Paper 201

Accessing FPL's Data Warehouse Using SAS/ACCESS® SQL Procedure Pass-Through Facility

Guillermo J. Anton, Florida Power & Light, Miami, FL

ABSTRACT

This paper describes **F**lorida **P**ower and **L**ight's (FPL) multi RDBMS (**R**elational **D**ata**B**ase **M**anagement **S**ystem) environment. This environment consists of three databases. The smaller two called CMS (**C**ollections **M**anagement **S**ystem) and SAMS (**S**trategic **A**ccounts **M**anagement **S**ystem) use Sybase SQL (**S**tructured **Q**uery **L**anguage) Server Release 11. The larger one called MDW (**M**arketing **D**ata **W**arehouse) uses Oracle 8i. These are physically located inside 2 IBM RS/6000 computers, linked together to form the FPL Data Warehouse. Each server runs IBM's AIX (IBM's UNIX variant) operating system, Version 4.3.3.

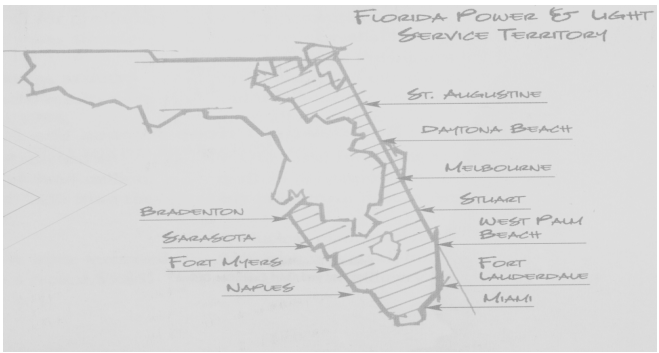
The author describes how the Revenue Recovery Department at FPL uses SAS/ACCESS® (V8) SQL Procedure Pass-Through Facility to retrieve, manipulate, and analyze data from either the Oracle or Sybase databases via ODBC (**O**pen **D**ata**B**ase **C**onnectivity).

INTRODUCTION

In this article, a detailed description of the ODBC setup is depicted. Followed by two examples of SAS/ACCESS® programs used to retrieve data from the FPL Data Warehouse. The goal in this paper is to demystify the usage of SAS/ACCESS® to access various types of relational databases, namely Oracle and Sybase. It should be of interest to both data warehouse practitioners and general SAS® System users. In the future, pursuing a comparison of the ODBC driver versus the native SAS® system driver should be interesting.

THE COMPANY

Florida Power & Light Company is the principal subsidiary of FPL Group. It serves over seven million people (Approximately 3.8 million accounts) along the eastern seaboard and the southern portion of the state of Florida.



THE WAREHOUSE – SOFTWARE

The **M**arketing **D**ata **W**arehouse (MDW) uses Oracle 8i Server, Release 8.1.6 as the database engine. At FPL the **I**nformation **M**anagement Department (IM) uses Trillium Software to perform data extraction, transformation, and cleansing, the art of reorganizing data to make it suitable for the data warehouse. Trillium is used to manipulate the data from sources such as the mainframe, into data sets IM loads into the data warehouse. Trillium Software comes from Harte-Hanks Software (http://www2.harte-hanks.com). One of the distinguishing features that led IM decide to use Trillium was the geocoding capabilities of this software, along with the relative ease of doing many of the tasks usually associated with writing complex UNIX C programs and scripts. Trillium allowed IM to bring several new subject areas on line in a relatively short amount of time, compared with writing their own programs and scripts. The massive data transportation between the IBM mainframe, and the AIX servers is done via Computer Associates's(CA) TransportIT (formerly Platinum's InfoTransport) or FTP (**F**ile **T**ransfer **P**rotocol).

THE WAREHOUSE - HARDWARE

The FPL Data Warehouse is located inside 2 IBM RS/6000 computers. The **M**arketing **D**ata **W**arehouse (MDW) resides in an IBM RS6000 S80 with 12 * 450 MHz processors and 16 GB of RAM. The DASD (**D**irect **A**ccess **S**torage **D**evice) is housed in an IBM **E**nterprise **S**torage **S**erver (ESS). In short, a very powerful **S**ymmetric **M**ulti-**P**rocessor (SMP) machine. CMS and SAMS share the 2nd IBM RS/6000 SP. It is a Sybase server. (http://www.rs6000.ibm.com)

REVENUE RECOVERY

The mission of the Revenue Recovery Department is to increase customer loyalty and encourage good customer payment behavior by deploying competitive credit and collection processes, while managing risk in the most safe and cost effective manner. To accomplish these tasks, we have deployed a very aggressive data mining program. It consists primarily of retrieving, analyzing and exploiting data through SAS/ACCESS® and SAS/STAT®.

