

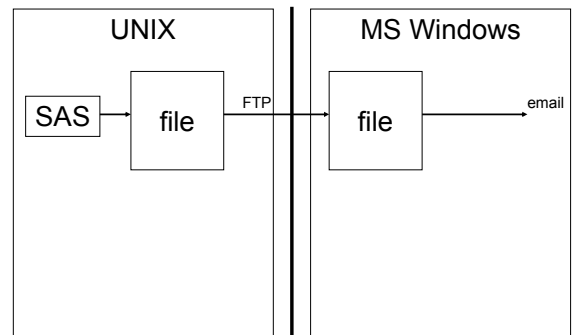
Compressing Output Files in a SAS® job, on a UNIX platform

Leslie J. Somos
Great Works Informatics, LLC

1

© 2008 by Great Works Informatics, LLC

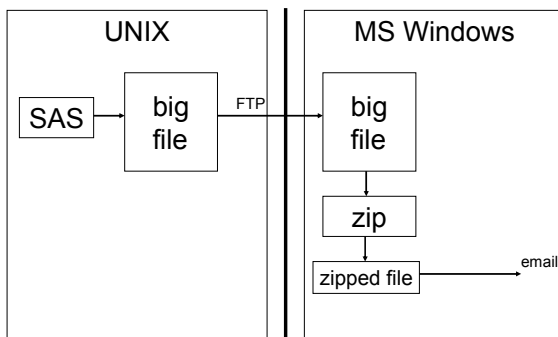
Original situation



2

© 2008 by Great Works Informatics, LLC

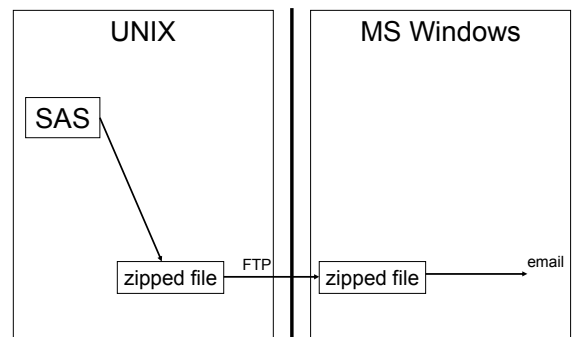
Original situation



3

© 2008 by Great Works Informatics, LLC

Goal



4

© 2008 by Great Works Informatics, LLC

Items we will touch on

- 'pipe' access method of FILENAME statement
- 'nobs=' and 'point=' options of SET statement
- compilation phase v. execution phase of DATA Step

5

© 2008 by Great Works Informatics, LLC

Coding Conventions

- SAS keywords in lowercase
- User-chosen words in uppercase
- Spaces supplied even where not required by the syntax
- Optional period after a macro variable always supplied

6

© 2008 by Great Works Informatics, LLC

Original code

```
filename OUT
'BIG.TXT'
lrecl=32760 ;
ordinary file

data _null_ ;
set OURDATA ;
file OUT ;
put <field>...<field> ;
run ;
```

7

© 2008 by Great Works Informatics, LLC

Modified code

```
filename OUT pipe
'compress > NOTSOBIG.Z'
lrecl=32760 ;
"unnamed pipe"
Program compress reads its STDIN, receives data written to filename OUT;
and writes to its STDOUT, which is redirected to NOTSOBIG.Z.

data _null_ ;
set OURDATA ;
file OUT ;
put <field>...<field> ;
run ;
```

8

© 2008 by Great Works Informatics, LLC

Original code :: Modified code

<pre>filename OUT 'BIG.TXT' lrecl=32760 ; ordinary file data _null_ ; set OURDATA ; file OUT ; put <field>...<field> ; run ;</pre>	<pre>filename OUT pipe 'compress > NOTSOBIG.Z' lrecl=32760 ; "unnamed pipe" data _null_ ; set OURDATA ; file OUT ; put <field>...<field> ; run ; No changes to data step, only to filename. When you uncompress, supply name BIG.TXT.</pre>
---	--

9

© 2008 by Great Works Informatics, LLC

Original code

```
filename OUT
'BIG.TXT'
lrecl=32760 ;

data _null_ ;
set ONEDATA ;
file OUT ;
put <field>...<field> ;
run ;

data _null_ ;
set TWODATA ;
file OUT mod ;
put <field>...<field> ;
run ;
```

10

© 2008 by Great Works Informatics, LLC

Original code :: Modified code

<pre>filename OUT 'BIG.TXT' lrecl=32760 ; data _null_ ; set ONEDATA ; file OUT ; put <field>...<field> ; run ; data _null_ ; set TWODATA ; file OUT mod ; put <field>...<field> ; run ;</pre>	<pre>filename OUT pipe 'compress > NOTSOBIG.Z' lrecl=32760 ; data _null_ ; set ONEDATA ; file OUT ; put <field>...<field> ; run ; data _null_ ; set TWODATA ; file OUT mod ; put <field>...<field> ; run ; No changes to data steps - problem! Program compress gets called a 2nd time, and overwrites NOTSOBIG.Z. Only TWODATA is in the resulting file.</pre>
---	---

