Integrating Large Datasets from Multiple Sources
Calgary SAS Users Group (CSUG)

October 25, 2017

Hotel Le-Germain

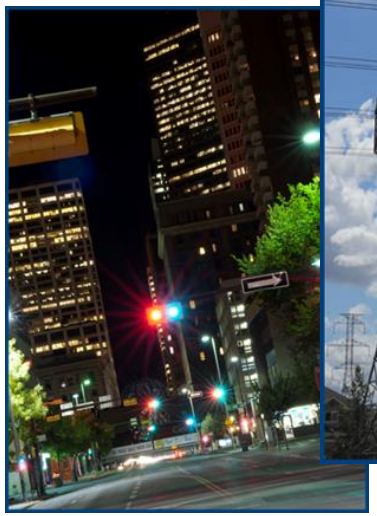
Public

Outline

- About the AESO
- Large Datasets: AESO Context
- Usual Process
  - Obtain data
    - Connecting to ODBC drivers for Oracle and SQL server data
    - Query data on Oracle directly
    - Import from Excel/CSV
  - Process data using Data/Proc steps, including SQL commands
  - Export data to SQL server or Excel for further simulation or analysis and presentation
- About the Presenters and Market Simulation Team

About the AESO

- Independent System Operator that plans/operates provincial transmission grid
- Continuous evaluation; 20-year planning horizon
- Provides access for new generation and load
- Operates wholesale electricity market
  - $2B in 2016 power market transactions; 200+ participants
- Not-for-profit corporation; no grid asset ownership



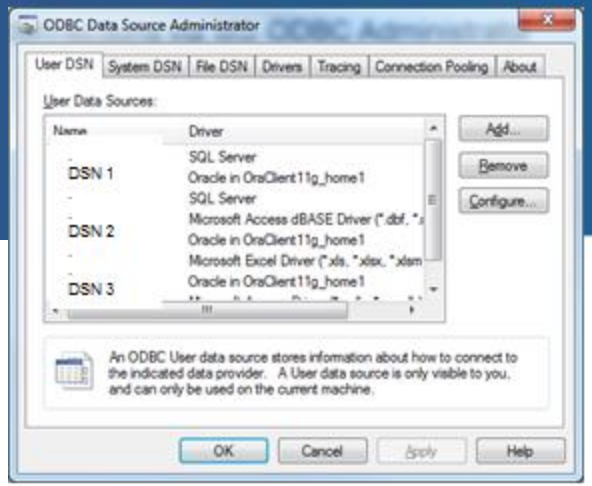
Large datasets - AESO context

- The Alberta Electric System Operator (AESO) manages and operates the provincial power grid.
  - Manage and plan the power grid 24 hours a day (operations data)
  - Manage and operate the energy markets (market data)
  - Plan the future of the system and its infrastructure (simulated operations and market data)

Large datasets - AESO context

- Typical projects for the Forecasting and Analytics team include analyzing multiple repositories of datasets that are not always 100% compatible because of different
  - Intervals (seconds, minutes, hours),
  - Units of analysis (individual power plants, transmission lines),
  - Granularity (individual, regional, Alberta-wide),
  - Time stamps (i.e., multiple versions of same record)
  - Mix of historical and forecast data from internal and external sources
  - Results from multiple iterations (Monte Carlo simulations)
  - Real and nominal dollar values
- Requires blending and merging data carefully, efficiently

Using SAS to Obtain Data



- Connecting and Importing from Oracle
  1. Set up connections via MS ODBC Administrator
  2. Create CSV with database name, data source, user name and password – all as defined in MS ODBC Administrator
  3. Open SAS program node with the following