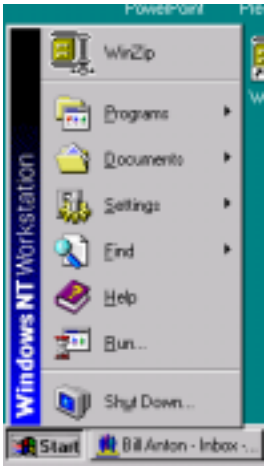
Access to the FPL Data Warehouse is through any ODBC (**O**pen **D**ata**B**ase **C**onnectivity) compliant application, such as Microsoft Access, and SAS/ACCESS®. FPL's **R**evenue **R**ecovery **D**epartment's (RRD) preferred access method is SAS/ACCESS®. With SAS/ACCESS®, RRD is able to extract, modify, combine and analyze very complex queries in a customary manner.

To leverage and complement the software investment, RRD has also licensed an effective product from Canada's Angoss Corporation called KnowledgeSTUDIO (Version 3.0.1). (http://www.angoss.com).

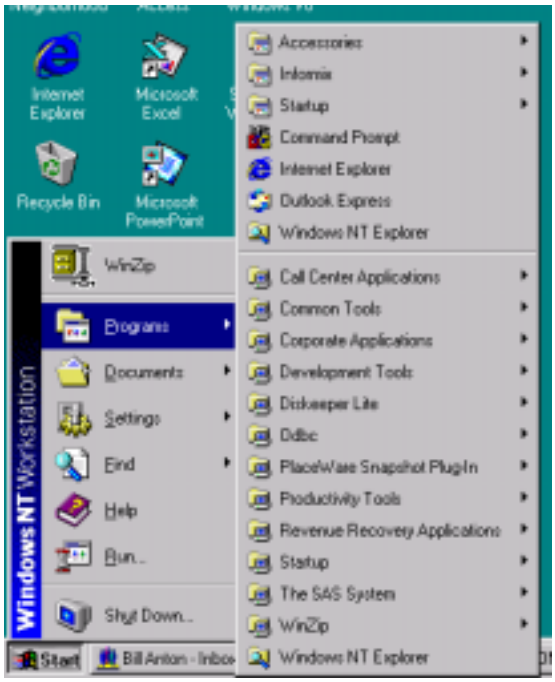
ODBC

At FPL the Microsoft Windows NT environment is tightly controlled. For the purpose of this paper the assumption is made that the required ODBC drivers have already been loaded or are accessible to the Windows NT workstation. To set up the ODBC driver to the MDW (Marketing Data Warehouse) the following steps had to be completed:

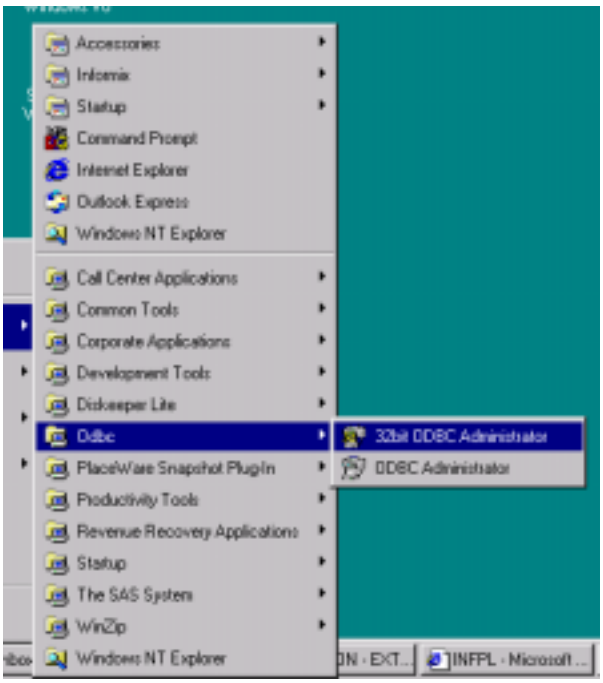
Step 1: Click on the 'Start' Button in the lower left corner of the Windows NT desktop.



