

**First Impressions of SAS/Access®
Interface To ODBC, For Windows**
Jim Anderson
Blue Cross Blue Shield of Nebraska

ODBC is a Microsoft standard for database access through a common interface. The SAS/ACCESS Interface to ODBC, for Windows enables the use of the SQL Pass-Through Facility with ODBC compliant products. In addition, to making observations that are specific to the ODBC product, limited comments will be offered concerning alternative methods for exchanging SAS® and non-SAS data within the Windows™ environment.

Introduction

Open Database Connectivity (ODBC) is an interface that allows, through the use of SQL, an application to access multiple ODBC compliant data sources. ODBC consists of a Driver Manager, Drivers, and application interfaces. All three of the components must be installed for the interface to work. The user needs to recognize that ODBC is an emerging, immature standard.

SQL

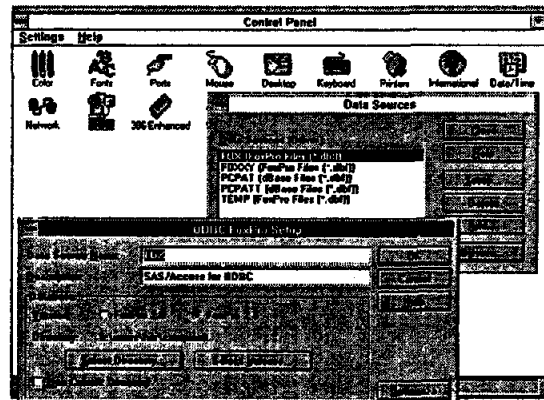
SQL is a widely accepted standard for data retrieval that originated from relational database concepts. Data storage locations are defined as tables, rows, columns, keys, and indexes. A full discussion of Structured Query Language is beyond the scope of this paper.

ODBC Driver Manager

ODBC's Driver Manager is a Microsoft® provided component. Accessible through the Control

Panel of Microsoft Windows, the Driver Manager's primary function is to load drivers. Using an ODBC.INI file, the Driver Manager maps a data source to a DLL for a specific driver. Initialization and ODBC call validation are additional Driver Manager functions.

Windows Control Panel



ODBC Driver

A driver for ODBC is supplied by software vendors. Each driver is a DLL that can connect with and submit requests to a data source, translate data formats, and return results or error codes to the application. In order to claim API compliance, a driver must support a core set of functions as defined by the X/Open and SQL Access Group's (SAG) Call Level Interface (CLI). Two levels of API extended functionality are also defined. Drivers are encouraged to support as much functionality as they can, however, compliance at a given level may be claimed only if all functions for that level are supported.

SAS/ACCESS Interface to ODBC

An interface for the application, or client, is used to request

data source connection, issue SQL request to the source, report query results or errors, and terminate data source connection. The SAS interface utilizes the SQL Procedure's Pass-Through facility. The interface from SAS is designed for the lowest level of ODBC compliance. However, if the driver support a higher level of API compliance, the SAS interface may extend into some advanced features and special queries.

Pass-Through

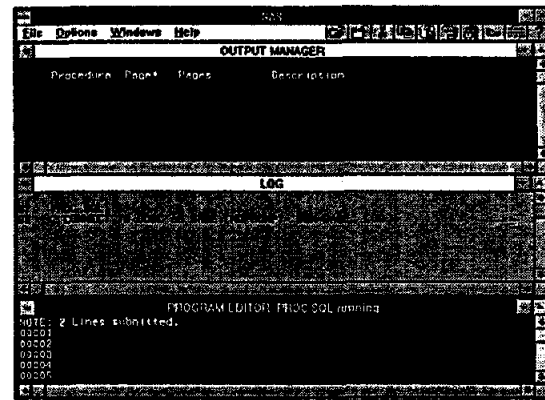
Pass-Through is a facility of PROC SQL. With Pass-Through, SAS variable names and formats can be assigned to DBMS columns and data types. Of particular value for table joins, queries are optimized for data sources that have an optimizer and when there are indexes on source columns. Optimization also occurs for queries that summarize, GROUP BY, or include expressions that create columns.

Pass-Through -- Connect

CONNECT is a Pass-Through statement used to establish a DBMS connection. A SAS session remains connected until termination of the SQL procedure or until a DISCONNECT is issued. Multiple, simultaneous ODBC connections are supported, however, an alias for each connection must then be used. By storing CONNECT statement arguments, PROC SQL views allow for automatic, direct and transparent ODBC connections to source data.

```
PROC SQL;
  CONNECT TO ODBC
    AS BASE (NOPROMPT='DSN=BASE' LOG);
```

SAS for Windows 6.08
CONNECT



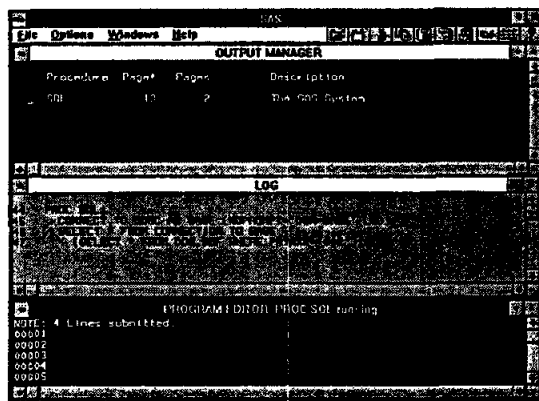
Pass-Through -- CONNECTION TO

Usable wherever a PROC SQL SELECT statement is valid, the CONNECTION TO is a Pass-Through facility component. CONNECTION TO enables DBMS queries. With multiple connections supported for one PROC SQL step, it is possible to access, merge, and otherwise act upon data in a variety of formats in the same SAS session.

