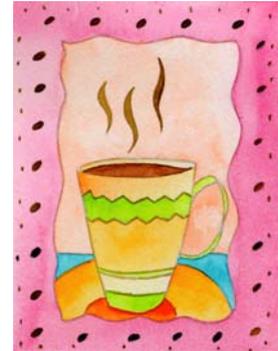# DCSUG Newsletter

Third Quarter Meeting

When: Tuesday, September 14, 2004
Time: 8:30a.m. to Noon
Place: Bureau of Labor Statistics, Conference Center
             Postal Square Building
             Room G440
             2 Massachusetts Ave., N.E.
             (use the First Street entrance)
             Washington DC

Agenda: 8:30 - 9:00   Registration and refreshments
         9:00 - 9:15   Announcements
         9:15 - 10:15   Bruce Gilsen, Federal Reserve Board
                            More Tales from the Help Desk: Solutions for Simple SAS Mistakes
         10:15 - 10:30   Break
         10:30 - 10:45   User-to User Questions and Answers
                            Moderators - DCSUG officers
         10:45 - 11:45   Frank DiIorio, CodeCrafters Inc.
                            The SAS Debugging Primer
         11:45 - Noon   Wrapup
                            *Presentation times subject to change**

Visitors must use the visitor's entrance on First Street NE. Directions for those taking Metro: Take the Red Line to Union Station. Exit the station via the Amtrak exit, but do not go up to the train level. Instead, after going up to and out through the farecard gates, turn to the left where there is an exit to the street (First Street NE). Directly across from the exit are the employee and visitor's entrances to the building. After checking in with security, take the main elevators down to the G level. **BRING PICTURE ID!**

If you have special needs and are planning to attend this meeting, please contact any DCSUG officer no later than 09/02/04

SAS® is a registered trademark of the SAS Institute, Cary, NC

NOTE: If you are not currently a paid member of the DCSUG and are planning to attend the meeting, please email Paul Gorrell at pgorrell@s-3.com or call him at 301-628-3237 by September 8th and leave your name. Security at the Bureau of Labor Statistics is very tight and having your name on a list of possible attendees will make your entrance to the building easier.

****

President's Message
Peter Knapp

With the first day of Autumn (9/22) fast approaching, it's hard for me to believe that the year is going by so quickly. Probably one of the reasons time has been passing so quickly for me is that I'm enrolled in the AIDS Marathon Training Program, a worthy cause that raises money for the Whitman Walker Clinic. Feel free to ask me about this program if you care to.

Of course there's never a dull moment in the SAS world. This year's SESUG is in Nashville, TN from October 31 - November 2. NESUG is a little closer to home in Baltimore, MD from November 14 - 17. Hopefully you'll be able to attend either or both of these regional conferences.

And of course I hope you'll be able to attend the September 14 DCSUG meeting here in Washington. Our featured speakers are Bruce Gilsen and Frank DiIorio who will each give presentations about a topic we've all had to wrestle with -- debugging. I look forward to seeing you at the September meeting.

Speaker Abstracts

More Tales from the Help Desk: Solutions for Simple SAS Mistakes

Bruce Gilsen, Federal Reserve Board

In 20 years as a SAS consultant at the Federal Reserve Board, I have seen SAS users make the same mistakes year after year. This paper reviews the following common mistakes and shows how to fix them.

1. Creating a macro variable and using it in the same DATA step.
2. Overlapping macro variable in a main macro and a called macro.
3. Converting variables from numeric to character and character to numeric.
4. Unexpected values caused by an implicit retain in a MERGE statement.

In the context of reviewing these mistakes, the paper provides details about SAS system processing that can help users employ the SAS system more effectively.

Bruce Gilsen has worked at the Federal Reserve Board for 21 years. He spends most of his time as an in-house SAS consultant. His Bruce Force fantasy baseball team has finished 2nd, 1st, and 2nd the last 3 years.

The SAS Debugging Primer

Frank DiIorio, CodeCrafters Inc.

