and how to use them to write a .cmd file: sas.cmd, which creates a date+time-stamped .log file.

audience : programmers of all levels

keywords : *.bat, *.cmd, sas.exe

In this paper:	Setup: change format of date and time	2
	DOS commands and functions	2
	Saving date+time-stamped altlog: sas.cmd	4
	Regression testing: execute *.bat	7
	Conclusion	8
	References	9

List of programs:	1	Windows sas.cmd and *.bat	2
	2	Windows change data format menu	2
	3	demo-evar-assign-and-reference.cmd	3
	4	demo-evar-translate.cmd	4
	5	demo-evars-system.cmd	4
	6	sas.cmd	5
	7	demo-date-time-to-seconds.cmd	6
	8	my-program.bat	6
	9	my-program.sas	6

10	demo-dos-dir.cmd	7
11	08-exec-all-bat.cmd	7

Introduction

Once upon a time at a Code Doctor session a question came up for us:

How can we add a date+time stamp to our SAS program .log?

Where our past practice was to write a sas.cmd file and my-program.bat file with just these lines in each file:

Program 1 Windows sas.cmd and *.bat

```
rem sas.cmd
"C:\...\sas.exe" %*

rem my-program.bat
sas my-program

sas -sysin job -log job.log -print job.lst
```

In order to create a filename this task requires three variables: job name, date+time-stamp, and userid.

Setup: change format of date and time

The first task is to change the format of the date part of the date-time stamp. The default is mm-dd-ccyy. We want to change that to ccyy-mm-dd. This section reviews the Windows 10 settings for date and time. To get to this menu, go to Start, Settings, Time & Language, date, time and regional formatting. At bottom of page choose: Change data formats

Program 2 Windows change data format menu

```
Change data formats
  calendar:  Gregorian calendar
first day of week:  Sunday
  short date:  04-05-2017           <---<<<
    long date:  Wednesday, April 5, 2017
    short time:  09:40 AM
    long time:   09:40:07 AM
```

```
change short date to 2017-04-05
```

DOS commands and functions

overview

This section reviews the set of dos commands used in this paper.

This section has these topics.

- dos command list
- set environment variables
- function translate

- system environment variables
-

dos command list

To get to the DOS prompt, go to Start, command.

Help for each of these commands can be viewed by typing:

```
help <command-name>
```

```
call : used to call subroutines, which must end with goto eof;
      similar to SAS link :label with return;
del : delete or remove a file
dir : directory listing; option /b (bare) removes the header and footer to
      produce just a list of filenames without date+time and size information
echo : writes arguments to screen, or with redirection symbol (>) to file
findstr : find string: find occurrences of one or more 'words' within a file and
          write to screen or output file; option /i means 'ignore case'
for : an iterative loop for each item parsed from output of command
      note: the syntax for the %~ functions are in this help file
goto : :label; compare to call
pause : during the program execution;
        used while testing to review results
rem : remark: comments
set : allocate an environment variable
type : display contents of a file to the screen
```

set environment variables

Program 3 shows the syntax of the allocation verb `set` for environment variables (evars). Evar names may begin with special character underline (`_`). Names may be in camelCase, PascalCase, kebab-case, or snake_case. The verb `set` followed by the evar name writes the value to the screen.

References to evars are enclosed in percent signs: `%evar%`; the example written to file — `>%out%` — shows concatenation with no space and with a special character: colon.

Program 3 demo-evar-assign-and-reference.cmd

```
set out=demo-evar-assign-and-reference.txt
set abc=123
set abc
set _def=456
set _def
set g-h-i=7.8.9
set g-h-i
echo %abc%_%_def%:%g-h-i%>%out%
type %out%
pause
```

```
.txt 123456:7.8.9
```

function translate

Program 4 shows the syntax of the DOS translate function as well as the write-to-file (>) and append-to-file (>>) characters.

Program 4 demo-evar-translate.cmd

```
set out=demo-evar-translate.txt
echo . . . . . : %time% >%out%
echo translate space to zero.. : %time: =0%>>%out%
echo translate colon to hyphen: %time: :=-%>>%out%
type
pause
```

```
. . . . . : 9:16:44.42
translate space to zero.. : 09:16:44.43
translate colon to hyphen: 9-16-44.44
```

notes: .42 the reference to the time function returns at least a +0.01 second difference in each call

system environment variables

Program 5 shows the list of system environment variables.

Program 5 demo-evars-system.cmd

```
set job=demo-evars-system.txt
set >%job%
type %job%
pause
```

```
APPDATA=C:\Users\User\AppData\Roaming
COMPUTERNAME=DESKTOP-QZOFEKHT
ComSpec=C:\Windows\system32\cmd.exe
job=demo-evars-system.txt
OS=Windows_NT
Path=C:\program-files\texlive\2023\bin\windows;
      C:\Windows\System32\WindowsPowerShell\v1.0\;
      C:\Program Files\IDM Computer Solutions\UltraEdit;
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
ProgramData=C:\ProgramData
PROMPT=$P$G
SESSIONNAME=Console
SystemDrive=C:
TEMP=C:\Users\User\AppData\Local\Temp
USERNAME=RonFehd
```

notes: later we will use USERNAME in the name of the log file;
note that case does not matter: username

Saving date+time-stamped altlog: sas.cmd

overview

This section has these topics.

