



Running SAS Enterprise Guide Locally

- **Setting up SAS Enterprise Guide to access remote data from 'Local'**
(Using SAS EG the same way as Base SAS)
- **Scheduling Jobs from SAS EG**

Presented by Rahman Sarker
Royal Bank of Canada

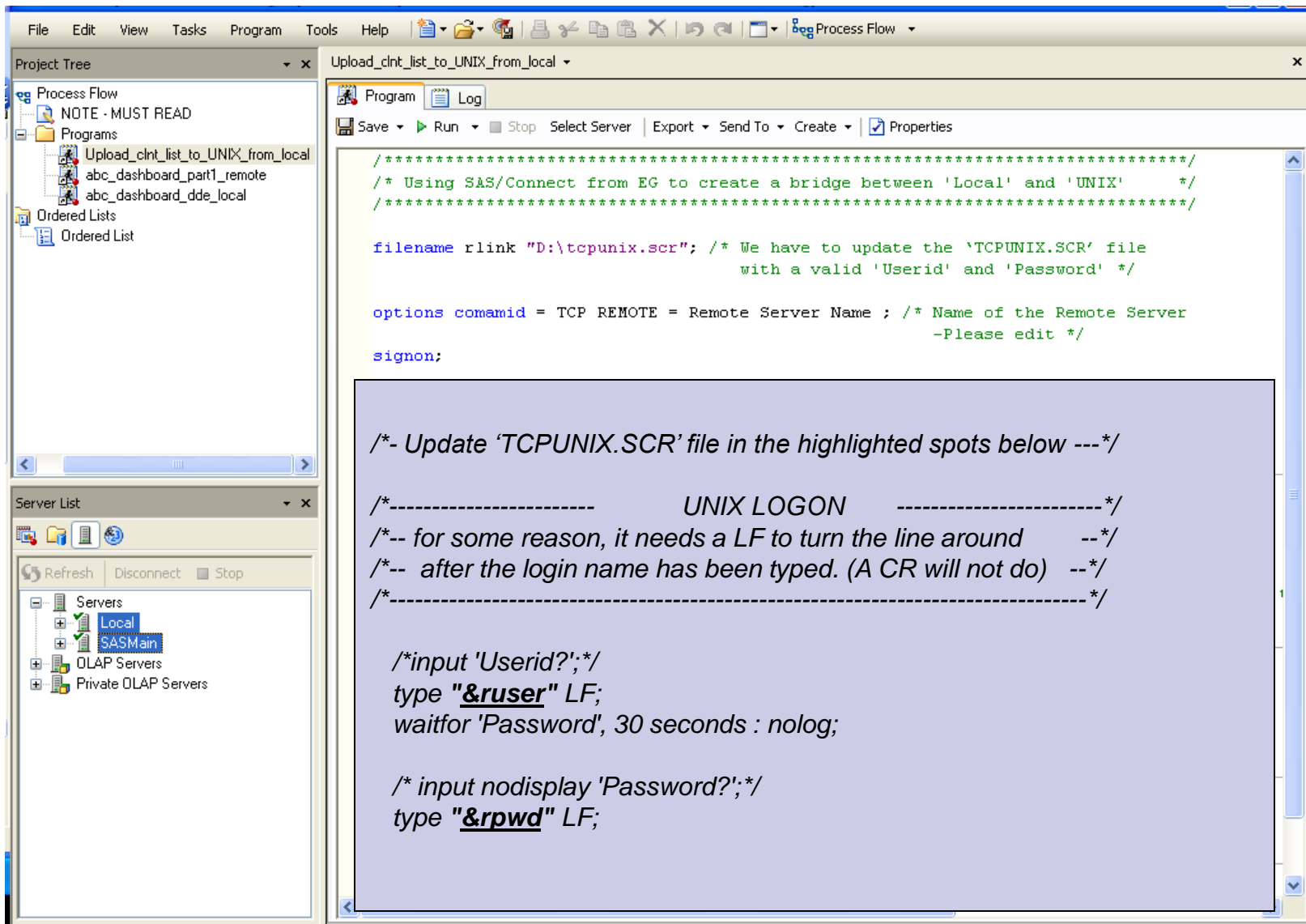
SAS Enterprise Guide

(Typical Scenario for Remote Server Version)