Meet an accomplished SAS programmer and you meet someone who's probably learned by making (and fixing) lots of mistakes along the way. The breadth of the SAS System's target applications, the variety of its "dialects" (Base SAS, macro, SCL, IML, SQL), and the quirky procedural/non-procedural environmental mix conspire to make mastery of the SAS System a slippery slope to ascend. Debugging is the art of gracefully recovering and learning from falls during the ascent. This paper discusses techniques for debugging SAS programs. Its purpose is two-fold. First, it provides behavioral and technical tips for fixing code (how to read error messages in the SAS Log, knowing when there is a problem with the program even if SAS says there isn't, using the DATA step debugger, identifying system options, using PROCs for data validation, using macro variables to control debugging output, etc.) The second focus of the paper is its presentation of design and coding methods that make the programming process more reliable, thus reducing the need for debugging in the first place.

The paper's target audience is relative newcomers to the SAS System. More seasoned users may find or rediscover some of the techniques and features being discussed. Emphasis is placed on Base SAS and the macro language, although the techniques themselves are applicable to SCL and other products.

Frank DiIorio is President of CodeCrafters, Inc. and the author of "SAS Applications Programming: A Gentle Introduction" and (with Ken Hardy) "Quick Start to Data Analysis with SAS." Both titles are part of SAS Institute's Books by Users series and have sold over 25,000 copies. Frank has been active in the SouthEast SAS Users Group (SESUG) since its inception, co-chairing the 1994 and 1996 conferences.

He has, much to his astonishment after doing the math, nearly 30 years' experience with SAS software. When not writing *about* SAS, Frank writes *in* SAS, primarily data management and reporting applications in the pharmaceutical industry.

Questions and Answers

Moderated by DCSUG Officers

Bring your SAS programming or procedure questions to DCSUG and its members. During this open session, you can ask questions of and get answers from experienced SAS users. All are welcomed to participate.

SAS Puzzler: Why Was DEATH_YR Recoded As 9998 Rather Than 9999?

Carl Pierchala, National Highway Traffic Safety Administration

Consider the SAS code given below. The first step (PROC SQL) extracts selected variables from a dataset with many variables, and puts the selected variables in a better order into the output dataset. The second step (the DATA step) among other things recodes the two digit year numbers like 96 and 97 to four digit year numbers 1996 and 1997 respectively, and recodes 99s to 9999s. At the end of the first step, twelve cases were printed where DEATH_YR=99. But at the end of the second step, the twelve cases were recoded to DEATH_YR=9998 (rather than 9999). Why did this happen, and how do you fix it?   (See page 8 for the answer.)

```
* SELORDWH.SAS REVISED 30-AUG-04;
%let year=1997;
%let yy=%substr(&year,3);
PROC SQL;
CREATE TABLE WORK.LAG_EXPLORE_FARS&yy._PERSON AS
  SELECT ST_CASE, STATE, INJ_SEV,
         LAG_HRS, LAG_MINS,
         &YEAR AS YEAR, MONTH, DAY, HOUR, MINUTE,
         DEATH_YR, DEATH_MO, DEATH_DA, DEATH_HR, DEATH_MN, DEATH_TM
    FROM FARS&yy..PERSON
; QUIT;

title "WORK.LAG_EXPLORE_FARS&yy._PERSON";
Proc print data=WORK.LAG_EXPLORE_FARS&yy._PERSON(where=(death_yr=99));run;

 data _last_;
  set _last_;
    IF YEAR LT 1998 THEN DO;
      IF DEATH_YR=99 THEN DEATH_YR=9999;
        ELSE IF DEATH_YR GT 0 THEN DEATH_YR=DEATH_YR+1900;
    END;
    IF DEATH_YR IN (0,9999) OR DEATH_MO IN (0,99) OR DEATH_DA IN (0,99)
         OR DEATH_HR = 99 OR DEATH_MN = 99
    THEN DO;
       DEATH_DATE=.;
          DEATH_SEC=.;
          END;
    ELSE DO;

DEATH_DATE=INPUT(PUT(DEATH_MO,z2.)||'/'||PUT(DEATH_DA,z2.)||'/'
  ||PUT(DEATH_YR,4.),mmddyy10.);
       FORMAT DEATH_DATE DATE9.;
       DROP DEATH_DATE;
       DEATH_SEC=INPUT(PUT(DEATH_DATE,DATE9.)||'/'||
         PUT(DEATH_HR,z2.)||':'||PUT(DEATH_MN,z2.),DATETIME15.);
       FORMAT DEATH_SEC DATETIME20.;
       END;
RUN;

title "WORK.LAG_EXPLORE_FARS&yy._PERSON";
Proc print data=WORK.LAG_EXPLORE_FARS&yy._PERSON(where=(death_yr gt
2099)); run;
```

