Paper CC-07

Some Common and Not So Common Uses of Pipes in a Windows Environment

Brian Varney, COMSYS IT Partners, Inc., Kalamazoo, Michigan

ABSTRACT

Utilizing pipes from Windows programs opens up new possibilities and functionalities. This paper will introduce and show examples of some common and not so common uses of pipes in a Windows environment. Pipes can also be a very powerful tool in other operating systems but that is outside the scope of this paper.

INTRODUCTION

The Windows operating system on servers and clients holds information that can make programming tasks easier without having to shell out of SAS. Using pipes along with the powerful SAS programming language can result in new possibilities and reduced effort in getting the job done.

This paper is intended for those with previous experience in SAS/Base and the Data Step.

Disclaimer: DOS commands may differ across the different flavors of the Windows operating systems.

CONCEPTS

First we will review the concepts and types of problems addressed in this paper. I have gathered the help on the DOS commands available on my computer by submitting:

```
filename helpp pipe 'help';
data help;
   infile helpp truncover;
   input line $char200.;
   command=scan(line,1,' ');
   lencomm=length(command);
   description=left(substr(line,lencomm+1));
run;
```

The computer used for developing this paper has Windows XP Professional service pack 1. Other types of Windows platforms may produce slightly different results.

The table below displays the results from the data step above. This will give us an idea of the different types of commands which are possible to use via pipes. This paper will not address all of these commands but reviewing these commands may trigger ideas of how one can utilize pipes in ways not covered in this paper.

DOS Command	Description
ASSOC	Displays or modifies file extension associations.
AT	Schedules commands and programs to run on a computer.
ATTRIB	Displays or changes file attributes.
BREAK	Sets or clears extended CTRL+C checking.
CACLS	Displays or modifies access control lists (ACLs) of files.
CALL	Calls one batch program from another.
CD	Displays the name of or changes the current directory.
CHCP	Displays or sets the active code page number.
CHDIR	Displays the name of or changes the current directory.
CHKDSK	Checks a disk and displays a status report.
CHKNTFS	Displays or modifies the checking of disk at boot time.
CLS	Clears the screen.
CMD	Starts a new instance of the Windows command interpreter.

DOS Command	Description
COLOR	Sets the default console foreground and background colors.
COMP	Compares the contents of two files or sets of files.
COMPACT	Displays or alters the compression of files on NTFS partitions.
CONVERT	Converts FAT volumes to NTFS. You cannot convert the current drive.
COPY	Copies one or more files to another location.
DATE	Displays or sets the date.
DEL	Deletes one or more files.
DIR	Displays a list of files and subdirectories in a directory.
DISKCOMP	Compares the contents of two floppy disks.
DISKCOPY	Copies the contents of one floppy disk to another.
DOSKEY	Edits command lines, recalls Windows commands, and creates macros.
ECHO	Displays messages, or turns command echoing on or off.
ENDLOCAL	Ends localization of environment changes in a batch file.
ERASE	Deletes one or more files.
EXIT	Quits the CMD.EXE program (command interpreter).
FC	Compares two files or sets of files, and displays the differences between them.
FIND	Searches for a text string in a file or files.
FINDSTR	Searches for strings in files.
FOR	Runs a specified command for each file in a set of files.
FORMAT	Formats a disk for use with Windows.
FTYPE	Displays or modifies file types used in file extension associations.
GOTO	Directs the Windows command interpreter to a labeled line in a batch program.
GRAFTABL	Enables Windows to display an extended character set in graphics mode.
HELP	Provides Help information for Windows commands.
IF	Performs conditional processing in batch programs.
LABEL	Creates, changes, or deletes the volume label of a disk.
MD	Creates a directory.
MKDIR	Creates a directory.
MODE	Configures a system device.
MORE	Displays output one screen at a time.
MOVE	Moves one or more files from one directory to another directory.
PATH	Displays or sets a search path for executable files.
PAUSE	Suspends processing of a batch file and displays a message.
POPD	Restores the previous value of the current directory saved by PUSHD.
PRINT	Prints a text file.
PROMPT	Changes the Windows command prompt.
PUSHD	Saves the current directory then changes it.
RD	Removes a directory.
RECOVER	Recovers readable information from a bad or defective disk.
REM	Records comments (remarks) in batch files or CONFIG.SYS.