- We connect to remote using Metadata Server (It runs on remote mode - we do not need to print 'RSUBMIT' and 'ENDRSUBMIT' before and after each piece of code)
- 'PROC UPLOAD' and 'PROC DOWNLOAD' do not work
 - **WARNING: PROC UPLOAD must be invoked with the RSUBMIT command.**
- DDE functionality does not work as "X" command usually does not work in EG and SAS EG does not recognize the local machine (local and network drives)
- 'PROC IMPORT' and 'PROC EXPORT' do not work from UNIX to 'Local'

Using SAS/Connect from EG

(Way of doing everything in EG the same way we do in Base SAS)



The screenshot displays the SAS Enterprise Guide (EG) interface. The main window shows a SAS program titled "Upload_clnt_list_to_UNIX_from_local". The program code is as follows:

```
/*----- UNIX LOGON -----*/  
/*-- for some reason, it needs a LF to turn the line around --*/  
/*-- after the login name has been typed. (A CR will not do) --*/  
/*----- UNIX LOGON -----*/  
  
/*input 'Userid?';*/  
type "&ruser" LF;  
waitfor 'Password', 30 seconds : nolog;  
  
/* input nodisplay 'Password?';*/  
type "&rpwd" LF;
```

The code includes comments explaining the purpose of the program: "Using SAS/Connect from EG to create a bridge between 'Local' and 'UNIX'". It also includes instructions for updating the 'TCPUNIX.SCR' file and the 'UNIX LOGON' section.

The interface also shows a Project Tree on the left with folders for "Process Flow", "Programs", and "Ordered Lists". The "Server List" panel at the bottom left shows a list of servers, including "Local" and "SASMain".

Using SAS/Connect from EG

(Way of doing everything in EG the same way we do in Base SAS ... contd.)

The screenshot displays the SAS Enterprise Guide interface. The main window shows a SAS program with the following code:

```
/* *****  
/* Using SAS/Connect from EG to create a bridge between 'Local' and 'UNIX' */  
/* *****  
  
filename rlink "D:\tcpunix.scr"; /* We have to update the 'TCPUNIX.SCR' file  
                                with a valid 'Userid' and 'Password' */  
  
options comamid = TCP REMOTE = Remote Server Name ; /* Name of the Remote Server  
                                                    -Please edit */  
  
signon;  
  
libname pc "T:\Input\repot"; /* Local Data location - Please edit */  
  
%let my_dir = /sas/r_server/data; /* Remote Data location - Please edit */  
libname r_server "&my_dir." server = Remote Server Name;  
  
/* Create client list */  
  
data pc.client_list;  
  
infile "T:\Input\client_list.txt" delimiter = ',' MISSOVER DSD; /* Need to create a demo  
input cust_no;  
run;  
  
/* Upload the client list to Remote Server  
  
rsubmit;  
%let my_dir = /sas/r_server/data;  
libname r_server "&my_dir." ;  
  
proc upload data = pc.client_list  
out = r_server.client_list;  
run;  
endrsubmit;
```

The 'Server List' pane on the left shows a tree view of servers. The 'Local' server is selected. The 'Select Server' dialog box is open, showing a table of servers:

Servers	Description
Local	The SAS server on
SASMain	

The 'Local' server is highlighted in the table. The dialog box has 'OK' and 'Cancel' buttons.

