A Long-Time SAS® Programmer Learns New Tricks

DASUG Presentation

by Lisa Horwitz,

October 20, 2017

Presenter

Lisa Horwitz, SAS

- Lisa Horwitz is Partner Program Manager in the Global Alliances and Channels organization at SAS.
- Lisa has been at SAS for more than 30 years. Prior to her current role in GAC she was an Instructor in the Education division, a Technical Manager in Sales, and an Advisory Technical Architect in the Customer Loyalty organization. She was the author of a monthly SAS Talks blog and is a frequent presenter at local and regional SAS user group conferences and at SAS Global Forum.
- Lisa holds a B.S. from Old Dominion University in Computer Science and an M.A. in English from the University of Illinois. She is originally from New York City and is currently located in Cary, NC.
- She is a long-time programmer in SAS.

How Long?

USA for Africa - We are the World

Phil Collins - Sussudio

Whitney Houston - Saving All My Love For You

David Bowie & Mick Jagger - Dancing In The Street

Simple Minds - Don't You (Forget About Me)

Starship - We Built This City

How Long?

How Long?

Ronald Reagan

Kansas City Royals

Villanova Wildcats men's basketball

The Project:

- Validate and reconcile data from numerous regularly refreshed sources
- Manipulate and transform data for analysis
- Produce monthly standard reports and numerous additional ad hoc reports.

The Challenge:

The Problem:

After doing things the same way for a long time, it's easy to fall into a programming rut.

The Solution:

Update the logic and try some new techniques that simplify the task.

The Approach:

- SAS® Enterprise Guide to provide the framework, and especially the INPUT task to create SAS data sets from Excel spreadsheets
- For Round 2 of this project, update the logic by modularizing as much as possible to:
  - ensure consistent results
  - simplify the process of making updates

The Approach (continued):

- Incorporate new features (or features new to me!)

  ✓ The PRESENV option and procedure

  ✓ Fuzzy matching with the COMPGED function

  ✓ The ODS POWERPOINT statement

  ✓ SAS Enterprise Guide enhancements.

PRESENV OPTION AND PROCEDURE

PRESENV makes a recording of your SAS programming environment, enabling you to restore it later on.

“Start the recording”

1. Submit the PRESENV option at the beginning of your SAS session:

```
options presenv;
```

Proceed as usual:

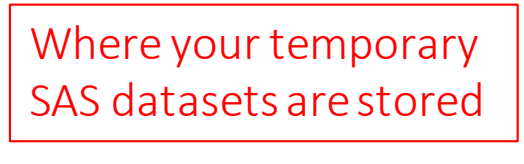
- update process flows in Enterprise Guide
- write and edit SAS code
- run reports
- perform maintenance.

“Stop the recording”

2. At the end of the session, submit the PRESENV procedure with instructions for saving the temporary SAS data sets and SAS code:

```
libname savedata 'c:\presenv\sasdata';
filename savecode 'c:\presenv\code\store.sas';

proc presenv permdir=savedata sascode=savecode
    show_comments;
run;
```

“Stop the recording” (continued)




```
libname savedata 'c:\presenv\sasdata';
filename savecode 'c:\presenv\code\store.sas';

proc presenv permdir=savedata sascode=savecode
    show_comments;
run;
```

“Stop the recording” (continued)

Where the code to restore your session is stored

```
libname savedata 'c:\presenv\sasdata';
filename savecode 'c:\presenv\code\store.sas';

proc presenv permdir=savedata sascode=savecode
    show_comments;
run;
```

Restore the environment

3. Run the code to restore temporary SAS data sets, reassign macro variables, recompile macro code, and reissue global statements and options:

```
%include 'c:\presenv\code\store.sas';
```

Environment restored

Notes in the SAS Log:

SAS data sets are copied from the storage location back into the WORK library

```
NOTE: Copying SAVEDATA.E_E_SORTED to WORK.E_E_SORTED (memtype=DATA).
NOTE: There were 1458 observations read from the data set SAVEDATA.E_E_SORTED.
NOTE: The data set WORK.E_E_SORTED has 1458 observations and 20 variables.
NOTE: Copying SAVEDATA.IDENTIFY_TIERS to WORK.IDENTIFY_TIERS (memtype=DATA).
NOTE: There were 1148 observations read from the data set SAVEDATA.IDENTIFY_TIERS.
NOTE: The data set WORK.IDENTIFY_TIERS has 1148 observations and 16 variables.
NOTE: Copying SAVEDATA.OTHER_ORGANIZATIONS to WORK.OTHER_ORGANIZATIONS (memtype=DATA).
NOTE: There were 1458 observations read from the data set SAVEDATA.OTHER_ORGANIZATIONS.
NOTE: The data set WORK.OTHER_ORGANIZATIONS has 1458 observations and 20 variables.
```

Macro variables are reassigned and macro code is recompiled

```
NOTE: Copying entry CHECKFMT.MACRO from catalog SAVEDATA.SASMACR to catalog WORK.SASMACR.
NOTE: Copying entry ECLIBASSIGN.MACRO from catalog SAVEDATA.SASMACR to catalog WORK.SASMACR.
```

FUZZY MATCHING WITH THE COMPGED FUNCTION

The COMPGED function returns the number of steps it takes to change one character string into another, allowing you to perform fuzzy matches.

The Problem:

You have a lookup table of values against which you must match input data.

For example, an organization name on the lookup table is **A Big Company Inc.** The variations on this name as it appears in input data are endless:

Explanation of Variation	Examples
Name has different corporate designation	A Big Company Ltd.; A Big Company Inc.
Name has different spacing	ABigCompany; A BigCompany
Name has different punctuation	A Big Company, Ltd.; A Big Company Ltd.
Name has different capitalizations	A Big COMPany; A BIG COMPANY
Name is misspelled	A Bigg Company
Name is abbreviated	ABC Ltd.
Name is similar to another organization's name	Big Large Company

The Solution:

Numerous techniques exist in SAS to compare and standardize values.

There's always if-then logic:

```
if trim(left(organization)) in ( 'A Big Company Ltd.',
   'A Big Company Inc.' 'ABIGCOMPANY','A Bigg Company')
   then new_organization='A Big Company Inc.';
```

This technique will get cumbersome and difficult to keep up with, very quickly.

An interesting alternate technique is to use a matching function like COMPGED.

COMPGED measures the number of steps it takes to make one string look like another string.

- COMPGED returns 0 when the two strings are identical
- A certain number of steps are added for each difference in the capitalization, punctuation, spacing, variation of letters, and so on.

```
difference=compged(lookup,organization);
```

- Options available to the function, as well as such additional functions as COMPRESS, TRIM, and so on can refine the results of COMPGED.

COMPGED results:

Lookup value: A Big Company Inc.

```
difference=compged(lookup,organization);
```

organization	difference	difference2
A Big Company Ltd.	300	300
A Big Company Inc.	0	0
ABigCompany	210	150
A BigCompany	200	150
A Big Company, Ltd.	330	300
A Big Company Ltd.	300	300
A Big COMPany	490	150
A BIG COMPANY	990	150
A Bigg Company	210	220
ABC Ltd.	900	800
A Big Company	190	150
Big Large Company	1140	1000

COMPGED results:

Lookup value: A Big Company Inc.

```
difference2=compged(compress(lookup,' ,.'),compress(organization,' ,.'),'i');
```

organization	difference	difference2
A Big Company Ltd.	300	300
A Big Company Inc.	0	0
ABigCompany	210	150
A BigCompany	200	150
A Big Company, Ltd.	330	300
A Big Company Ltd.	300	300
A Big COMPany	490	150
A BIG COMPANY	990	150
A Bigg Company	210	220
ABC Ltd.	900	800
A Big Company	190	150
Big Large Company	1140	1000

Tuning is everything:

- Adjust the amount of compressing, trimming, upper- or lower-casing, etc. based on the values you are comparing

  (for example: does it help or hurt to remove words like “the” and “a” before comparing strings with COMPGED?)

- How many steps provide you with the most accurate matches – 100? 200?

  (for example: increasing the number of allowable steps between strings may increase – or decrease! - the number of false positive matches)

- Review the results carefully – with millions of rows of data, what percentage of mismatches is acceptable?