DOS Command	Description
REN	Renames a file or files.
RENAME	Renames a file or files.
REPLACE	Replaces files.
RMDIR	Removes a directory.
SET	Displays, sets, or removes Windows environment variables.
SETLOCAL	Begins localization of environment changes in a batch file.
SHIFT	Shifts the position of replaceable parameters in batch files.
SORT	Sorts input.
START	Starts a separate window to run a specified program or command.
SUBST	Associates a path with a drive letter.
TIME	Displays or sets the system time.
TITLE	Sets the window title for a CMD.EXE session.
TREE	Graphically displays the directory structure of a drive or path.
TYPE	Displays the contents of a text file.
VER	Displays the Windows version.
VERIFY	Tells Windows whether to verify that your files are written correctly to a disk.
VOL	Displays a disk volume label and serial number.
XCOPY	Copies files and directory trees.

Now we can focus on utilizing a few of the commands from above in programming problems.

1. TREE: READING DIRECTORY TREE STRUCTURE

TREE Command Documentation
Graphically displays the folder structure of a drive or path. TREE [drive:][path] [/F] [/A] /F Display the names of the files in each folder. /A Use ASCII instead of extended characters.

```
filename pipetree pipe 'tree "c:\"  /F /A' lrecl=5000;

data a;
   infile pipetree truncover;
   input dirlist $char1000.;
run;
```

Partial Listing from data set above

dirlist
Folder PATH listing
Volume serial number is 71FAE346 0CB8:2B26
C:\
| aegen.rtf
| agestats.sas7bdat

dirlist
| Andrew.pdf
| atlog.txt
| AUTOEXEC.BAT
| BrazilGameatMoralez.jpg
| briansclassdata.sas7bdat

2. SET: Reading WINDOWS ENVIRONMENT VARIABLES

Issuing the Windows SET command gives one access to the current Windows environment variables. These could be read into macro variables and used in programming tasks.

SET Command Documentation
Displays, sets, or removes cmd.exe environment variables.
SET [variable=[string]]
variable Specifies the environment-variable name.
string Specifies a series of characters to assign to the variable.
Type SET without parameters to display the current environment variables.

```
filename pipeset pipe "set";

data pipeset;
   infile pipeset;
   input line $char200.;
run;

ods rtf;
proc sql flow=30;
   select line
   from pipeset;
quit;
ods rtf close;
```

Output from PROC SQL above

line
APPDATA=C:\Documents and Settings\bvarney\Application Data
CommonProgramFiles=C:\Program Files\Common Files
ComSpec=C:\WINDOWS\system32\cmd.exe
FT15F001=FT15F001.DAT
HOMEPATH=\
INSTALL=( )
MYSASFILES=?CSIDL_PERSONAL\My SAS Files\9.1
OS=Windows_NT
Path=C:\Program Files\SAS\SAS 9.1;C:\Oracle\product\10.1.0\Client_2\bin;C:\Oracle\product\10.1.0\Client_2\jre\1.4.2\bin\client;C:\Oracle\product\10.1.0\Client_2\jre\1.4.2\bin;C:\Program Files\Compaq\C
PROCESSOR_ARCHITECTURE=x86
PROCESSOR_LEVEL=6
ProgramFiles=C:\Program Files
SAMPSIO=("!sasroot\core\sample" "!sasext0\dmine\sample" "!sasext0\dquality\sample" "!sasext0\access\sample" "!sase

line
SAMPSRC=("!sasroot\core\sample" "!sasext0\dmine\sample" "!sasext0\dquality\sample" "!sasext0\access\sample" "!sase
SASAUTOS=("!sasroot\core\sasmacro" "!sasext0\dmine\sasmacro" "!sasext0\dquality\sasmacro" "!sasext0\cpe\sasmacro"
sasext0=C:\Program Files\SAS\SAS 9.1
sasroot=C:\Program Files\SAS\SAS 9.1
SAS_EXECFILENAME=
SESSIONNAME=Console
SystemRoot=C:\WINDOWS
TKPATH=C:\Program Files\SAS\SAS 9.1;C:\Program Files\SAS\SAS 9.1\core\sasext
USERDNSDOMAIN=MSVCS.INT
USERNAME=bvarney
WIN32DMIPATH=C:\Program Files\Compaq\Compaq Management Agents\Dmi\Win32