SAS Enterprise Guide

(With customized connection):

- We connect to remote the same way as Base SAS using 'SAS/Connect spawner'
- We can toggle between code that can be run from 'Local' and 'UNIX' server and can get our job done
- Using SAS EG we can organize our project very well and create 'Ordered List' that gives us the control to choose the sequence in which the code needs to be run (please see next slide for example)

SAS Enterprise Guide

(Run job in an organized manner):

The screenshot displays the SAS Enterprise Guide interface. On the left, the 'Project Tree' shows a project named 'Upload_cnt_list_to_UNIX_from_local' with sub-items 'abc_dashboard_part1_remote' and 'abc_dashboard_dde_local'. Below it, the 'Server List' shows a tree of servers including 'Local', 'SASMain', and 'Private OLAP Servers'. The main window shows a SAS program with the following code:

```
/* *****  
/* Using SAS/Connect from EG to create a bridge between 'Local' and 'UNIX' */  
/* *****  
  
filename rlink "D:\tcpunix.scr"; /* We have to update the 'TCPUNIX.SCR' file  
                                with a valid 'Userid' and 'Password' */  
  
options comamid = TCP REMOTE = Remote Server Name ; /* Name of the Remote Server  
                                                    -Please edit */  
  
signon;  
  
libname pc "T:\Input\reprot"; /* Local Data location - Please edit */  
  
%let my_dir = /sas/r_server/data; /* Remote Data location - Please edit */  
libname r_server "%my_dir." server = Remote Server Name;  
  
/* Create client list */  
  
data pc.client_list;  
  infile "T:\Input\c  
  input cust_no;  
run;  
  
/* Upload the client list to the remote server */  
  
rsubmit;  
%let my_dir = /sas  
libname r_server "
```

An 'Ordered List' dialog box is open in the foreground, showing a table of items to be uploaded:

Name	Server
Upload_cnt_list_to_UNIX_from_local	Local
abc_dashboard_part1_remote	SASMain
abc_dashboard_dde_local	Local

The dialog box includes buttons for 'Add', 'Remove', 'Up', 'Run', 'Save', 'Cancel', and 'Help'.