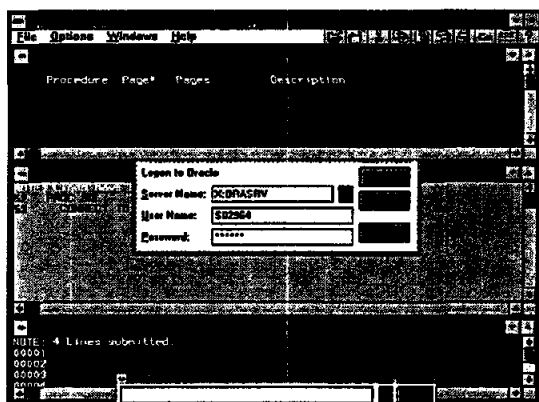
```
PROC SQL;
  CONNECT TO ODBC
    AS BASE (NOPROMPT='DSN=BASE' LOG);
  SELECT * FROM CONNECTION TO BASE
    (SELECT * FROM DDS.DBF
      WHERE PRTAXX='002');
```

DBMS column names are used as default labels in PROC SQL outputs. However, SAS uses the first eight non-blank characters of column names and append numbers as needed into the rightmost used characters to cause returned data to be named uniquely. Leading numerics or characters in the DBMS column names that are not valid for the SAS system are converted to underscores.

SAS for Windows 6.08
CONNECTION TO -- dBASE



SAS for Windows 6.08
CONNECTION TO -- ORACLE



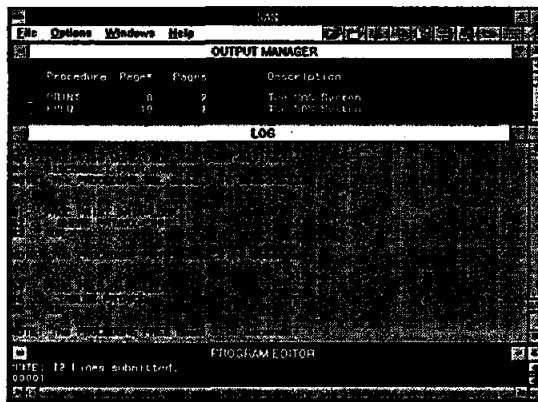
Pass-Through DISCONNECT

PROC SQL accepts concurrent connections to multiple sources. An ODBC connection may be terminated with the DISCONNECT statement without ending an executing PROC SQL. Execution of the SQL Procedure continues until a Quit, PROC, or DATA statement is submitted.

```
PROC SQL;
  CONNECT TO ODBC
    AS BASE (NOPROMPT='DSN=BASE' LOG);
  CREATE VIEW LOOK
    AS SELECT * FROM CONNECTION TO BASE
      (SELECT * FROM DDS.DBF);
  DISCONNECT FROM BASE;
PROC PRINT DATA=LOOK;
  WHERE PRTAXX='002';
RUN;
```

```
PROC FREQ DATA=LOOK;
  TABLES PRTAXX;
RUN;
```

SAS for Windows 6.08
DISCONNECT -- VIEW



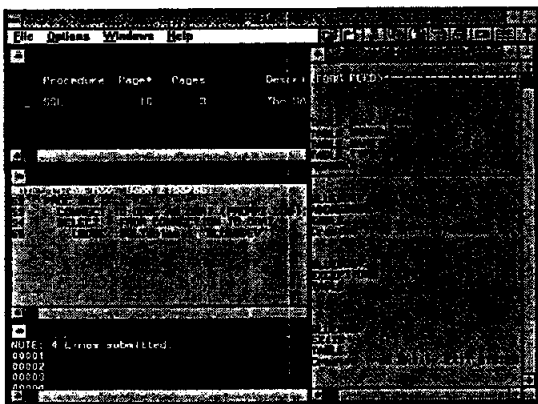
Pass-Through EXECUTE

Statements that are specific to a DBMS may be sent to the DBMS through use of the EXECUTE statement. EXECUTE allows for execution of dynamic, non-query statements. The SQL statements must be supported by the DBMS.

Special Queries

ODBC supports and SAS allows for a set of DBMS catalog functions.

```
SQLColumnPrivileges
SQLColumns
SQLForeignKeys
SQLPrimaryKeys
SQLProcedureColumns
SQLProcedures
SQLSpecialColumns
SQLStatistics
SQLTablePrivileges
SQLTables
```



Special queries are not supported or are selectively supported by each DBMS. The commands include arguments and are case sensitive.