3. DIR: RETRIEVING INFORMATION ABOUT WINDOWS DIRECTORIES AND FILES

DIR Command Documentation
Displays a list of files and subdirectories in a directory. DIR [drive:][path][filename] [/A[[:]attributes]] [/B] [/C] [/D] [/L] [/N] [/O[[:]sortorder]] [/P] [/Q] [/S] [/T[[:]timefield]] [/W] [/X] [/4] [drive:][path][filename] Specifies drive, directory, and/or files to list. /A Displays files with specified attributes. attributes D Directories R Read-only files H Hidden files A Files ready for archiving S System files - Prefix meaning not /B Uses bare format (no heading information or summary).

```
filename pipedir pipe ' dir "c:\" /S' lrecl=5000;

data b;
   infile pipedir truncover;
   input line $char1000.;
   length directory $1000;
   retain directory;
   if line =' ' or
      index(upcase(line),'<DIR>')   or
      left(upcase(line))=:'VOLUME'  then
     delete;
   if left(upcase(line))=:'DIRECTORY OF' then
   directory=left(substr(line,index(upcase(line),'DIRECTORY OF')+12));
   if left(upcase(line))=:'DIRECTORY OF' then
      delete;
   if input(substr(line,1,10),?? mmddyy10.) = . then
      substr(line,1,10)='12/31/2999';
   date=input(substr(line,1,10),?? mmddyy10.);
   format date mmddyy10.;
run;

proc sort data=b;
```

```
   by directory descending date;
run;

data Directory_Summary(drop=i line);
   set b;
   by directory;
   length filename $75;
   retain number_of_files_in_directory directory_size;
   if first.directory then
   do;
      number_of_files_in_directory=input(scan(line,2,' '),32.);
      directory_size=input(scan(line,4,' '),comma32.);
   end;
   file_size=input(scan(line,4,' '),comma32.);
   filename=' ';
   do i=5 to 100;
      filename=trim(left(filename))||' '||scan(line,i,' ');
      if scan(line,i,' ')=' ' then
         leave;
   end;
   if index(upcase(line),'FILE(S)') then
      delete;
   if date ge '30DEC2999'd then
      delete;
run;

proc sort data=directory_summary;
   by descending directory_size descending file_size;
run;
```

Partial Listing from directory_summary data set

directory	Filename	directory_size	file_size
c:\orion\ordetail	myorderfact.sas7bdat	160931840	53773312
c:\orion\ordetail	order_item.sas7bdat	160931840	46408704
c:\orion\ordetail	orders.sas7bdat	160931840	24318976
c:\orion\ordetail	customer.sas7bdat	160931840	16761856
c:\orion\ordetail	street_code.sas7bdat	160931840	11625472
c:\orion\ordetail	postal_code.sas7bdat	160931840	2294784
c:\orion\ordetail	postal_code.sas7bndx	160931840	1192960
c:\orion\ordetail	price_list.sas7bdat	160931840	1020928
c:\orion\ordetail	city.sas7bdat	160931840	742400
c:\orion\ordetail	county.sas7bdat	160931840	553984

4. MKDIR: CREATING NEW WINDOWS DIRECTORIES

One can create windows directories using pipes without the DOS window popping up as with other methods

```
filename pipmkdir pipe "mkdir c:\newdir";

data b;
   infile pipmkdir;
run;
```

CONCLUSION

As one can imagine, there are many opportunities to apply these techniques in our day to day programming. When developing SAS code to be used on a Windows platform, opportunities exist to interact with and leverage information from the operating system.

REFERENCES

DOS Help Command

SAS Version 9 Online Help

ACKNOWLEDGMENTS

I would like to take this time to thank everyone I have worked with and learned from in my exciting SAS adventures.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. The author may be contacted at:

Brian Varney
COMSYS IT Partners, Inc., National SAS Practice
5278 Lovers Lane
Portage, Michigan 49002
Work Phone: (800) 831-6314
Fax: (269) 344-1887
Email: BVarney@COMSYS.com