SAS-Enhanced Project Schedule and Cost Estimation Techniques

Bryan K. Beverly, DigitalNet

INTRODUCTION

This program demonstrates how managers and developers can use SAS to estimate the schedules and costs of their projects. By 'running the numbers' in planning work, one can reduce the probability of time and cost overruns.

The program is executed through a macro. The macro contains the following parameters: (1) *effort* - the initial 'guestimated' number of person hours needed to perform a task -> usually provided by a subject matter expert, (2) *product* - the percentage of productive hours in an eight hour workday -> eight hours minus meetings, administrative tasks, training, breaks, etc., (3) *avail* - the percentage of productive time that can be devoted to a task -> a person devoting half of their productive time to a task would have an availability value of .5, (4) *resource* - the skill and experience categorization of the person performing the task, (5) *task* - the number that identifies the task being performed and (6) *rate* - the hourly pay rate for the resource.

There are three scenarios represented within the macro invocations - the results from each scenario follow each set of invocations.

SCENARIOS

We have a task that requires 40 hours of effort to complete. We have to determine which one of three resources to use, and determine how to best allocate that resource in the most efficient manner. The senior level resource is fairly productive (the national average of productivity in the U.S is .75), but can only devote a third of his/her time to this task. The mid range level resource is around average productivity, but can put half his/her time on this task. The junior level resource is the least productive of the three resources (due to orientation, training, reading, etc.), but has fewer competing projects and can thus be available two-thirds of his/her productive hours for this task.

```
%macro est(effort=,  /*effort hours needed for the task*/
           product=, /*percentage of productive hours in an 8 hour day*/
           avail=,   /*percentage of time that can be devoted to this task*/
           resource=,/*resource skill and experience level*/
           task=,    /*task number*/
           rate=);   /*hourly pay rate*/

  data proj_est;
    effort_hours=&effort;
    productivity=&product;
    availability=&avail;
    rate=&rate;
    est_hours=round((effort_hours/productivity)/availability); /*Schedule Estimation Formula*/
    est_cost=est_hours*rate
    worker="&resource";
    task=&task;
    prod_index=availability/productivity; /*Productivity Index-> high score = high productivity*/
    difference=est_hours-effort_hours;    /*Estimated Hours - Guestimated Hours*/
  run;

  proc print data=proj_est;
    var task effort_hours worker availability
        prod_index rate est_hours est_cost difference;
    title "Completion Time and Cost Estimates for Task &Task";
  run;

%mend est;

/* SCENARIO #1 - Initial Estimate Based on Existing Resource Allocation */

%est(effort=40,product=.8,avail=.33,resource=Senior Level,task=1,rate=50);
%est(effort=40,product=.7,avail=.5,resource=Mid Range Level,task=1,rate=40);
%est(effort=40,product=.6,avail=.66,resource=Junior Level,task=1,rate=30);
```

As noted in the results below, using the existing allocation mix would make the task completion 8 to 14 days late (a 'difference' range of 61 to 112 hours). Of special note - the most productive resource would have the worst performance. Because he/she was spread so thin across multiple projects, he/she was less than half-productive (41%):

```
SCENARIO #1 - Results from Initial Estimate Based on Existing Resource Allocation


          effort_                                        prod_     est_
  Obs task hours       worker    availability productivity index rate hours est_cost

   1    1    40    Senior Level    0.33          0.8       0.4125  50   152   7600     112
   1    1    40    Mid Range Level 0.5           0.7       0.71429 40   114   4560      74
   1    1    40    Junior Level    0.66          0.6       1.1     30   101   3030      61
```

```
/* SCENARIO #2 - First Adjustment - Increased Availability for Task 1 */
%est(effort=40,product=.8,avail=.85,resource=Senior Level,task=1,rate=50);
%est(effort=40,product=.7,avail=.85,resource=Mid Range
Level,task=1,rate=40%est(effort=40,product=.6,avail=.85,resource=Junior Level,task=1,rate=30);
```

