



I How can I change the current SQL id associated with a DB2® process using SAS/ACCESS® interface to DB2?

O Changing the current SQL id is supported by IBM®'s DB2 product through the use of a SET CURRENT SQLID= statement. When used, this statement is valid as long as the current connection or thread to DB2 is maintained. It can be executed using the SQL procedure's Pass-Through facility, beginning with Release 6.07 of the SAS System.

In a SAS batch job, the SET CURRENT SQLID= statement can be used within an execution of the SQL procedure. Any subsequent DB2 DBMS command in the same SQL task uses the new SQL id supplied as the task's current id. The following code illustrates this for batch programs:

```
/* Associate libref with SAS data library. This */
/* library contains four view descriptors, mapping to */
/* four DB2 tables. */
libname viewlib 'sas.data.library';

proc sql;
  connect to db2(ssid=xxxx);
  execute (SET CURRENT SQLID='authid') by db2;
  select * from viewlib.member1;
  insert into viewlib.member2
    values('Smith', 'Jeff', '1993-03-10');
  update viewlib.member3
    set cost=cost * 1.10
    where product='Wiggitt';
  delete viewlib.member4
    where product='Oldstock';
  disconnect from db2;
  other sql queries
quit;
```

Once the PROC SQL QUIT or DISCONNECT statement is executed, the connection to DB2 is terminated, and therefore, the SET CURRENT SQLID= statement is no longer active.

For interactive users, the above application also runs fine. If

your site has SAS/AF software, you will be able to maintain a connection to DB2 throughout your application by using a SUBMIT CONTINUE SQL block in the INIT section of the initial program that connects to DB2. Upon termination of your application, the TERM section executes to disconnect from DB2. The following code illustrates this:

```
INIT:
  submit continue sql;
    connect to db2(ssid=xxxx);
    execute (SET CURRENT SQLID='authid') by db2;
  endsubmit;
return;

MAIN:
  submit continue;
    reporting code
  endsubmit;
return;

TERM:
  submit continue sql;
    disconnect from db2;
  endsubmit;
return;
```

As long as the SAS/AF program has not been exited, the SET CURRENT SQLID= statement remains in effect. The only drawback to this technique is that a DB2 commit point does not occur until the SAS/AF program is terminated. This will be of concern to sites where concurrent update access is needed.

One final note on using the SET CURRENT SQLID= statement is that your DB2 subsystem must be installed with the appropriate DB2 authorization exit to support it. Consult with your database administrator to see if you can issue SET CURRENT SQLID= before you try it in a SAS application.

Answer provided by Mark Lochbihler. Mark is a marketing analyst at SAS Institute where he specializes in the SAS/ACCESS products. Copyright © 1993 by SAS Institute Inc.