



## Using Eventus® for WRDS through PC SAS Remote Submit

**Arnold R. Cowan**

*Cowan Research LC and Iowa State University*

This version: December 2007

Wharton Research Data Services offers a web query for Eventus® software, and also provides Eventus in programming mode through SAS® for Unix. The web query is quite user-friendly but may not be suitable for all research uses. The Unix option allows one to use all the power of SAS and Eventus in a flexible way, but many researchers find the Unix environment less efficient or less user-friendly than Microsoft Windows.

A third option is to use your Eventus for WRDS subscription from SAS® on your PC. This document shows how to do so using the Remote Submit method. This method and does not require installing Eventus or the CRSP™ database on your PC. You must have a WRDS account assigned to you by the institution at which you are a current student, faculty member or staff member, and the institution must subscribe to Eventus for access through WRDS. SAS 9.1.3 for Windows is recommended as of this writing.

Special considerations for the Remote Submit method include opening and closing a WRDS remote session, separating remotely executed and locally executed statements, and uploading and downloading files.

### Opening and closing a WRDS remote session

The first step is to tell SAS for Windows to sign in to the WRDS server. Copy the four statements below into the SAS Editor window, and run them. To run statements in SAS, click the “runner” icon, press the F8 key, or select the menu sequence Run, Submit. To run only some of the statements in a given editor window, first select (highlight) them, then run.

```
options nomautosource noimplmac; run;
%let wrds = wrds.wharton.upenn.edu 4016;
options comamid=TCP remote=WRDS;
signon username=_prompt_;
```

A prompt will pop up asking for your WRDS user name and password. Later, when done running remotely submitted statements, end the connection by running the statement:

```
signoff;
```

If SAS log messages indicate that your session is not open when you still want it to be, run the signon statement again.

## Separating server statements and PC statements

Remote Submit requires the user to think about which statements SAS on the WRDS server is to execute and which SAS for Windows is to execute on the PC. The user runs all statements from the PC SAS editor window, but server statements must come between `rsubmit` and `endrsubmit` statements:

```
rsubmit;  
/* server statements here */  
endrsubmit;
```

## Uploading and downloading files

Remote Submit is not quite as simple as working with a local network drive on a PC; the WRDS server cannot directly read a file on your PC or vice-versa. PC files and WRDS server files are logically distinct and must be physically transferred between the two locations when needed. This is easy to do in SAS with `PROC UPLOAD` and `PROC DOWNLOAD`, which are server statements placed between `rsubmit` and `endrsubmit` statements. Use `PROC DOWNLOAD` to copy a server file to the PC and `PROC UPLOAD` to copy a PC file to the server.

To copy a text file or raw data file, first use `filename` statements to create SAS file shortcuts on both the PC and WRDS. This does not apply to SAS data sets. For PC files, as an alternative to a `filename` statement, you can also use the SAS Explorer pane to navigate into SAS File Shortcuts, then right-click and choose New File Shortcut. For WRDS files, the `filename` statement is a server statement. Once the shortcuts are defined, use the `infile` and `outfile` options of the procedures to point to the files through their shortcuts. The syntax is:

```
filename pcfile 'C:\Folder\file.etc';  
rsubmit;  
filename wrdsfile '~\directory\file.etc';  
proc download infile=wrdsfile outfile=pcfile;  
    /* or */  
proc upload infile=pcfile outfile=wrdsfile;  
endrsubmit;
```

where `pcfile` and `wrdsfile` are arbitrary examples of SAS file shortcuts. In `PROC DOWNLOAD`, the file to which `wrdsfile` points should already exist, and `pcfile` should point to the downloaded file to be created. The converse holds for `PROC UPLOAD`. Other server statements can appear between `rsubmit` and `endrsubmit`.

To copy a SAS data set, use the `libname.membername` style of reference and the `data=` and `out=` options of the `PROC DOWNLOAD` and `PROC UPLOAD` statements. The `libname`, or SAS library name, is a shortcut that points to a folder on the PC or a directory on the WRDS server. Use the `libname` statement to define a library name. For PC data sets, one can also use the SAS Explorer to define the `libname` interactively within Librar-

ies. The membertname is the name of a SAS data set within the folder. SAS manages the physical file or files that make up a given data set.

The general syntax for transferring SAS data sets is

```
libname pclib 'C:\Folder\Subfolder';
rsubmit;
libname wrdslib '~\directory\subdirectory';
proc download data=wrdslib.mydsname out=pclib.mydsname;
/* or */
proc upload data=pclib.mydsname out=wrdslib.mydsname;
endrsubmit;
```

where pclib and wrdslib are arbitrary examples of SAS library names. When using PROC DOWNLOAD, wrdslib.mydsname should already exist, as should the folder pointed to by pclib; the procedure creates the PC copy of the data set in pclib. The converse holds for PROC UPLOAD. Other server statements can appear between rsubmit and endrsubmit.

It is not necessary to define library names if you use the pre-defined WORK library on both the PC and the WRDS server. The WORK library is a temporary library; the data sets in it are erased at the end each SAS session. Omit the libname and dot from the SAS data set reference when transferring WORK data sets (but not when using the data sets in Eventus statements). The syntax for this type of transfer is:

```
rsubmit;
proc download data=mydsname out=mydsname;
/* or */
proc upload data=mydsname out=mydsname;
endrsubmit;
```

where mydsname is the name of a data set that already exists in the WORK library of the WRDS server's SAS session in the case of PROC DOWNLOAD or on the PC in the case of PROC UPLOAD. The original and copy data sets can have different names.