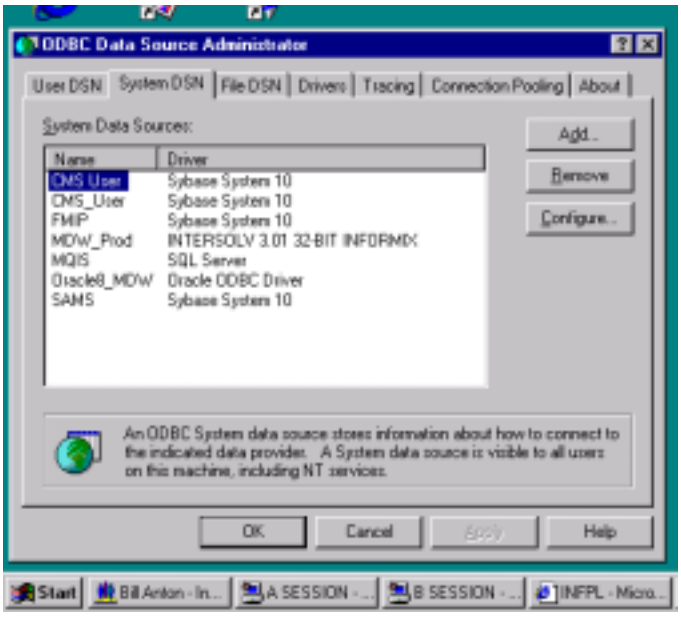
Step 2: Highlight the 'Programs' entry, and navigate to the 'ODBC' entry.



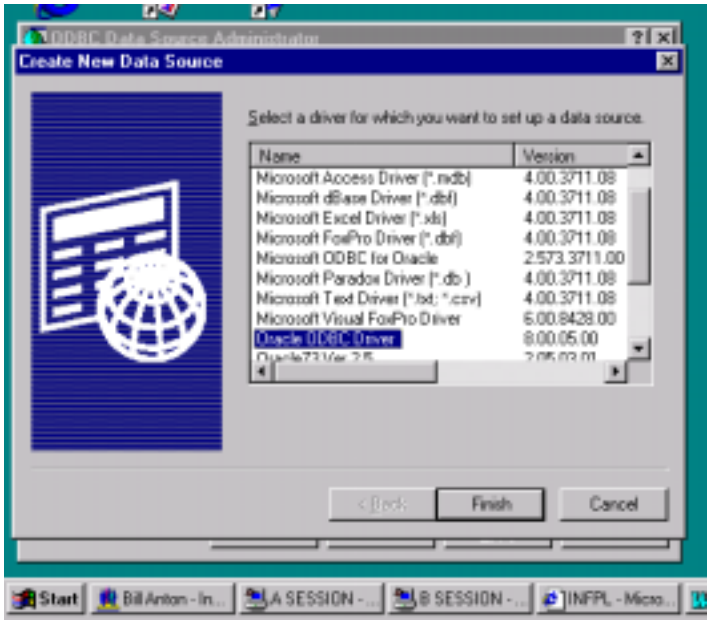
Step 3: Navigate towards the '32bit ODBC Administrator' tag, left click on it, and the "ODBC Data Source Administrator" panel appears.



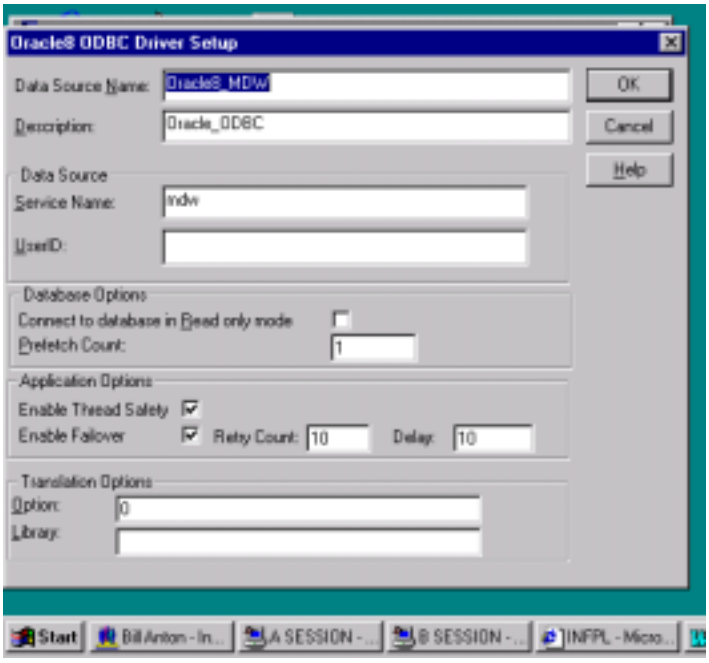
Step 4: Click on the 'System DSN' heading tag. This panel displays the list of drivers currently installed in the workstation. Click on the "Add" button, and this action activates the 'Create New Data Source' panel.