In an attempt to get the task on schedule, we make a task priority change such that 85% of productive work time had to be devoted to Task 1. The results are much better, but the completion date is still late:

```
SCENARIO #2 - Results from First Adjustment - Increased Availability for Task 1

          effort_                                        prod_      est_
   Obs task hours      worker     availability productivity index rate hours est_cost difference


    1   1    40    Senior Level     0.85          0.8      1.0625   50    59   2950       19
    1   1    40    Mid Range Level  0.85          0.7      1.21429  40    67   2680       27
    1   1    40    Junior Level     0.85          0.6      1.41667  30    78   2340       38
```

```
/* SCENARIO #3 - Second Adjustment - Reducing Scope and Increasing Availability  */

%est(effort=35,product=.8,avail=.90,resource=Senior Level,task=1,rate=50);
%est(effort=35,product=.7,avail=.90,resource=Mid Range Level,task=1,rate=40);
%est(effort=35,product=.6,avail=.90,resource=Junior Level,task=1,rate=30);
```

This project is in better shape but still headed for schedule failure. We get approval to reduce the scope by agreeing to develop the 'bells and whistles' later. Hence the effort hours are now 35 instead of 40. Moreover we get approval for each resource to be available for 90% of productive work time for Task 1. The results were improved from the last adjustment, but still beyond the target of 35 hours. Unless there is a management decision to overhaul resource allocations or to allow overtime, then this task will finish 14 hours/two days late - at best:

```
SCENARIO #3 - Results from Second Adjustment - Reducing Scope and Increasing Availability

          effort_                                        prod_      est_
   Obs task hours      worker     availability productivity index rate hours est_cost difference

    1   1    35    Senior Level     0.9           0.8      1.125    50    49   2450       14
    1   1    35    Mid Range Level  0.9           0.7      1.28571  40    56   2240       21
    1   1    35    Junior Level     0.9           0.6      1.5      30    65   1950       30
```

CONCLUSION

This program shows that by using SAS to enhance project management practices, problems that would put the project at schedule or cost risk can be proactively identified and managed. These are some of the take-aways: (1) Unless you have an ideal scenario where a resource is 100% productive and working on only 1 task, then the expected completion date will not match the actual completion date; (2) It is worthwhile to play 'what if' scenarios on the front end to prevent back-end disasters. As in this case, without running the numbers, the most productive and most expensive resource would have taken almost twice as long as the least productive and least expensive resource. In most cases, the natural response would be to use the junior resource without finding out why the senior resource was twice as bad; (3) It is also worthwhile to track tasks during execution. The scenarios presented here assumed a static environment in which nothing changed within the expected 35/40 hour performance periods. If anything occurred that impacted productivity, availability, or cost, then a different outcome would have emerged.

Calendar of Meetings and Events

This feature is meant to give users an idea of some of the activities of area users groups and special interest groups, as well as regional and international groups. If your group would like to be regularly included in this feature, please call Mike Rhoads or one of the DCSUG officers. There is a DCSUG directory with phone, fax, and E-mail contact information elsewhere in this newsletter.

