Paper 1297-2014

ODBC Connection to a Database using Keywords and SAS® Macros

Jesper Michelsen, Group Risk Management, Nykredit A/S

ABSTRACT

This ePoster paper will show the audience how to connect to a database without any registration of the ODBC connection in neither the Windows ODBC Administrator tool nor in the Windows Registry database. Furthermore, it is shown how the connection can be more flexible and better managed by building it into a SAS® macro.

INTRODUCTION

ODBC is an API which connects your application (e.g. SAS®) with your DBMS through a database provider (a database client).

It is smart because you do not have to recode your application or programs if you change your DBMS. You just change the ODBC connection.

Usually an ODBC connection is set up by making a System-DSN or a User-DSN. These are registered in the Windows registration database and the task is completed by using the ODBC Data source administrator which is a tool in the Windows Administration group.

However the System-DSN and User-DSN connections

- is difficult to maintain when there are several users on different hardware in your organization. And with different needs!?
- is difficult to distribute because they are sort of embedded into Windows

Also there are security considerations - are you allowed to make these changes in the registry? Or do you need help from IT department - who probably are busy doing other things?

THE NEED OF FLEXIBILITY

At Nykredit we needed a flexibility which was not possible to achieve using the traditional approach to ODBC connections.

Especially, flexible maintenance together with easy and swift distribution was critical. Also we wanted to make the connecting string a little more generic. Not splitting the connection into several “sub-routines” or parts located in different drives and folders on the network were also taken into consideration.

However there are other possibilities than using the System- or User-DSN.

Please note! You still need a DBMS specific client or provider installed – but should be a part of your basic system and handled by your IT department.

THE FILE-DSN

The File-DSN is an external registration of the connection-keywords in a file and values necessary to connect to your DBMS. It is

- Easy to distribute
- Relatively easy to maintain compared to the registrations in the Windows Registry Base
- Easy to implement in a LIBNAME statement – just use the DATASRC= option (and a little extra – you still need a password)

But it proved not to be flexible enough for our use, mainly because we had to split up the total connection string in some that were stable and some that was generic making it more complex than intended.

However, I will recommend that you start out with a File-DSN to see which connection keywords are necessary for your specific DBMS. The keywords are stored in an external file of the type .dsn. Just open the file using e.g. notepad.exe and examine it.

A SCRIPTED CONNECTION

A scripted connection to your DBMS is a little trickier to implement in a SAS® LIBNAME Statement. You have to use the NOPROMPT= option in connection to the ODBC engine. But it is worth the effort.

It is easy to distribute and it is flexible and it can be designed to be rather generic especially if you use the SAS® macro-language.

EXAMPLE

The following example shows a flexible macro (some of the variable values might be in Danish)

```
   /* librefCOUPROD.sas ************************************************/
   /* jmic@nykredit.dk, 10. Jan. 2014                                  */

1  %macro librefCOUPROD(_srv);
2  %local _APP _srv;
3  %global _rcCOUPROD _RC cancel noexec;

4  %if %quote(%str(&_srv)) =   %then
5      %let _srv=&_sqlsrv;

   /* If the OS is WINXP */
6  %if &sysscpl = WIN_PRO %then
7      %let _APP=Microsoft Data Access Components;
8  %else
9  %if &sysscpl=W32_VSPRO %then
10     %let _APP=Microsoft® Windows® Operativsystem;

11 %readPWkrypt(MART01P1);

12  libname COUPROD ODBC
13     noprompt="DRIVER=SQL Server;UID=&sysuserid;PWD=&_PASSWORD.;DATABASE=MART01P1;
14                  WSID=KONTOR-SASVM;APP=&_APP.;SERVER=%upcase(&_srv.)"
15                      schema=COUPROD;

    /* Checking status */
17  %let _RCCOUPROD=%sysfunc(libref(COUPROD));
18  %let _RC=&_RCCOUPROD;
19  %put NOTE: SQL-Server %upcase(&_srv);
20  %put NOTE: _RCCOUPROD (_RC) = &_RCCOUPROD (&_RC);

21  %mend librefCOUPROD;
```

In this short example only have to specify which server you want to connect to. This can of course be made more complex if you need to do so. E.g. the UID keyword can be something different than sysuserid, the DATABASE keyword could be the result of which server you connect to and so on.

SOME REMARKS ON THE MACRO EXAMPLE

The test in line 4 and 5 requires that a global macrovariable called _sqlsrv is at hand and has a value.

The readPwkrypt macro referred to in *line 11* is an in-house developed macro that reads an encrypted password. The password is encrypted by another in-house developed macro called PWkrypt which stores the encrypted user password in SASUSER. This has enabled us to make a sort of “silent” login to servers and data stores without having to write the passwords in clear text in the sourcecode. It also makes it easy for the users to administer their passwords for the many data sources we have.

Note that the NOPROMPT options starting in *line 13* are surrounded by double quotes. This is to secure that macro-variables will be resolved by the macro compiler. Be aware that some of the keyword values might be case-sensitive.

CONCLUSION

A scripted ODBC connection between SAS® and your DBMS

- used with the SAS® macro language is generic and flexible
- common code can be used by many users and for many databases and even for many DBMS
- frees up scare resources at the IT dept.

REFERENCES - AND RECOMMENDED READING

SAS/ACCESS® 9.3 Interface to ODBC, 2nd edition (SAS Institute Inc.)
http://support.sas.com/documentation/cdl/en/acreldb/65247/HTML/default/viewer.htm#p1g72kbb0m01y1n1gm1lh532n5ru.htm

SAS/ACCESS® 9.3 Interface to PC Files: Reference (SAS Institute Inc.), chapter on Libname Access and Excel Engines on Microsoft Windows.
http://support.sas.com/documentation/cdl/en/acpcref/63181/HTML/default/viewer.htm#titlepage.htm

Using Connection String Keywords with SQL Server Native Client, SQL Server 2012 (Microsoft Technet)
http://technet.microsoft.com/en-us/library/ms130822.aspx

CONTACT INFORMATION

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

Seniorspecialist
Jesper Michelsen
Nykredit a/s, Group Risk Management
Kalvebod Brygge 47
1560 Copenhagen V
Denmark

Work Phone: +45 4455 1362
Email: jmic@nykredit.dk