- sas.cmd
 - granularity: date+time to hh:mm:ss.99
 - my-program.*: .bat, .sas
 - results of my-program.bat
-

sas.cmd

Program 6 contains the following steps:

rem : remarks; it's left to the user to turn off echo by removing @rem:
@echo off
allocations : of environment variables used in executing *.sas programs and creating a date+time-stamped log file
submit : %sas%
%* is DOS syntax for: *pass all (*) parameters from calling program*
find e | w : the findstr functions parses each line in a file for the word(s) in double quotes; output, if any, is written to file;
if the file is empty (size=0) then it is deleted
subroutines : FileDateTime and FileSize

Program 6 sas.cmd

```
@rem @echo off
rem * name: sas.cmd, parameter: job
set sas="C:\program-files\SASHome\SASFoundation\9.4\sas.exe"
rem * allocate file-date-time, replace space w/hyphen
echo %date%+%time%>z.log
call :FileDateTime file-d-t z.log
rem ** translate space and colon to hyphen
set file-d-t=%file-d-t: =-%
set file-d-t=%file-d-t::=-%
rem * allocate name of .log
set out=%job%_%file-d-t%_%username%
del %job%_*.txt
rem * allocate file with errors or warnings
set out-e-w=%out%-log-issues.txt
rem * allocate directory for altlog
set alt-dir =..\altlog\
set alt-dir =
@echo on
%sas% %* -log %job%.log -altlog %alt-dir%%out%.log
rem * does log have error | warning? if not then delete
findstr /i "error warning" %job%.log>%out-e-w%
call :FileSize file-size %out-e-w%
if %file-size% ==0 del %out-e-w%
goto :eof
rem * subroutine FileDateTime: parm1= datetime(parm2)
:FileDateTime
set %~1=%~t2
goto :eof
rem * subroutine FileSize: parm1= filesize(parm2)
:FileSize
set %~1=%~z2
goto :eof
:eof
```

**granularity: date+time
to hh:mm:ss.99**

Program 7 shows commands used to allocate an environment variable with a date+time stamp accurate to hundredths of a second: hh:mm:ss.99. Note that the granularity of the date+time stamp of program 6, `sas.cmd`, is hh:mm.

Program 7 demo-date-time-to-seconds.cmd

```
set out=demo-date-time-to-seconds.txt
echo %date%+%time% > %out%
set file-d=%date%
set file-t=%time%
rem translate space to zero
set file-t=%file-t: =0%
echo %file-t% >>%out%
set file-d-t=%file-d%_%file-t%
rem translate colon and dot to hyphen
set file-d-t=%file-d-t:=-%
set file-d-t=%file-d-t:.-%
echo %file-d-t% >>%out%
type %out%
```

```
.txt
2023-09-31+ 8:13:21.34
08:13:21.34
2023-09-31_08-13-21-34
```

**my-program.*: .bat,
.sas**

Program 8 shows the allocation of environment variable `job`, which is the parameter for program 6, `sas.cmd`.
! → Note carefully the naming conventions used here:
command files for `*.sas` are `*.bat`;
command files demonstrating Windows and DOS statements are `*.cmd`.

Program 8 my-program.bat

```
set job=my-program
sas %job%
```

Program 9 is a demonstration program for program 8, `my-program.bat`.

Program 9 my-program.sas

```
proc print data = sashelp.class;
run;
```

**results of
my-program.bat**

```
dir my-program.*

2023-09-09 11:56      31 my-program.bat
2023-09-13 11:19     660 my-program.log
2023-09-13 11:19   1,802 my-program.lst
2023-09-12 08:44      40 my-program.sas
2023-09-13 11:19     660 my-program_2023-09-31-11-19_RonFehd.log
```

Regression testing: execute *.bat

overview

Once we have modified any one of the programs in a project directory we want to submit all of them to check that modifications to one program have not affected others. This process is called *regression testing*. This section addresses that question:

How can we submit all the programs in a directory?

This section has these topics.

- demo DOS dir command
- execute *.bat

demo DOS dir command

Program 10 illustrates the output of the DOS `dir` command. The option `/b` removes the header, footer, and file information. This bare list of files is the input to the `for` command in program 11, `08-exec-all-bat.cmd`.

