

Migration of SAS® Data from Unix Server to Linux Server

Hengwei Liu, Daiichi Sankyo Inc.

ABSTRACT

When a company decided to migrate SAS data from Solaris Unix server to Red Hat Enterprise Linux (RHEL) server, one issue was how to deal with the formats catalog files. The formats catalog files are not platform independent. If they are copied from Unix server to Linux server, they will not be of any use in the Linux server.

On the other hand, SAS datasets can be used across different platforms. It was decided to convert the formats catalog files to SAS datasets in the Unix server, migrate those SAS datasets to Linux server, and convert them back to formats catalog files.

In this paper we discuss the data migration project. Unix command and SAS macros are used in the project.

INTRODUCTION

The company had all the clinical study data in a Solaris Unix server. The study folders have compound level, study level and time point level.

For example, in the path `/sasdata/abcmed/abcmd01/IA/data/raw`, `abcmed` is the compound level, `abcmd01` is the study level, and `IA` is the timepoint level.

The raw data folders contained both SAS datasets and formats catalog files.

When the team decided to migrate SAS data from the Solaris Unix server to an RHEL server, two approaches were considered.

One approach is to use PROC CPORT with the DATECOPY option to create a single transport file for each raw data folder. The DATECOPY option copies the SAS internal date and time at which the SAS file was created and the date and time at which it was last modified to the resulting transport file. These are the sample codes:

```
libname raw "/sasdata/abcmed/abcmd01/IA/data/raw";  
filename xptfile '/sasdata/abcmed/abcmd01/IA/data/raw/abcmd01.xpt';  
proc cport lib=raw file=xptfile datecopy;  
run;
```

The transport file is subsequently copied to the Linux server, and PROC CIMPORT is used to convert the transport file back to SAS datasets and catalog file in Linux. If we run PROC CONTENTS on the files in Linux, we can see the date and time the SAS files were created and last modified. But the timestamps of the files are different from those in the Unix server.

The other approach is to convert the formats catalog files to SAS datasets in the Unix server, migrate all the SAS datasets to Linux server, and convert the SAS datasets for the formats back to formats catalog files in the Linux server. As the SAS datasets are directly migrated into Linux, a programmer will get this message in the log when he works on the SAS datasets in Linux:

NOTE: Data file XXXX is in a format that is native to another host, or the file encoding does not match the session encoding. Cross Environment Data Access will be used, which might require additional CPU resources and might reduce performance.

With the second approach, the timestamps for SAS datasets in the Linux server would be the same as those in the Unix server. Only the timestamps for the catalog files would be changed.

The team decided to take the second approach.

THE NUTS AND BOLTS

Here is an outline of the process:

1. Use Unix command "find" to find all the formats catalog files in the Unix server.
2. Use PROC FORMAT to convert all the formats catalog files to SAS datasets in the Unix server.
3. Migrate the SAS datasets to the Linux server and use PROC FORMAT to convert them back to formats catalog files.

The consultant working on this project was given read and write permission to all the study data folders so that he could run Unix command and SAS programs on all the data folders.

STEP 1

In this step the Unix command "find" is used to find all the formats catalog files.

The command `find -name formats.sas7bcat` is run inside the top level folder `/sasdata`. The output 1 shows sample output.

```
./abcmed/abcmd01/IA/data/raw/formats.sas7bcat
./abcmed/abcmd01/csr/data/raw/formats.sas7bcat
./abcmed/abcmd02/IA/data/raw/formats.sas7bcat
./abcmed/abcmd02/csr/data/raw/formats.sas7bcat
./abcmed/abcpk01/csr/data/raw/formats.sas7bcat
./luk/lukmd01/dsur/data/raw/formats.sas7bcat
./luk/lukmd01/csr/data/raw/formats.sas7bcat
./luk/lukpk01/csr/data/raw/formats.sas7bcat
```

Output 1. Sample Output of the Unix Command `find -name formats.sas7bcat`

STEP 2

In this step PROC FORMAT is used to convert all the catalog files to SAS datasets in Solaris Unix server.

The output from step 1 is saved in a text file, say `dir.txt`. The SAS macro `cat2sas.sas` reads the `dir.txt` and parses the text in that file to get the path for each folder that contains a formats catalog file. PROC FORMAT is used to convert the catalog file to a SAS dataset.

```
%macro cat2sas(indir=);
*indir is the path for the text file that shows the folders containing
formats.sas7bcat;
*usage: %cat2sas(indir=/sasdata/dir.txt);

data dir;
    infile "&indir" print pad missover;
    input text $ 1-132;
run;
```

```

data dir; length myfile $15. mypath $120.;
set dir;
myfile=scan(text,-1,'/');
myind=index(text,myfile);
mypath=substr(text,1,myind-2);
mypath=tranwrd(mypath, '.', '/sasdata');
run;

data _null_;
set dir end=eof;
i+1;
call symput(compress("mypath"||put(i,best.)), trim(left(mypath)));
if eof then call symput('tot',trim(left(put(_n_,best.))));
run;

%do k=1 %to &tot;
libname lib&k "&&mypath&k";

proc format lib=lib&k cntlout=lib&k..formats;
run;

%end;

%mend;

```

STEP 3

After the SAS datasets for the format catalog files are copied to the Linux server, they are converted back to formats catalog files.

This step is done in the Linux server.

The SAS macro sas2cat.sas reads the dir.txt and parses the text in that file to get the path for each folder that contains a formats catalog file in the Solaris Unix server. PROC FORMAT is used to convert the SAS dataset formats.sas7bdat to catalog file. The SAS dataset formats.sas7bdat is deleted after the formats catalog file is created.

```

%macro sas2cat(indir=);
*indir is the path for the text file that shows the folders containing
formats.sas7bcat;
*usage: %sas2cat(indir=/sasdata/dir.txt);

data dir;
    infile "&indir" print pad missover;
    input text $ 1-132;
run;

data dir; length myfile $15. mypath $120.;
set dir;
myfile=scan(text,-1,'/');
myind=index(text,myfile);

```

```

mypath=substr(text,1,myind-2);
mypath=tranwrd(mypath, '.', '/sasdata');

data _null_;
set dir end=eof;
i+1;
call symput(compress("mypath"||put(i,best.)), trim(left(mypath)));
if eof then call symput('tot',trim(left(put(_n_,best.))));
run;

%do k=1 %to &tot;
libname lib&k "&&mypath&k";

proc format lib=lib&k cntlin=lib&k..formats;
run;

*remove the SAS datasets formats;
proc datasets memtype=data lib=lib&k;
delete formats;
run;

%end;
%mend;

```

CONCLUSION

The SAS catalog files are not platform independent. When there is a migration from an old server to a new server, the catalog files need special consideration. In this paper we have shown how to use Unix command to locate all the formats catalog files in the old server and use SAS macros to create new catalog files in the new server.

ACKNOWLEDGMENTS

The author thanks the management at Daiichi Sankyo, Inc. for support.

CONTACT INFORMATION

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

Hengwei Liu
Daiichi Sankyo, Inc.
211 Mount Airy Road
Basking Ridge, NJ 07920
Hengwei_liu@yahoo.com

Any brand and product names are trademarks of their respective companies.