If you have a WORK data set on the PC that you do not want to delete when you close SAS, define another library name if you don't already have one. In SAS Explorer, you can then cut and paste the data set from WORK to the other library in the familiar Windows manner.

## Examples

### *Basic example*

The code for the first example performs a few simple tasks to illustrate the basics of working with Eventus for WRDS and SAS for Windows together. The program uploads the user's request file to WRDS, runs an Eventus event study, extracts window CARs to a

SAS data set, downloads the SAS data set, and “prints” it to the SAS for Windows Output pane.

```
/* Define shortcut to request file on PC. */
filename pcreqst 'C:\Documents\file.txt';

/* Sign on to WRDS */
options nomautosource noimplmac; run;
%let wrds = wrds.wharton.upenn.edu 4016;
options comamid=TCP remote=WRDS;
signon username=_prompt_;
/* (user responds to the user name/password popup) */

/* Begin server statements */
rsubmit;
options ps=50;

/* Define shortcut to request file on WRDS Unix server. */
/* ~/ means use my home directory. */
/* File name need not be same on both systems. */
/* Make WRDS shortcut be request as expected by Eventus. */
filename request '~/demorequest.txt';
proc upload infile=pcreqst outfile=request; run;
/* Run the event study and write CAR for selected windows */
/* to temp SAS data set democar. Remove CusiPerm if needed.*/
Eventus;
  Request CusiPerm AutoDate;
  Windows (-10,3) (-5,+5) (-30,+30);
  Evtstudy OutWin=work.democar;

/* Copy democar from WRDS SAS work library to PC work library */
proc download data=democar out=democar; run;

/* End server statement block and close WRDS remote session. */
endrssubmit;
signoff;

/* Print the data set to the SAS for Windows output. */
proc print data=democar;
  id cusip;
run;
```

*Fama-French calendar-time portfolio regression example*

The code for the second example shows how to run the calendar-time portfolio regression method using the Fama-French three-factor model. The example illustrates the following techniques.

- The use of PROC IMPORT in PC SAS to import an Excel spreadsheet (containing CUSIPs and event dates) into a SAS data set. The PROC IMPORT code was written by the File, Import Data menu command in PC SAS; the only manual modification is to add the data set option (rename=(issue\_date=EventDat)), in order to use the column name that Eventus expects for the event date. Current Eventus for WRDS subscribers can download the spreadsheet from <http://www.eventstudy.com/DocLatest/SEOs.xls>. Access to the file requires that your PC be on your subscribing organization's network (physically or possibly through a virtual private network client.)
- The PROC UPLOAD syntax to transfer a SAS data set from the PC to WRDS.
- The Monthly option of the Eventus statement to run a monthly return event study.
- The FFF option to point to the Fama-French factors data set. On WRDS as of this writing, the factors data sets that are in the format Eventus expects are ff.factors\_daily and ff.factors\_monthly. These data sets are maintained by WRDS staff; questions about them should be directed to WRDS support.
- The use of a request file that is a SAS data set, instead of a text file as in the previous example. When the request file is a SAS data set, the INSAS=option of the Request statement must point to it, and no filename statement is used.
- The Shift1=+1 option of the Request statement moves the event date one month later (for a monthly event study), and the EvtStudy statement options Pre=0 Post=23 define the event period as starting with the shifted event month and ending 23 months later. In this example, the dates in the request file are seasoned equity offering dates. Thus, each security will be a member of the calendar-time portfolio for the 24 months following the offering date.
- The FamaFrench option invokes the three-factor model. In the absence of the IRATS or TwoStep options, the FamaFrench option also invokes the calendar-time portfolio method. The Momentum option (not shown) can be added to expand the model to four factors including momentum.
- The ValueWeightSample option of the EvtStudy statement weights each security-event in the calendar-time portfolio by market capitalization. The market capitalization is determined the month before the event period, month -1 in this example. (Future versions will include more alternatives for value-weighted samples.)
- The NoNames option suppresses the listing of the requested security events and the number of months of data found for each, for convenience when viewing the output.

```

PROC IMPORT
  OUT=WORK.SEO_Offer_Request(rename=(issue_date=EventDat))
  DATAFILE=
    'C:\Documents\Eventus\Sample Programs and Data\SEOs.xls'
  DBMS=EXCEL REPLACE;
  SHEET="SEOs$";
  GETNAMES=YES;
  MIXED=YES;
  SCANTEXT=YES;
  USEDATE=YES;
  SCANTIME=YES;
RUN;

/* Sign on to WRDS */
options nomautosource noimplmac; run;
%let wrds = wrds.wharton.upenn.edu 4016;
options comamid=TCP remote=WRDS;
signon username=_prompt_;

/* (user responds to the user name/password popup) */

/* Begin server statements */
rsubmit;
options ps=50;

/* Upload the request file to WRDS. */
proc upload
  data=SEO_Offer_Request
  out=SEO_Offer_Request;
run;

/* Run the calendar-time event study. */
Eventus Monthly FFF=ff.Factors_monthly;
title 'Seasoned Equity Offerings: Offer Date';
title2 'A Sample for Demonstration Purposes Only';
Request insas=work.SEO_Offer_Request CusiPerm Shift1=+1;
Evtstudy Pre=0 Post=23 FamaFrench
  ValueWeightSample NoNames;

endrsubmit;
signoff;

```

The author is the creator of Eventus® software and president and CEO of Cowan Research LC, which is solely responsible for this document. Arnie also is Professor of Finance and Wells Fargo Professor in Finance at Iowa State University.

© 2004–2008 Cowan Research LC.