  (for example: in a mailing address, is it likely that a letter will be delivered correctly if “Avenue” is used instead of “Street”? On the other hand, even one letter wrong will render an email address unusable)

THE ODS POWERPOINT STATEMENT

The ODS POWERPOINT statement writes your SAS procedure output directly to Microsoft® PowerPoint.®

The Challenge:

Requests for summary information in PowerPoint format.

- The numbers change all the time
- Cutting and pasting the information into PowerPoint slides is a bad idea for a lot of reasons, including the likelihood of errors.

The Solution:

ODS POWERPOINT statement.

- Automates the generation of the slides, accounting for changing data values
- Embedding of headers, text, graphs and charts is supported.

The Code:

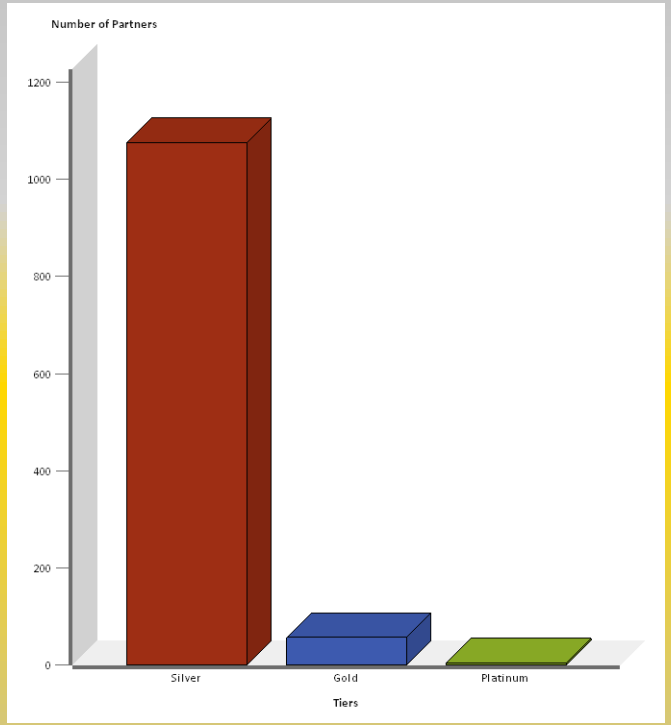
```
ods powerpoint file="c:\SGF 2017\Sample1.pptx"  style=styles.example1
layout=twocontent nogtitle nogfootnote;

proc freq  code

run;

proc gchart  code

run;
```

Other layouts include:

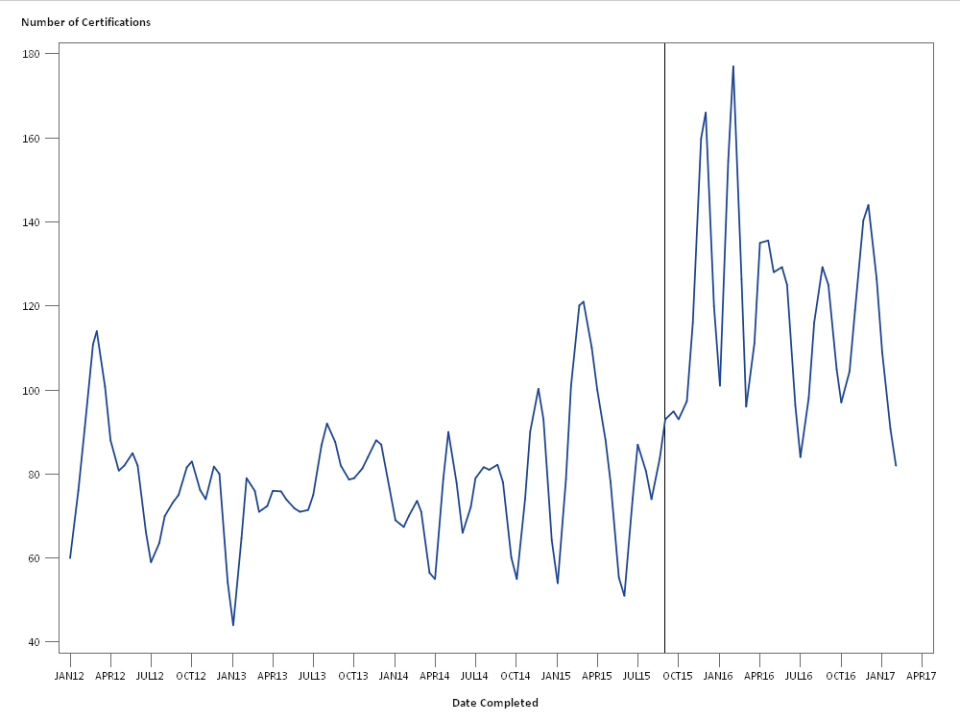
- TITLESLIDE
- _NULL_

Count of Organizations by Tier

Tier	Frequency	Percent
Silver	1075	94.63
Gold	57	5.02
Platinum	4	0.35



Certifications Earned by Partners, 2012-2017



SAS ENTERPRISE GUIDE ENHANCEMENTS

Each new release of SAS Enterprise Guide comes with a list of enhancements to try out and incorporate into your routine.

Some enhancements that I have found especially useful include:

- ✓ Copying and pasting nodes and process flows
- ✓ SAS Macro Variable viewer
- ✓ Data Step DEBUGGER.

The Challenge:

Want to re-use portions of SAS Enterprise Guide process flows:

- Nodes
- Branches
- Entire Process Flows

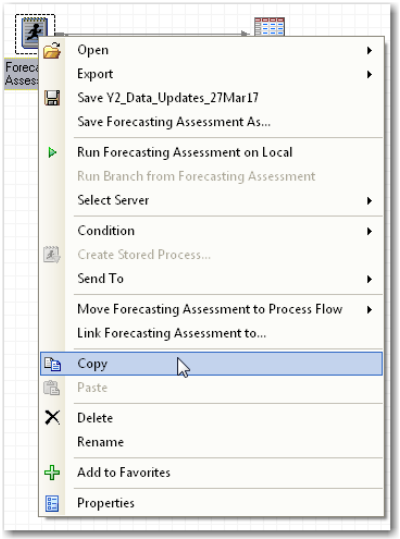
No Challenge at all!

Copy and paste nodes, branches and process flows – very intuitive.

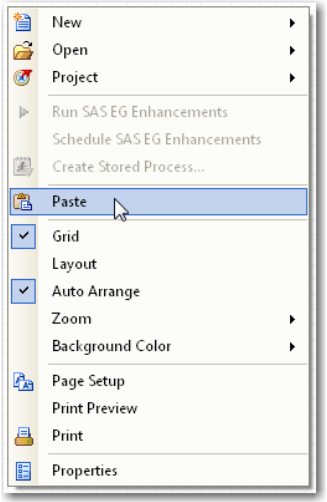
Enhancement #1: Copying and Pasting in SAS Enterprise Guide

to copy nodes:

1. Activate the node by clicking on it

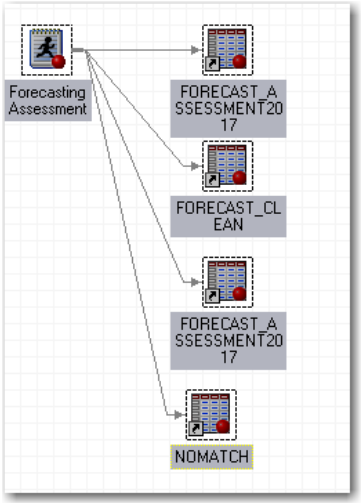


2. Right-click and select Copy

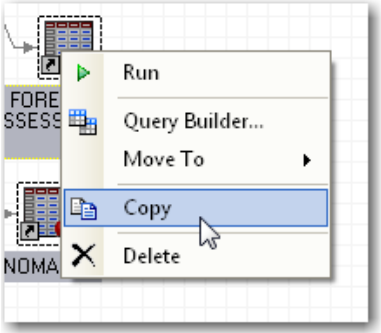


3. Right-click and select Paste at destination.

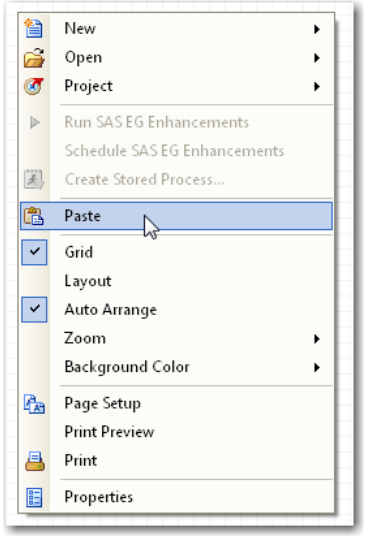
Enhancement #1: Copying and Pasting in SAS Enterprise Guide to copy branches: Option #1



1. Activate the nodes in the process flow by holding down the CNTL-key and clicking on the nodes in the branch

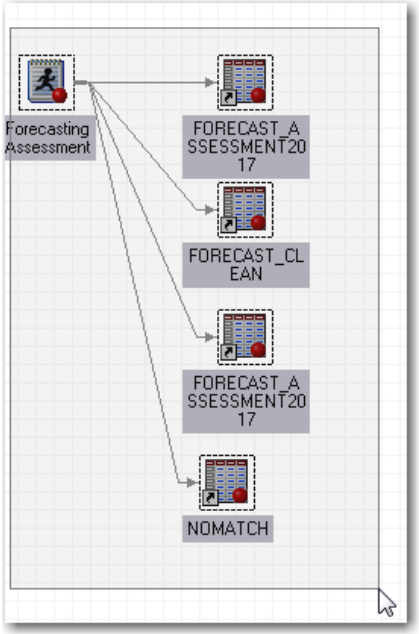


2. Right-click and select Copy

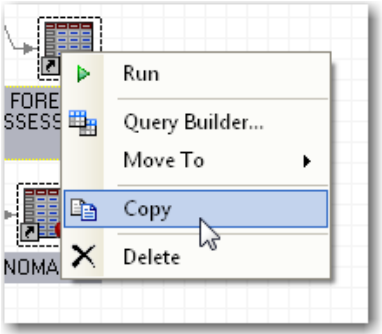


3. Right-click and select Paste at destination.

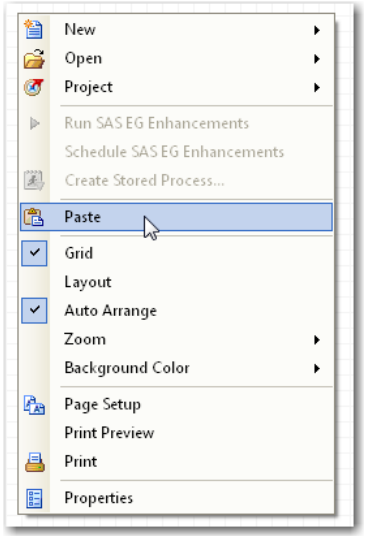
Enhancement #1: Copying and Pasting in SAS Enterprise Guide
to copy branches: Option #2