Program 10 demo-dos-dir.cmd

```
set out=demo-dos-dir.txt
dir *.cmd >%out%
echo * * * * >>%out%
dir /b *.cmd >>%out%
type %out%
```

```
Volume in drive C has no label.
Volume Serial Number is ABC01-9876
```

```
Directory of C:\SAS-projects\2023-batch-companion\dos
```

```
2023-09-11 09:38 181 08-exec-all-bat.cmd
2023-09-13 09:08 121 demo-dos-dir.cmd
2023-09-12 12:02 1,143 sas.cmd
* * *
08-exec-all-bat.cmd
demo-dos-dir.cmd
sas.cmd
```

execute *.bat

Program 11 produces a list of all the batch files (*.bat) for complete SAS software programs (*.sas), and calls each file.

! → Note carefully that the previous record of execution is deleted.

Program 11 08-exec-all-bat.cmd

```
rem name: 08-exec-all-bat.cmd
set jobname=08-exec-all-bat
for /f %%i in ('dir /b *.bat') do call %%i
del %jobname%*.txt
dir *.log>%jobname%_%date%.txt
rem
pause
```

notes: for iterative loop

- `/f`: passes the first blank-separated token from each line of file
- `%%i`: name of evar parsed
- `in (...)`: list of files: directory, bare, of *.bat
- `do`: call each batch (*.bat) file, value of evar %%i is <filename>.bat

Conclusion

Writing *.bat and *.cmd files are relatively easy, once one becomes familiar with the basic set of commands and functions to create, modify, and reference environment variables.

future work

Consider writing a file `09-history.csv`, which will contain the date+time stamp of each program submitted.

Suggested reading

companions : Fehd, "An Autoexec Companion, Allocating Location Names during Startup",
Fehd, "A Configuration File Companion: testing and using environment variables and options; templates for startup-only options `initstmt` and `termstmt`",
Fehd, "A Sysparm Companion, Passing Values to a Program from the Command Line",
Torres, "Customized Output with the SAS(R) Config File: Always Useful but Seldom Used"
DOS : *DOS commands: List, DOS: environment variable, DOS: redirection, DOS time command: replace leading space with zero, DOS command: call, DOS command: goto*
wikipedia : *definition: regression testing*

Author information

Ronald J. Fehd

Ron.Fehd.macro.maven at gmail dot com

LinkedIn
affiliation

<https://www.linkedin.com/in/ronald-fehd-5125991/>
senior maverick, theoretical programmer,
Fragile-Free Software Institute

also known as

macro maven on SAS-L

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.

References

- definition: regression testing* (n.d.). URL: https://en.wikipedia.org/wiki/Regression_testing.
- DOS command: call* (n.d.). URL: <https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/call>.
- DOS command: goto* (n.d.). URL: <https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/goto>.
- DOS commands: List* (n.d.). URL: https://en.wikipedia.org/wiki/List_of_DOS_commands.
- DOS time command: replace leading space with zero* (n.d.). URL: <https://serverfault.com/questions/147515/need-leading-zero-for-batch-script-using-time-variable>.
- DOS: environment variable* (n.d.). URL: https://en.wikipedia.org/wiki/Environment_variable.
- DOS: redirection* (n.d.). URL: <https://helpdeskgeek.com/how-to/redirect-output-from-command-line-to-text-file/>.
- Fehd, Ronald J. (2018). "A Sysparm Companion, Passing Values to a Program from the Command Line". In: *SouthEast SAS Users Group Conference Proceedings*. 8 pp.; shows use of sysparm as macro variable and option which can be assigned value on command line in batch programs; program parse-sysparm parses a list of comma-separated values (csv) of form var1=value1,var2=value2,...,varN=valueN into macro variables. URL: http://www.lexjansen.com/sesug/2018/SESUG2018_Paper-197_Final_PDF.pdf.
- (Sept. 2022). "An Autoexec Companion, Allocating Location Names during Startup". In: *Western Users of SAS Software Annual Conference Proceedings*. 17 pp.; document design, running text: title, footnote; environment and macro variables, file searches: library: librefs, programs: filerefs; format search: fmtsearch; functions: cexist, exist, ifc, getoption, %sysfunc; macro autocall: file search: mautosource, sasautos; macros, catalog of compiled and stored: mstored, sasstore; procedure fcmp data set of functions: cmplib. URL: <https://www.lexjansen.com/wuss/2022/WUSS-2022-Paper-65.pdf>.
- (May 2023). "A Configuration File Companion: testing and using environment variables and options; templates for startup-only options initstmt and termstmt". In: *Pharmaceutical SAS Users Group Conference Proceedings*. 18 pp.; URL: <https://www.pharmasug.org/proceedings/2023/AP/PharmaSUG-2023-AP-009.pdf>.
- Torres, Zeke (June 2020). "Customized Output with the SAS(R) Config File: Always Useful but Seldom Used". In: *SAS Global Forum Annual Conference Proceedings*. 14 pp.; .bat, .log, .lst, macro, naming conventions, operating system, sysparm. URL: <https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2020/5170-2020.pdf>.
-