The LIBNAME Engine Compared to SQL Pass-Through

Steven Feder, Federal Reserve Board of Governors, Washington, DC

ABSTRACT

Both the LIBNAME engine and SQL pass-through provide access to databases. SQL pass-through passes code directly to the database. Without pass-through, SAS generates an optimized database query. LIBNAME coding is simple and familiar, and the efficiency is optimized by SAS®. Is there any reason for most users ever to use SQL pass-through? This paper will compare code and results for DB2® using SAS ON OS/390, and ODBC drivers to the same DB2 server on Windows SAS.

WINDOWS

The pass-through code uses a CONNECT statement to specify the database instead of the LIBNAME statement. Both specify the ODBC driver which points to the DB2 server.

PASS-THROUGH

```
proc sql;
  connect to odbc(dsn=M1DB2P user=xxxxx password=xxxxx);
  create table out.fcexrcri3 as
  select * from connection to odbc
    (select fcex.*,rcri.dt
    from fdrp.cuv_fcex01 as fcex
    inner join fdrp.cuv_rcri01 as rcri
    on fcex.id_rssd=rcri.id_rssd
    where rcri.dt=19591231);
  disconnect from odbc;
quit;
```

LIBNAME

```
libname in odbc complete="dsn=M1DB2P;Uid=xxxxx;pwd=xxxxx"
        schema=fdrp;

proc sql;
  create table out.fcexrcri1 as
  select fcex.*,rcri.dt
  from in.cuv_fcex01 as fcex
  inner join in.cuv_rcri01 as rcri
  on fcex.id_rssd=rcri.id_rssd
  where rcri.dt=19591231;
quit;
```

DATA STEP

```
data out.fcexrcri2;
  merge in.cuv_fcex01(in=infcex in=infcex)
        in.cuv_rcri01(in=inrcri keep=id_rssd dt in=inrcri
    where=(dt=19591231));
  by id_rssd;
  if infcex and inrcri;
```

```
run;
```

COMPARISON

The pass-through and LIBNAME code produce very similar mean run-times. The DATA STEP code uses significantly more user cpu, though slightly less system cpu. (See Table 1).

TABLE 1

PASS-THROUGH	1	2	3	Mean
real time	01:10.7	01:00.0	02:24.2	01:31.6
user cpu time	3.79	3.81	3.92	3.84
system cpu time	0.82	0.76	0.54	0.71
SQL				
real time	01:26.1	01:13.7	01:41.9	01:27.2
user cpu time	3.95	3.95	3.59	3.83
system cpu time	0.51	0.68	0.7	0.63
DATA STEP				
real time	02:58.4	01:13.1	02:05.0	02:05.5
user cpu time	4.85	5.03	4.76	4.88
system cpu time	0.59	0.53	0.42	0.51

The data step code, while less efficient, may be preferable when extensively manipulating the data.

OS/390

The coding is very similar, though there are some small differences in the CONNECT and LIBNAME. Only the differing statements are listed here:

PASS-THROUGH

```
CONNECT TO DB2(SSID=DSN);
```

LIBNAME

```
LIBNAME IN DB2 SSID=DSN AUTHID=FDRP;
```

COMPARISON

The results are similar to those for WINDOWS. PASS-THROUGH and LIBNAME are about the same. DATA STEP is worse, but not so bad as to be prohibitive. (See table 2.)

TABLE 2

	1	2	3	
PASS-THROUGH				
CPU time	01:00.6	00:59.9	00:57.7	00:59.4
Elapsed time	03:21.0	01:49.6	01:28.0	02:12.9
SQL				
CPU time	00:58.5	00:57.7	00:56.8	00:57.7
Elapsed time	01:35.2	01:48.9	01:20.3	01:34.8
DATA STEP				
CPU time	01:13.9	01:12.9	01:14.1	01:13.6
Elapsed time	01:58.2	01:49.4	01:42.0	01:49.8

IMBEDDED SELECT

Queries are commonly imbedded within other queries. The code as written by the user is again similar for both pass-through and LIBNAME:

```
proc sql;
  connect to odbc(dsn=M1DB2P user=m1shf00 password=lol81972);
  create table out.fcexrcri2 as
  select * from connection to odbc
    (select fcex.*
    from fdrp.cuv_fcex01 as fcex
    where id_rssd in(
      select id_rssd
      from fdrp.cuv_rcri01
      where dt=19591231));
  disconnect from odbc;
quit;

proc sql;
  create table out.fcexrcri1 as
  select fcex.*
  from in.cuv_fcex01 as fcex
  where id_rssd in(
    select id_rssd
    from in.cuv_rcri01
    where dt=19591231);
quit;
```

The results are similar as well. (See Table 3.) This suggests that for many uses pass-through and LIBNAME code will run with similar efficiency.

TABLE 3

	1	2	3	Mean
PASS-THROUGH				
real time	01:08.6	02:11.3	01:03.3	01:27.7
user cpu time	3.78	3.4	3.68	3.62
system cpu time	0.62	0.71	0.57	0.63

SQL real time	01:49.8	02:12.4	01:46.5	01:56.2
user cpu time	3.85	3.79	3.57	3.74
system cpu time	0.53	0.76	0.45	0.58

FUNCTIONS

Using the LIBNAME, SAS still passes certain functions to DB2 for processing. (See *SAS/ACCESS 9.1 Supplement for DB2 under z/OS*, p. 34 for a list of these.) So one would expect these functions to perform the same whether passed explicitly with PASS-THROUGH or implicitly with LIBNAME. For using these functions, all else being equal, LIBNAME will be preferable because the coding is slightly simpler. DB2 does offer, however, a range of other functions which are not available in SAS. (See chapter 3 of the *DB2 UDB SQL Reference for Cross-Platform Development Version 2* for a list of these.) For these, PASS-THROUGH must be used. For example, the VARCHAR function returns a character from a numeric variable:

```
select varchar(rcri.rcfd3230) as rcfd3230_v….
```

REFERENCES

International Business Machines Corporation. (2004), *DB2 UDB SQL Reference for Cross-Platform Development Version 2*.
<ftp://ftp.software.ibm.com/ps/products/db2/info/xplatsql/pdf/en_US/cpsqlrv2.pdf>

Fadden, Scott. 2004. “Improving Performance: Accessing DB2 Data with SAS 9.“ *SUGI Proceedings,* Montréal, Canada.
<http://www2.sas.com/proceedings/sugi29/107-29.pdf>

SAS Institute Inc. (2004), *SAS/ACCESS 9.1 Supplement for DB2 under z/OS.* Cary, NC: SAS Institute Inc.

ACKNOWLEDGMENTS

CONTACT INFORMATION

Steven Feder
Federal Reserve Board, Mail Stop 157
Washington, D.C. 20551
202-452-3144
email: steven.h.feder@frb.gov