Step 5: The panel below lists all the available ODBC drivers in the workstation. The Oracle Driver supplied by the Oracle Corporation was selected for this exercise.
Highlight the selected driver and click on the 'Finish' button.



Step 6: The "Oracle8 ODBC Driver Setup" panel appears. The entries in this panel are very important since they will determine the ODBC-connection-entries in the SAS /ACCESS® program. In my case for Data Source Name, I chose Oracle8_MDW . The Description is optional, but I entered Oracle_ODBC . The next entry: Data Source Service Name is found in the TNSNAMES.ORA file (All Oracle databases need to be defined in this file). It is similar to Windows ODBC.INI file. In my case it is **mdw**. For all other entries I chose the default values. Finally, click on the OK button to accept the changes.



Step 7: To verify the success of the entire operation. View the contents of the ODBC.INI file. To view this file, choose 'Start' at the bottom left side of the Windows NT desktop, then left click on 'Run' and enter "ODBC.INI". The file will be displayed. An ODBC.INI and a portion of a TNSNAMES.ORA examples are shown below:

ODBC.INI (SAMPLE)

```
[ODBC 32 bit Data Sources]
MS Access 97 Database=Microsoft Access
Driver (*.mdb) (32 bit)
dBASE Files=Microsoft dBase Driver (*.dbf)
(32 bit)
Excel Files=Microsoft Excel Driver (*.xls)
(32 bit)
FoxPro Files=Microsoft FoxPro Driver
(*.dbf) (32 bit)
Text Files=Microsoft Text Driver (*.txt;
*.csv) (32 bit)
Oracle8_MDW=Oracle ODBC Driver (32 bit)
SAMS=Sybase System 10 (32 bit)
CMS_User=Sybase System 10 (32 bit)
[MS Access 97 Database]
Driver32=C:\WINNT\System32\odbcjt32.dll
[dBASE Files]
Driver32=C:\WINNT\System32\odbcjt32.dll
[Excel Files]
Driver32=C:\WINNT\System32\odbcjt32.dll
[FoxPro Files]
Driver32=C:\WINNT\System32\odbcjt32.dll
[Text Files]
Driver32=C:\WINNT\System32\odbcjt32.dll
[Oracle8_MDW]
Driver32=S:\PRODDLLS\orant\bin\sqora32.dll
[SAMS]
Driver32=C:\WINNT\System32\sysybnt.dll
[CMS_User]
Driver32=C:\WINNT\System32\sysybnt.dll
```

TNSNAMES.ORA (EXCERPTED SAMPLE)

```
MDW.fpl.com =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (COMMUNITY =
tcp.fpl.com)(PROTOCOL = TCP)(Host =
mdw0)(Port = 1521))
     )
      (CONNECT_DATA = (SID = MDW))
  )
MDWDEV.fpl.com =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (COMMUNITY =
tcp.fpl.com)(PROTOCOL = TCP)(Host =
goxsd87)(Port = 1521))
     )
     (CONNECT_DATA = (SID = MDWDEV))
  )
MDWT.fpl.com =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (COMMUNITY =
tcp.fpl.com)(PROTOCOL = TCP)(Host =
MDW0)(Port = 1521))
     )
     (CONNECT_DATA = (SID = MDWT))
  )
```

BUSINESS SOLUTION

Two SAS/ACCESS® programs to access the FPL Data Warehouse will be shown. Both use the SAS/ACCESS® SQL Procedure Pass-Through Facility.

An Oracle Example - Source Code:

```
libname  wo_data  'c:\home\sasdata' ;
libname orclodbc  ODBC  dsn=Oracle8_MDW
                uid=gja0oti pwd=xyz1234;
* filename out1  'c:\home\data'    ;

/*  Get some Table Information   */
Proc datasets lib=orclodbc ;
run;
Proc contents data=orclodbc.account;
Proc contents data=orclodbc.premise;
Proc contents data=orclodbc.debits_fact ;
run;

/*  Get Some Data from a Table. In this
     case 'debits_fact'              */

proc sql;
 connect to ODBC (dsn=Oracle8_MDW
                  uid=gja0oti
                  pwd=xyz1234);


 %put &sqlxmsg;

 create table wo_data.odbctest  as
 select * from connection to ODBC

 (select * from  debits_fact
  where (relative_month=199907
  and debit_source_cd='D')       ) ;

%put &sqlxmsg;
 disconnect from ODBC;
proc contents data=wo_data.odbctest ;
run ;
Quit;
```