September

14   **DCSUG**   DC SAS Users Group, General Meeting
      Time: 8:30 am – 12:00 pm
      Site: Bureau of Labor Statistics
         2 Massachusetts Ave. NE
      Program: See front page announcement
      Contacts:

         Roger Goodwin at (301) 763-4774
         roger.l.goodwin@census.gov
         Peter Knapp at (202) 482-1359
         peter_knapp@ita.doc.gov

October/November

31-2   **SESUG**   SouthEast SAS Users Group Conference
      Co-Chairs: Imelda Go, Neil Howard
      Site: Nashville
      Further Info: www.sesug.org

November

14-17   **NESUG**   NorthEast SAS Users Group Conference
      Co-Chairs: Warren Stinson, Earl Westerlund
      Site: Baltimore
      Further Info: www.nesug.org

December

3   **DCSUG**   DC SAS Users Group, General Meeting
      Time: 8:30 am - 12:00 noon
      Site: TBA
      Program: TBA

Other Groups

CENSUG   Census SAS Users Group
      Contact: Andrea Potter at (301) 763-1687

MDSUG   Maryland SAS Users Group
      Contact: Jayne May Miller, (410) 455-6849
         jmiller@chpdm.umbc.edu
         www.umbc.edu/chpdm/mdsug/

NCHSSUG   National Center for Health Statistics SAS Users Group
      Contacts: Arlene Siller at (301) 458-4498
         asiller@cdc.gov
         Linda Tompkins at (301) 458-4533

NIHSUG   National Institutes of Health SAS Users Group
      Contact: Ray Danner at (301) 496-6037

NOVASUG   Northern Virginia SAS Users Group
      Contact: Gary McQuown at (703) 628-5681
         gmcquown@dasconsultants.com
         www.novasug.org

SIGSTAT   Special Interest Group in Statistics, CPCUG and WINFORMS
      Contact: Charlie Hallahan, 202-694-5051
         hallahan@ers.usda.gov

VASUG   Virginia SAS Users Group (Richmond area)
      Contact: Claudine Lougee at (804) 968-2083
         president@vasug.org
         www.vasug.org

DCSUG 2004 Steering Committee Officers

CHAIR -- Peter Knapp
(202) 482-1359 (202) 482-1388(fax)
peter_knapp@ita.doc.gov
SECRETARY -- Paul Gorrell
(301) 628-3237 (301) 628-3201 (fax)
pgorrell@s-3.com
TREASURER -- Arlene Siller
(301) 458-4498 (301) 458-4032 (fax)
asiller@cdc.gov
SAS LIAISON -- Roger Goodwin
(301) 763-4774 (301) 457-2744 (fax)
roger.l.goodwin@census.gov
OTHER COMMITTEE MEMBERS
Larry Altmayer
(301) 763-2569 (301) 457-4437 (fax)
Lawrence.W.Altmayer@census.gov
Linda Atkinson
(202) 694-5046 (202) 694-5715 (fax)
Atkinson@ers.usda.gov

Karen Dennis
kcdennis@comcast.net
Frank Fry
(202) 452-2666 (202) 452-6433 (fax)
frank.fry@frb.gov
Mike Rhoads
(301) 251-4308 (301) 294-2040 (fax)
rhoadsm1@westat.com
Howard Schreier
(202) 482-4180 (202) 482-4614 (fax)
Howard_Schreier@ita.doc.gov
Christopher Zogby
(703) 671-6330 (703) 671-6360 (fax)
czogby@erols.com
DESKTOP SPECIAL INTEREST GROUP
CO-CHAIR -- Joe Mirabal
(703) 265-4620
jmmirabal@aol.com
CO-CHAIR -- Yesvy Gustasp
(301) 625-9576 (202) 307-5888 (fax)
Yesvy_Gustasp@yahoo.com

Membership Application/Renewal

Date: _____

Name: _____

Company Name: _____

Mailing Address: _____
_____

City,_____   State _____   Zip _____
Phone # _____   Fax # _____

E-Mail: _____
Preferred mode of contact (check one):
                    ☐ Fax   ☐e-mail
SAS Version(s)_____
SAS Product(s):_____
Operating System(s) _____
Check one:
        ☐ Individual membership   for 1 year is $10.00
        ☐ Corporate membership (group)   for 1 year is $50.00
        ☐ Corporate sponsorship   for 1 year is $250.00

        Check one:
        ☐New membership   ☐Renewal membership
Mail to:
                Washington DC SAS Users Group
                 P.O. Box 44670
                Washington, DC 20026-4670

Puzzler Answer: Why Was DEATH_YR Recoded As 9998 Rather Than 9999?