DB | DB source | User | Password

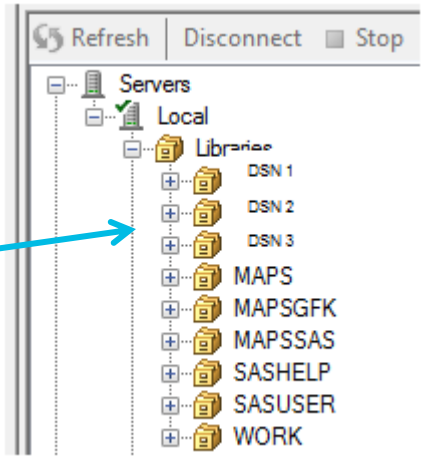
	A	B	C	D
1	Database_Name_1	DSN_Name_1	'Itovar'	"XXXXXXX"
2	Database_Name_2	DSN_Name_2	'Itovar'	"XXXXXXX"
3	Database_Name_3	DSN_Name_3	'Itovar'	"XXXXXXX"
4				

```
filename dbpw 'H:\SAS\dbpw.csv';

%macro libOdbc(lib, db, user, pw);
   libname &lib. odbc
      datasrc="&db." user="&user." pwd="&pw.";
%mend;

data _null_;
   attrib lib length=$32;
   attrib db length=$32;
   attrib user length=$32;
   attrib pw length=$256;
   infile dbpw dsd lrecl=1000;
   input lib db user pw;
   call execute(cats('%libOdbc(',lib,',',db,',',user,',',pw,')'));
run;
```

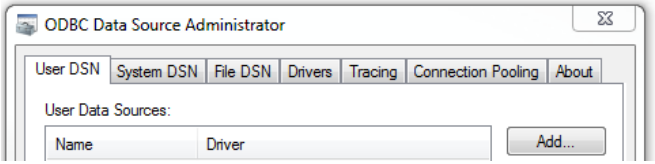
CSV file directory



Voila! Oracle servers are now accessible

Using SAS to Obtain Data

- Connecting and Importing from SQL Server
  1. Set up connections via MS ODBC Administrator
  2. Open SAS program node with the following




```
libname AURORA odbc
datasrc=XXX
noprompt=
"UID=;
PWD=;
DSN=AURORA;
SERVER=XXX;
DATABASE=dataset_repository;";
```

Library name in SAS

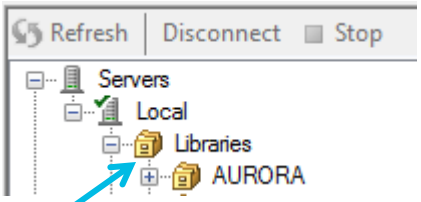
DB source (as in ODBC Admin)

Tip: leave username and password blank if same as Windows login

DB name (as in ODBC Admin)

DB server (as in ODBC Admin)

DB server (as in ODBC Admin)



Voila! SQL server datasets are now added next to Oracle servers

Using SAS to Obtain Data

- Query Data on Oracle or SQL Server Directly

DSN Name

```
proc sql;

   connect to odbc as DB_NAME(datasrc='XXX' autocommit=yes user=&user
pw=&password);

    create table work.XXX as
    select table.*
    from connection to DB_NAME (
        select
              TABLE_NAME.*
           from
              SCHEMA.TABLE TABLE_NAME
           where (1=1)
              and TABLE_NAME.local_date >= &dateBeg
              and TABLE_NAME.local_date < &dateEnd
              and TABLE_NAME.asset_short_name in ('AAA')
    ) table;

    disconnect from DB_NAME;

quit;
```

Need to create prompt message

Using SAS to Obtain Data

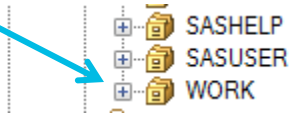
- Importing from Excel / CSV
  - Import a single file

```
proc Import
      datafile="C:\...\File_Name.xlsx"
      out=WORK.XXX
      DBMS="xlsx" REPLACE;
      SHEET="Sheet1";
      GETNAMES=YES;
      RANGE="RANGE_NAME";
Run;
```

  - Import multiple files

```
%macro ImportFile(num);
%do num = 1 %to 10;
proc Import
      datafile="C:\...\File_Name_&num..xlsx"
      out=WORK.XXX_&num.
      DBMS="xlsx" REPLACE;
      SHEET="Sheet1";
      GETNAMES=YES;
      RANGE="RANGE_NAME";
Run;
%end;
%mend ImportFile;
```

Unless otherwise specified, Excel/CSV imports will reside in the temp folder



Processing data

- Analyze Renewable Generation Data
  - Data tables: contain simulated hourly wind and solar data by site from 2014 to 2017

Filter and Sort | Query Builder | Where | Data | Describe | Graph | Analyze | Export | Send To