ODBC Tomorrow

While offering exciting functionality, ODBC is an immature standard. Drivers act at different levels of conformance and the computing press has carried reports of poor performance. Microsoft has announced that it is changing the database engine with the expectation of significantly improved query speeds. To provide an expanded interoperability with other SQL products, ODBC 3.0 will include full SAG CLI conformance at the lowest level of compliance.

SAS 6.10 will include an ODBC driver as a part of the Base product. The SAS system will accept requests from applications that comply to the ODBC specification. The Base SAS driver does not replace, nor is it a substitute for, the client interface of the SAS/ACCESS for ODBC product.

Alternatives

Depending upon the data source, the SAS System includes a number of options to SAS/ACCESS For ODBC. Assuming that the data source can be made into an ASCII file, Base SAS can access the data with INFILE. The major negative to this approach is that non-ASCII data is not directly available to the SAS system.

Support for Dynamic Data Exchange (DDE) provides a second Base SAS alternative. DDE dynamically enables a client Windows application to request data from a server Windows application. Many of the applications that support DDE are also ODBC compliant. However, use of DDE requires a good understanding of SAS, DOS, Windows, and the Server application. In the example below, DDE requires that the Microsoft EXCEL sheet be open and the SAS user know the row and columns dimensions.

```
*****
*** DYNAMIC DATA EXCHANGE ***
*****
*** POINT TO AN OPEN EXCEL SHEET ***
*****

FILENAME PRICE DDE
    'EXCEL|SHEET1.XLS!R9C1:R276C7';

*****
*** READ EXCEL -- TAB DELIMITED ***
*****

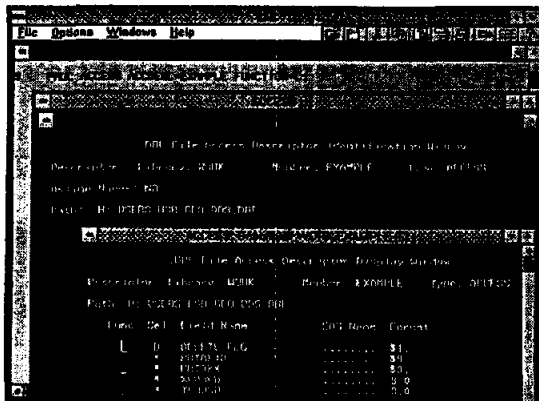
DATA LOOK;
    INFILE PRICE DLM='09'X NOTAB DSD MISSOEVER;
    INFORMAT VAR1 $4.
              FILL1 $1.
              VAR2 $5.
              FILL2 $1.
              VAR3 $9.
              FILL3 $1.
              VAR4 9.;
    INPUT VAR1 FILL1 VAR2 FILL2 VAR3 FILL3 VAR4;
    DROP FILL1 FILL2 FILL3;
RUN;
```

SAS/ACCESS® Interface to PC File Formats is a interface to DBF and DIF file formats. The product allows for a transparent

interface through descriptor files. The major negatives to this product are that it requires additional licensing and it applies to such a limited set for file formats.

SAS for Windows 6.08

SAS/ACCESS[®] Interface to PC File Formats



Conclusion

ODBC is a component of the Microsoft Windows Open Services Architecture and it is moving to full SAS CLI conformance. As the product matures, functionality will expand and access speeds will be improved significantly. For now, it is up to the user to maximize the configuration. If a choice of application drivers are available, choose the one that can claim the highest level of ODBC conformance.

The SAS System user might consider licensing SAS/ACCESS Interface to ODBC, for Windows for two main reasons. First, the product is SQL based. Second it offers the advantage of access to data in a growing variety of formats through a single, common interface.

Many users, whose access needs are restricted to Windows applications, can be satisfied by DDE. While DDE is not a direct

interface to source data as is ODBC, it carries the advantage of no additional SAS licensing.

I find little reason to license SAS/ACCESS Interface to PC File Formats. The limited number of PC file formats supported by the product do not justify the additional fees.

Even in ODBC's immature state, there is considerable power in the quantity of applications that have drivers available. SAS Institute Inc. recognizes that ODBC is a standard with considerable merit. However, having read Microsoft's ODBC Programmer's Reference, I question the amount of research and development that was required for the Institute to introduce this interface. For a company such as Blue Cross and Blue Shield of Nebraska that wants to access ORACLE data through SAS for Windows, ODBC is the only alternative. However, the SAS Institute's decision to market the client side of ODBC as a separate product will severely limit our corporate use and push our ORACLE access to competing software vendors.

SAS and SAS/Access are registered trademarks of SAS Institute Inc. in the USA and other countries. Microsoft is a registered trademark of Microsoft Corporation in the United States and in other countries. Windows is a trademark of Microsoft Corporation in the United States and in other countries. dBase is a registered trademark of Ashton-Tate Corporation. ORACLE is a registered trademark of Oracle Corporation.

Jim Anderson
Senior Systems Analyst
Blue Cross Blue Shield of Nebraska
7261 Mercy Road
PO Box 3248
Omaha, Nebraska 68180-0001
402-390-1800 Ex. 1310