11

© 2008 by Great Works Informatics, LLC

Original code :: Modified code

<pre>filename OUT 'BIG.TXT' lrecl=32760 ; data _null_ ; set ONEDATA ; file OUT ; put <field>...<field> ; run ; data _null_ ; set TWODATA ; file OUT mod ; put <field>...<field> ; run ;</pre>	<pre>filename OUT pipe 'compress > NOTSOBIG.Z' lrecl=32760 ; data _null_ ; file OUT ; set ONEDATA nobs=CONSTNOBS1 ; /* do POINTVAR = 1 to CONSTNOBS1 ; set ONEDATA point=POINTVAR ; put <field>...<field> ; end ; set TWODATA nobs=CONSTNOBS2 ; do POINTVAR = 1 to CONSTNOBS2 ; set TWODATA point=POINTVAR ; put <field>...<field> ; end ; stop ; /*<=REMEMBER!*/ run ; Only one data step, one call to compress. Problem if ONEDATA has zero observations. */</pre>
---	---

12

© 2008 by Great Works Informatics, LLC

Original code :: Modified code

```
filename OUT pipe
'compress > NOTSOBIG.Z'
lrecl=32760 ;

data _null_ ;
file OUT ;
if 0 then
set ONEDATA nobs=CONSTNOBS1 ;
do POINTVAR = 1 to CONSTNOBS1 ;
set ONEDATA point=POINTVAR ;
put <field>..<field> ;
end ;
if 0 then
set TWODATA nobs=CONSTNOBS2 ;
do POINTVAR = 1 to CONSTNOBS2 ;
set TWODATA point=POINTVAR ;
put <field>..<field> ;
end ;

stop ; /*<=REMEMBER!*/
run ;
```

Works fine, looks a little cluttered.

13

© 2008 by Great Works Informatics, LLC

Original code :: Modified code

```
filename OUT pipe
'compress > NOTSOBIG.Z'
lrecl=32760 ;

data _null_ ;
file OUT ;

do POINTVAR = 1 to CONSTNOBS1 ;
set ONEDATA point=POINTVAR
nobs=CONSTNOBS1 ;
put <field>..<field> ;
end ;

do POINTVAR = 1 to CONSTNOBS2 ;
set TWODATA point=POINTVAR
nobs=CONSTNOBS2 ;
put <field>..<field> ;
end ;

stop ; /*<=REMEMBER!*/
run ;
```

14

© 2008 by Great Works Informatics, LLC

DATA Step -- compilation phase v. execution phase

<http://v8doc.sas.com>
SAS OnlineDoc
.. Base SAS Software
.. SAS Language Reference: Concepts
... DATA Step Concepts
.... DATA Step Processing
..... Overview of DATA Step Processing

Flow of Action

When you submit a DATA step for execution, it is first compiled and then executed.

15

© 2008 by Great Works Informatics, LLC

SET

<http://v8doc.sas.com>
SAS OnlineDoc
.. Base SAS Software
.. SAS Language Reference: Dictionary
... Dictionary of Language Elements
.... Statements
..... SET

NOBS=variable

At compilation time, SAS reads the descriptor portion of each data set and assigns the value of the NOBS= variable automatically. Thus, you can refer to the NOBS= variable before the SET statement.

POINT=variable

POINT= causes the SET statement to use random (direct) access to read a SAS data set.

CAUTION:

Continuous loops can occur when you use the POINT= option.

When you use the POINT= option, you must include a STOP statement to stop DATA step processing, programming logic that checks for an invalid value of the POINT= variable, or both.

16

© 2008 by Great Works Informatics, LLC

Reading from and Writing to UNIX Commands (PIPE)

<http://v8doc.sas.com>
SAS OnlineDoc
.. Base SAS Software
... Host Specific Information
.... UNIX Environments (Companion)
..... Running the SAS System Under UNIX
..... Using External Files and Devices
..... Reading from and Writing to UNIX Commands (PIPE)

FILENAME fileref PIPE 'UNIX-command' <options>;

Under UNIX, you can use the FILENAME statement to assign filerefs not only to external files and I/O devices, but also to a pipe. Pipes enable your SAS application to receive input from any UNIX command that writes to standard output and to route output to any UNIX command that reads from standard input.

17

© 2008 by Great Works Informatics, LLC

Using Unnamed Pipes (MS Windows)

<http://v8doc.sas.com>
SAS OnlineDoc
.. Base SAS Software
... Host Specific Information
.... Microsoft Windows Environment (Companion)
..... Using SAS with Other Windows Applications
..... Using Unnamed and Named Pipes
..... Using Unnamed Pipes

FILENAME fileref PIPE 'program-name' option-list

Example:

```
filename DIR pipe 'dir /?' ;
data _null_ ;
infile DIR ;
input ;
put _infile_ ;
run ;
```

NOTE: The infile DIR is:
 Unnamed Pipe Access Device,
 PROCESS=dir /?, RECLEN=V, LRECL=256
Displays a list of files and subdirectories in a directory.
...
NOTE: 38 records were read from the infile DIR.
The minimum record length was 0.
The maximum record length was 75.
NOTE: DATA statement used (Total process time):
 real time 0.09 seconds
 cpu time 0.03 seconds

18

© 2008 by Great Works Informatics, LLC

Questions?

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.