	Iteration	dt	Date	HE	Year	Month	Weekday	Site	MW
1	1	20JAN2014:23:...	20JAN2014	23	2014	1	1	AESO106220	55.32
2	1	20JAN2014:23:...	20JAN2014	23	2014	1	1	AESO106223	27.97
3	1	20JAN2014:23:...	20JAN2014	23	2014	1	1	AESO106226	50.38
4	1	20JAN2014:23:...	20JAN2014	23	2014	1	1	AESO106227	50.39
5	1	20JAN2014:23:...	20JAN2014	23	2014	1	1	AESO106233	94.83

  - Information table: contains capacity and location of each site

Site	Area	Capacity
AESOsite1	SW	50
AESOsite2	SE	100

  - Merge two tables

    (the power of Hash!)

```
%macro MergeData(year);

    %do year = 2014 %to 2017;

        Data WORK.WIND_&year.;

            if 0 then set SASUSER.SITE_INFO;

            declare hash hmerge(dataset: 'SASUSER.SITE_INFO');
            hmerge.defineKey('Site');
        hmerge.definedata('Site','capacity','Area_Num','Area');
            hmerge.defineDone();

            do until (bEnd);
                set WORK.SIMU_WIND_&year.
                end = bEnd;
                rc = hmerge.find();
                output;
            end;

            stop;
            drop rc;
        Run;

    %end;

%mend MergeData;
```

Processing data

- Select Price Data
  - Background:
    - Multiple years of simulation
    - An even number of iterations
    - One price for each year that is closest to the median price of that year
  - Instead of using data step, we used the following approach:
    - Select the median price for each year (Proc Summary)
    - Calculate the difference between the price from each iteration and the median price for each year (Proc SQL)
    - Find the minimum difference for each year (Proc SQL)
    - Find which iteration that created the median price for that year (the power of Hash!)

Export data

- Processed data can be transferred to SQL server (for further simulations) or Excel (for analysis and presentation)

```
PROC SQL;

     connect to ODBC as DB_NAME
     (datasrc=SERVER_ADDRESS
          noprompt="UID=; PWD=; DSN=DSN_NAME; SERVER=SERVER_NAME;
DATABASE=DATABASE_NAME;");

     EXECUTE(
     CREATE TABLE HOURLY_AESO
        ( ID varchar (255),
          Date varchar(255),
          Hour varchar(255),
          DATA varchar(255),
          AESO_WIND_0 varchar(255),
          AESO_WIND_1 varchar(255),
          AESO_WIND_2 varchar(255),
          …
          [primary key] integer primary key
     )
     )
     BY DB_NAME;
     disconnect from DB_NAME;

QUIT;

PROC Append
Base = AURORA.HOURLY_AESO
Data = WORK.HOURLY_EXPORT FORCE;
Run;

libname AURORA clear;
```

Table created in SAS to be exported to SQL server

About the Presenters and Market Simulation Team

- **Jin Chen** joined the AESO's Market Simulation team as a Senior Analyst, Market Simulation in March 2017. Jin has extensive experience within the Alberta power market in numerous roles that have focused on market simulation, market fundamental and risk quantitative analytics. Through the past 16 years, Jin has been a key contributor in roles including Forecast Specialist and Risk Specialist (ENMAX), Market Analyst (the AESO), Senior Quantitative Analyst and Asset Optimization Analyst (ATCO Power), and Portfolio Analyst (EPCOR Merchant and Capital). Jin has a MA in Economics from the University of Calgary.
- **Leonardo Tovar** joined the AESO's Market Simulation team in February 2017. His immediate focus was on driving changes to AESO's market simulation model, championing the use of tools such as SAS for simulation activities and streamlining processes within the Forecasting and Analytics team to increase efficiency. Prior to joining the AESO, Leonardo has had significant government advisory and analytical experience with the Ontario Ministry of Finance in the roles of Special Policy Advisor and Senior Economist. Leonardo's educational background will support his activities within the team via a Master of Public Policy from the University of Toronto and a BA in both Political Science and Economics from the University of Calgary.

- **About the team**: The AESO's Market Simulation team are market advisors to the AESO and its stakeholders through the creation of market assessments and risk management strategies through detailed analysis of market fundamentals that incorporate a leading edge market simulation methodology. The team is part of Forecasting & Analytics which has a mission to be regarded by the AESO and its stakeholders as a trusted "Centre of Excellence" that provides quantitative and qualitative data-driven recommendations and assessments whilst maintaining an agile and efficient team. The larger team is strategically located within Regulatory & External Affairs to be ready to provide analytic support and advice across the AESO.