A Sybase Example - Source Code:

```
libname uarD8456 'c:\home\sasdata' ;
options nocenter yearcutoff=1925 linesize=080
pagesize=055 pageno=01 date ;
Title  "Res & Comm/Ind Accounts 'Written-Off'
From 01/01/2000 To 06/30/2000 ";
Title2 "Sybase Example. 'Written-Off' Accounts.
First 6 Months of 2000";

 proc sql;
  connect to ODBC (dsn=CMS_User
                   uid=gja0oti
                   pwd=xyz1234);
 %put &sqlxmsg;

  create table uarD8456.uaryr99  as
  select * from connection to ODBC

 (select ky_ba, District=cd_dist,
  account_name=nm_compressed,
  payment_history=tx_coll_hist,
  written_off_Date =
  convert(char(10),rpt_dt,102),
  dt_terminated =
  convert(char(10),dt_ba_term,102),
  dt_open =
  convert(char(10),dt_ba_open,102),
  original_connect_dt =
  convert(char(10),dt_orig_conn,102),
  fpl_base_Amt=at_fpl_base_amt, cd_rev,
  rate=cd_tar_sch,
  deposit_type=cd_dep_type, tot_uar_Amt=
  at_uncoll_chg_off,
  premise_type= cd_prem_type,
  waive_reason= cd_waive_rsn,
  premise_no= ky_prem_no, bnkcy=
  cd_bnkcy, rpms_indicator,
  current_diversion_cd=cd_crdv
  from csarsum..ar052_uar_detail
  where ( (rpt_dt between '01/01/2000' and
  '06/30/2000')  ) );

 %put &sqlxmsg;

disconnect from ODBC ;

proc contents  ;
run;
Quit;
```

SAS Institute, Inc. maintains a magnificent technical support section at their Web site. The URL (**U**niform **R**esource **L**ocator) I found must useful for this endeavor is at: (http://www.sas.com/service/techsup/sample/unix_access.html)

CONCLUSION

Deregulation and/or re-regulation of the electric utility industry has caused FPL and other electricity providers to compete for customers. The business of delivering electricity has changed radically, from the traditional focus on accounts, meters, and premises to becoming strong marketers. The FPL data-warehousing project seeks to provide an integrated business solution by consolidating data from many different systems into a coherent customer-centric database. The SAS® System's superb arsenal of data accessibility tools, data management, data exploitation, and reporting capability will enable RRD to deliver timely and extensively detailed customer information to key decision-makers.

REFERENCES

Li, Leiming (1999), "A Process for Automatically Retrieving Database Using ODBC and SAS/ACCESS® SQL Procedure Pass-Through Facility". Proceedings of the Twenty-Fourth SAS® Users Group International Conference. SAS Institute, Inc. Cary, NC

Price, Jennifer (1999), "An Update on SAS® and ODBC". Proceedings of the Twenty-Fourth SAS® Users Group International Conference. SAS Institute, Inc. Cary, NC

Riba, S. David and Riba, Elisabeth (1996), "ODBC: Windows to the Outside World". Proceedings of the Twenty-First SAS® Users Group International Conference. SAS Institute, Inc. Cary, NC

Riba, S. David (1999), "More Windows to the Outside World: Configuring and Using ODBC in Version 7 of the SAS® System". Proceedings of the Twenty-Fourth SAS® Users Group International Conference. SAS Institute, Inc. Cary, NC

SAS Institute, Inc., SAS/ACCESS® Software for Relational Databases: Reference, Version 8 (ODBC Chapter), Cary, NC: SAS Institute, Inc., 1999.

SAS Institute, Inc., SAS/ACCESS® Software for Relational Databases: Reference, Version 8 (ORACLE Chapter), Cary, NC: SAS Institute, Inc., 1999.

SAS Institute, Inc., SAS/ACCESS® Software for Relational Databases: Reference, Version 8 (SYBASE Chapter), Cary, NC: SAS Institute, Inc., 1999.

SAS Institute, Inc., SAS OnlineDoc®, Version 8, February 2000, PDF Format. SAS Institute, Inc. Cary, NC: SAS Institute, Inc., 2000.

ACKNOWLEDGMENTS

Special thank-you to Information Management's (IM) Marketing Data Warehouse (MDW) Department at FPL for their contributions. IM's MDW team developed/built, maintains and supports the MDW.

Thank-you to M. C. Anton, L. A. Lubert, B. R. Perez, and A. M. Ruiz for proofreading the paper.

Thank-you to Heidi Markovitz for inviting me to present this paper in her section at this year's SouthEast SAS® Users Group Conference (SESUG 2K).

CONTACT INFORMATION

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

Guillermo J. Anton
Florida Power & Light - RRD/GO
P. O. Box 029100
Miami, FL 33102-9100
Work Phone: 305-552-4004
Fax: 305-552-2570
Email: bill_anton@fpl.com