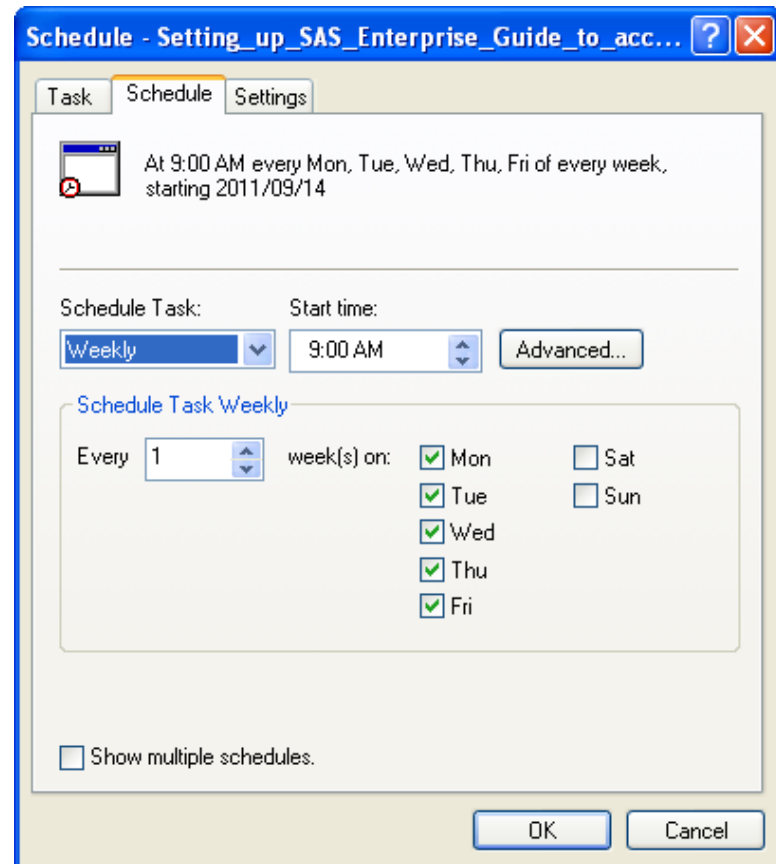
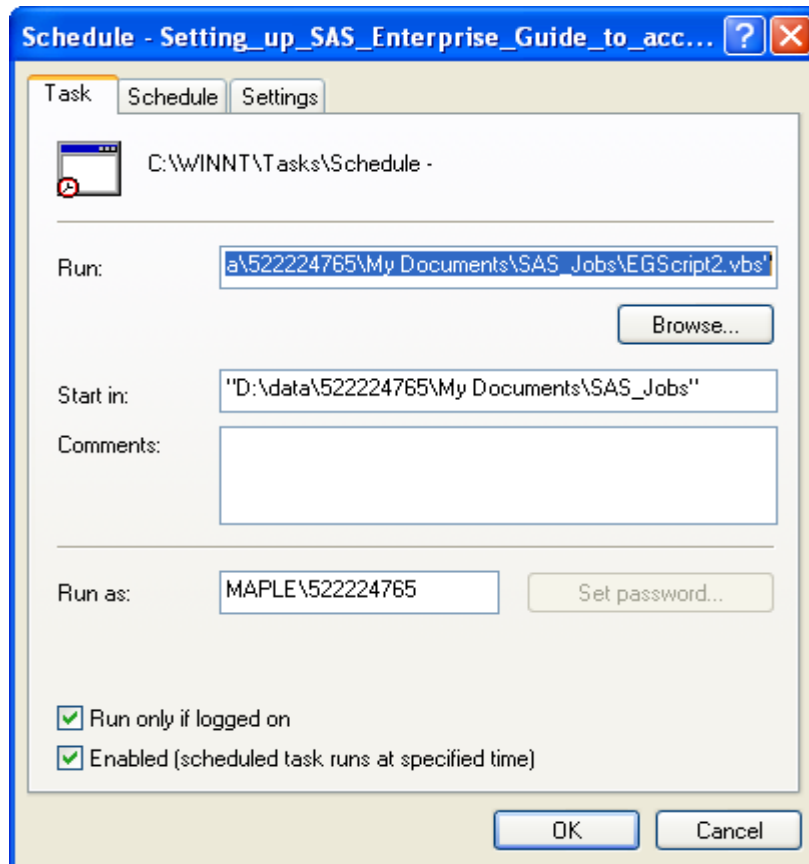
Scheduling Project (SAS EG)

Prerequisites:

- Save the project on the local drive
- Embed all the SAS programs in the project
- Computer has to stay turned on at the scheduled time of the job

How to schedule a job / project (SAS EG – 4.2):

- Open SAS EG project that we want to schedule and click 'File' > Schedule - 'Project Name'
- It should take us to the following screen (final step of Windows Task Scheduler):



Scheduling ...

Good to have 'code' in automated job / program:

- If we want to review the log after it has run, we should use '**proc printto**' options inside the SAS program.

Example:

```
proc printto log =  
"&logdir./&prog._%sysfunc(today()),yymmddn8.)_%sysfunc(compress(%sysfunc(time()),time6.),:)  
.log" new; run;)
```

- Send email confirmation saying that the job is complete.

Example:

```
FILENAME mail EMAIL "rahman.sarker@rbc.com"  
SUBJECT="&sasjob. finished at &tt";  
DATA _NULL_;  
FILE mail;  
  put "Hi Rahman,";  
  put ' ';  
  put "&sasjob. finished at &tt.";  
  put "Check the log here: " ;  
  put "&logdir";  
RUN;
```

Paper Used for this Presentation

- Usage Note 18249: Setting up SAS Enterprise Guide to access remote data from a SAS/Connect spawner or server.

<http://support.sas.com/kb/18/249.html>

- Programming with Enterprise Guide 4 1 (Find Out What You're Missing: Programming with SAS EG - presented in RBC SAS User Group Meeting Oct 21 2008 by Kamran Jafry)



Questions?

Thank you

Contact: rahman.sarker@rbc.com