(Hint: In the first data set, DEATH_YR has a length of 3 bytes.)

In the FARS SAS data sets for years prior to 1998, DEATH_YR was saved with LENGTH 3 (bytes). Prior to 1998, DEATH_YR is stored as a two-digit number, with 99 being the code for Unknown. Beginning in 1998, DEATH_YR is stored in four bytes as a four-digit number, with 9999 being the code for Unknown.) My colleague John Kindelberger gave the following answer to the SAS Puzzler.

> In the first data set, DEATH_YR has a length of 3 bytes. I found written somewhere that 3 is enough to hold integers from 1 to 8192. So, at least four bytes are needed to properly represent 9999.
>
> I found I could get the right answers by adding the statement:
>
> LENGTH DEATH_YR 4 ;
>
> to the second step. (Right after SET _LAST_).

To expand on John's answer, remember that numbers in SAS are stored as a fraction times a power of 2 (the exponent). With a length of 3 bytes, the fractional part of the number has 13 bits. Now, 2**13=8192. If a number is greater than 8192, it needs more than 13 bits for the fraction, but the lower order bits are not saved. So a number written to a SAS data set may only approximate the true value. To illustrate this point, run the following program on your computer and examine the output before and after Obs 8192.

```
* MAXLEN3.SAS REVISED 31OCT03;
data maxlen3;
length i3 3;
do i=1 to 9999;
i3=i;
output;
end;
run;
DATA _LAST_;
SET _LAST_;
I3EQI=(I3 EQ I); * Logical expression: 1=true, 0=false;
title 'MAXLEN3';
PROC PRINT DATA=MAXLEN3;
RUN;
```

Note that if LENGTH 4 were to be used in this program, there would be an additional 8 bits (1 byte) for the fraction, so bigger integers would be saved exactly in WORK.MAXLEN3, but eventually the same behavior would kick in.

Incidentally, from the results from MAXLEN3.SAS, it can be seen (at least on PC systems) that had we in 1998 revised the code for unknown as 9998 rather than 9999, we could have continued to use LENGTH 3 for DEATH_YR in the FARS SAS files.

Nominations for 2005 DCSUG Officers

It's that time of year again. We need to elect officers for DCSUG for next year and we need your help. Please nominate people to fill the following positions. Give your nominations to any current DCSUG officer or send them to:

DCSUG
P.O. Box 44670
Washington, DC 20026

President	Secretary
Treasurer	SAS Liaison

DCSUG Evening Special Interest Group

The Evening Special Interest Group (SIG) will (Re)Start This Fall!

TOPIC: An Introduction to SAS Enterprise Guide

SPEAKER: Yesvy Gustasp

Learn how Enterprise Guide can be used instead of the SAS Display Manager. Submit SAS jobs on a server and use SAS locally. Learn what else Enterprise Guide can do. Generate code you can use right away, or modify it to suit your special task!

DATE/TIME: Planned for late September. Stay tuned.

LOCATION: Social & Scientific Systems, Inc.
        (near the Silver Spring metro)

You are invited to the 12th annual conference of the SouthEast SAS Users Group in Nashville, TN (October 31- November 2, 2004) at the Loews Vanderbilt Hotel!

The SESUG web site at http://www.sesug.org now has the latest information about the conference and links to the online registration system. Register online today by **September 12** to take advantage of early registration savings. Early registration is only $250 for the academic rate and $350 for the nonacademic rate. Past September 12, register by October 10 and save $100 over onsite registration fees.

In Nashville, the music is only half the fun!

