The Interaction Between the SYNTAXCHECK and OBS= System Options During Batch Unix Sessions
Robert Adams, Merck & Co., Inc., North Wales, PA

ABSTRACT
A SAS program submitted to Unix in batch mode doesn't behave the same way as a program submitted to Unix interactively. One of the reasons is because of the SYNTAXCHECK and OBS= System Options. This paper will examine the interaction between these options and will show how they need to be set in order for programs to create the output intended. As a result, programmers new to batch processing on Unix can spend more time getting output and less time investigating the inner workings of the SAS system.

INTRODUCTION
SAS programs developed to manipulate large data sets often can take 3-4 hours to run. In this case, it is desirable to use Unix to submit the program via batch, allowing it to run in the background. Non-interactive, batch Unix programs make use of the SYNTAXCHECK System Option. This option, along with the OBS= Option it impacts, need to be understood in order for the program to produce output.

EXAMPLE
Below is a SAS/ACCESS statement that queries an Oracle table and creates a permanent data set. Let's examine what happens when we run this block of code.

```sas
proc sql;

    CONNECT TO ORACLE(user=MYUNAME orapw=XXXXXXXX path="@ABCD.test");

    ** lab result info **;

    create table datadir.plabres2 as
    select * from connection to oracle ( 
        select *
        from TEST01.LAB_RSLT
        where PRTTN_YR >= '2007'
    );

    NOTE: Table DATADIR.PLABRES2 created, with 0 rows and 15 columns.

    disconnect from oracle;
quit;

NOTE: SAS set option OBS=0 and will continue to check statements.
This may cause NOTE: No observations in data set.
```

You will notice that our table gets created, but it doesn't contain any observations. You will also notice that there is a note saying that SAS set the OBS System Option to 0. That explains why there are no observations in my data set, but why did it happen?
EXPLANATION

There is a SAS System Option called SYNTAXCHECK | NOSYNTAXCHECK that enables syntax checking mode for statements that are submitted within a batch Unix session. The default setting is SYNTAXCHECK.

Syntax checking mode causes the DATA and PROC steps in the program to be validated, something important for any SAS program. It also causes SAS to stop processing when it encounters an error, something especially important for long running batch sessions. The only problem is that when SAS enters syntax checking mode, it sets the OBS System Option to 0, and no observations are output.

When this happens, SAS acts as follows:

- reads the remaining statements in the DATA step or PROC step
- checks that statements are valid SAS statements
- executes global statements
- writes errors to the SAS log
- creates the descriptor portion of any output data sets that are specified in program statements
- does not write any observations to new data sets that SAS creates
- does not execute most of the subsequent DATA steps or procedures in the program (exceptions include PROC DATASETS and PROC CONTENTS)

The dilemma is that you want syntax checking mode enabled so that SAS stops processing when it encounters an error, but you don't want it to set the OBS System Option to 0 because you need your data set to be output.

SOLUTION

Looking back at our example program, you will notice that there were no errors. Since there were no errors, SAS should have continued selecting and processing observations for the data set. However, the SYNTAXCHECK option set OBS= 0, so no observations were processed.

One of the things that SAS does after it sets the option OBS= 0 is to execute global statements. That means we can explicitly specify a SAS options statement to set OBS = MAX after the option is initially set to 0. The statement would look like the following:

```
proc sql;
    connect to oracle(user=MYUNAME orapw=XXXXXXXX path="@ABCD.test");
    options obs = max;
    ** lab result info **;
    create table datadir.plabres2 as
        select * from connection to oracle ( 
            select *
            from test01.lab_rslt
            where prttn_yr >= '2007'
        );
    disconnect from oracle;
quit;
```
The key is to understand what is happening behind the scenes when you submit your SAS programs in batch mode. The order of operations for the updated code above would be as follows:

- SAS enters syntax checking mode, since that is the default and what we want
- SAS sets OBS=0
- global statements are executed and the OBS System Option is set to MAX
- the program creates the desired output, a new permanent dataset

When this happens, you will see the following in your log:

```sas
proc sql ;
   CONNECT TO ORACLE(user=MYUNAME orapw=XXXXXXXX path="@ABCD.test") ;
   options obs = max ;
   ** lab result info **;
   create table datadir.plabres2 as
   select * from connection to oracle ( select * from TEST01.LAB_RSLT where PRTTN_YR > '2007' );
   NOTE: Table DATADIR.PLABRES2 created, with 7222749 rows and 15 columns.
   disconnect from oracle;
   quit;
   NOTE: SAS set option OBS=0 and will continue to check statements. This may cause 
   NOTE: No observations in data set.
```

SAS doesn't show a note stating that it set option OBS= MAX, but we can see that it did since our permanent data set was created with 7,222,749 observations.

**APPLICATION**

Since each batch SAS program on Unix is treated like a new session, you will need to include an OBS= MAX statement at the beginning of each program to allow SAS to both check the syntax of your program as well as to output observations to your data sets.

**CONCLUSION**

The goal of this paper was to demonstrate the need for global options statements during batch SAS sessions on Unix. These statements are essential in order to compensate for changes that SAS makes to session settings during default batch processing.

**REFERENCES**

SAS OnlineDoc 9.1.3: http://support.sas.com/onlinedoc/913/docMainpage.jsp
CONTACT INFORMATION

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

   Robert Adams
   Merck and Company, Inc.
   UG1D-10
   351 North Sumneytown Pike
   North Wales, PA 19454
   267-305-8249
   robert_adams2@merck.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 trademarks of their respective companies.