WHY YOU SHOULD GO
We'll see you there for ...
* The Opening Session Keynote "Five Things Everyone Needs to Know about the Output Delivery System" by David Kelly, Manager of the Output Delivery System Development at the SAS Worldwide Headquarters in Cary, NC
* The Featured Speaker Session "Business Results Through Relationships" by Barrett Joyner, Senior VP of Sales and Marketing, Mi-Co, Research Triangle, NC
* Over 100 academic presentations and demonstrations
* Hands-on workshops on Monday and Tuesday at no additional charge to attendees

* The Exhibit and Demo Room with the latest in SAS products and services; and publications at a 20% discount
* Unique opportunities to network with SAS professionals in academic and social gatherings
* Optional pre-conference workshops on October 30-31 ($99 for half-day and $198 for full-day workshops)

Optional Pre-Conference Workshops on Saturday, October 30

Time	Instructor	Title
1:00 pm - 5:00 pm	Patricia Cerrito	Using SAS/Stat to Mine Data
1:00 pm - 5:00 pm	Steven First	SAS Efficiencies - Techniques for Improving Your Code

Optional Pre-Conference Workshops on Sunday, October 31

Time	Instructor	Title
8:00 am - 12:00 pm	Steven First	What's New in SAS 9 - An Overview of a Few Hundred New Things
8:00 am - 5:00 pm	David Cassell	Sampling with the SAS System
8:00 am - 5:00 pm	Frank DiIorio	Building SAS Utilities: Putting Dictionary Tables, SQL and the Macro Language to Work
1:00 pm - 5:00pm	Mal Foley	Merging, Combining and Subsetting Files(Tricks, Traps and Techniques)

Note: Photo is courtesy of the Nashville Convention & Visitors Bureau.

NESUG 2004

Registration is now open!

Early Registration ends September 12, 2004

Regular Paper Registration Deadline must be postmarked by November 3, 2004
Regular Web Registration ends November 11, 2004

The seventeenth annual NESUG Conference will be held **November 14 - 17**,
at the Renaissance Harborplace Hotel in Baltimore, Maryland.
The hotel overlooks Baltimore's Inner Harbor and is within walking distance
of many popular attractions, including:

- The National Aquarium
- Maryland Science Center
- Harborplace Pavilion and The Gallery, with restaurants,
shops, boutiques, galleries and sidewalk cafes
- Baltimore Maritime Museum

Other attractions a short distance away are the Walters Art Gallery,
Fort McHenry National Monument and Historic Shrine, USS Constellation Museum,
the B&O Railroad Museum, the Babe Ruth Museum, and Camden Yards.
Baltimore is a beautiful, exciting city!

We are planning an informative, exciting conference for you.
Planned concurrent sessions for Monday, Tuesday and Wednesday include:

- Programming and Manipulation
- Applications
- Analytics
- Presentation
- Storage and Management
- Administration and Support
- Ins and Outs
- Hands-on Workshops
- Posters

Join us also for fun at the Opening Session, the Sunday night mixer,
the NESUG Coffee House Monday evening, and the Tuesday night party.
SAS will provide on-line demonstrations in the staffed Demo Area.

We would like your help:
volunteers are always needed to present papers, serve as section chairs,
session coordinators or registration staff, and aid with the Code Clinic.
Fill out the Conference Volunteer form when you register, or contact the **Conference Co-Chairs**:

Warren Stinson	**Earl Westerlund**
D&B	University of Rochester
Global Decision Sciences	Clinical Trials Coordination Center
899 Eaton Avenue	1351 Mount Hope Avenue, Suite 223
Bethlehem, PA 18025	Rochester, NY 14620
610-882-7158	585-275-8871
stinsonw@dnb.com	**ERWesterlund@nesug.org**

2004 Meetings

Plan ahead and mark your calendars now. The DCSUG meetings in 2004 are scheduled for Sept 14, Dec 3. Meeting times and locations to be announced. If at any time you have a question about an upcoming meeting or other DCSUG information, call one of the members of the Steering Committee.

You are invited

To visit the DCSUG home page

The URL for the DCSUG home page on the Web:
http://dc-sug.org

Remember to Join DCSUG!!!

Don't forget to renew your DCSUG membership for 2004. Dues are used to defray the costs of producing and mailing the newsletter and provide refreshments at our meetings. You'll make sure that you keep receiving the newsletter and keep DCSUG going. If you have not joined DCSUG, now is a great time to do so. Joining DCSUG is as easy as completing the membership form included in this newsletter. Individual memberships are only $10; corporate membership is $50 and corporate sponsorship is $250