1. Draw a box around the nodes you want to copy to activate them



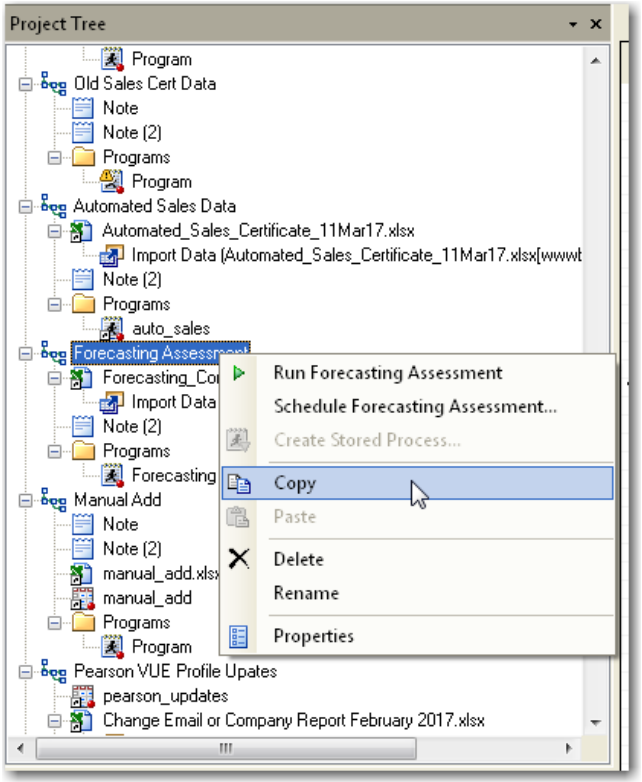
2. Right-click and select Copy



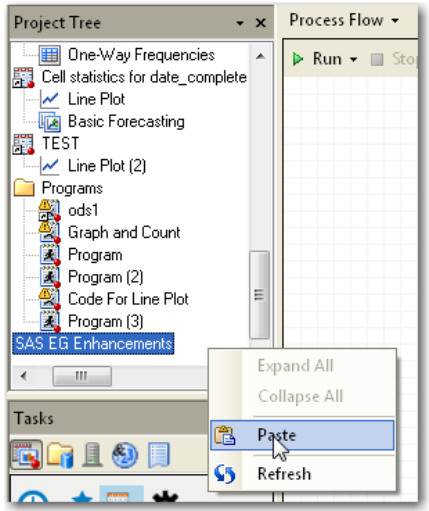
3. Right-click and select Paste at destination.

Enhancement #1: Copying and Pasting in SAS Enterprise Guide

to copy entire process flows:



1. Find the process flow listed in the Project Tree. Click to highlight it, and right-click to select Copy



2. Right-click in the destination Project Tree to Paste it.

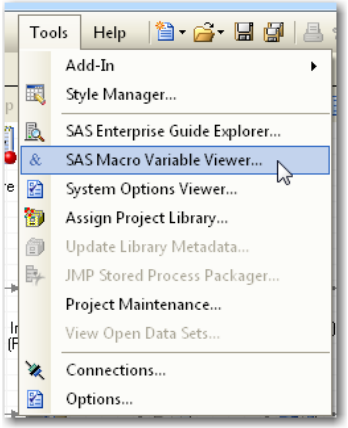
The Request:

To see all the assigned macro variables and their values in your Enterprise Guide Session.

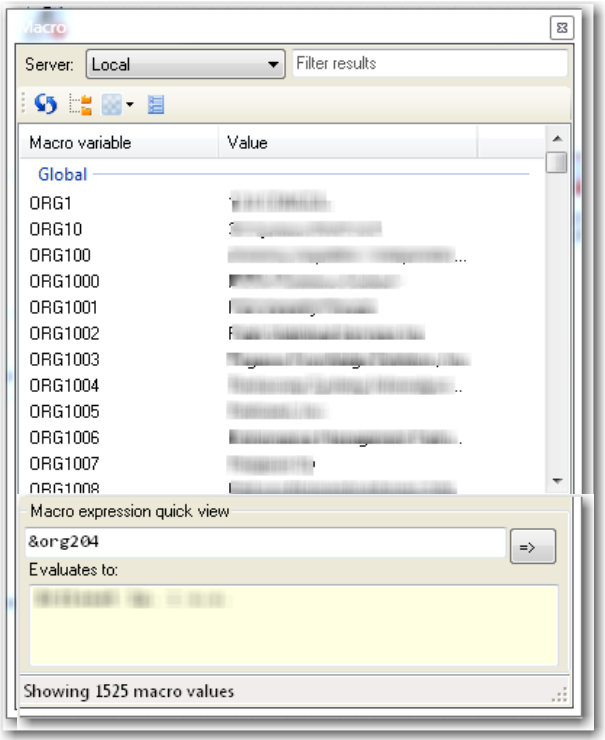
Easy!

The Macro Variable viewer displays system-defined and user-defined macro variables for easy review.

Enhancement #2: Macro Variable Viewer



1. From the Tools pulldown menu, select SAS Macro Variable Viewer



2. All of the assigned macro variables are listed in alphabetical order (order can be changed)

3. You can type macro expressions and see the result, here.

The Wish:

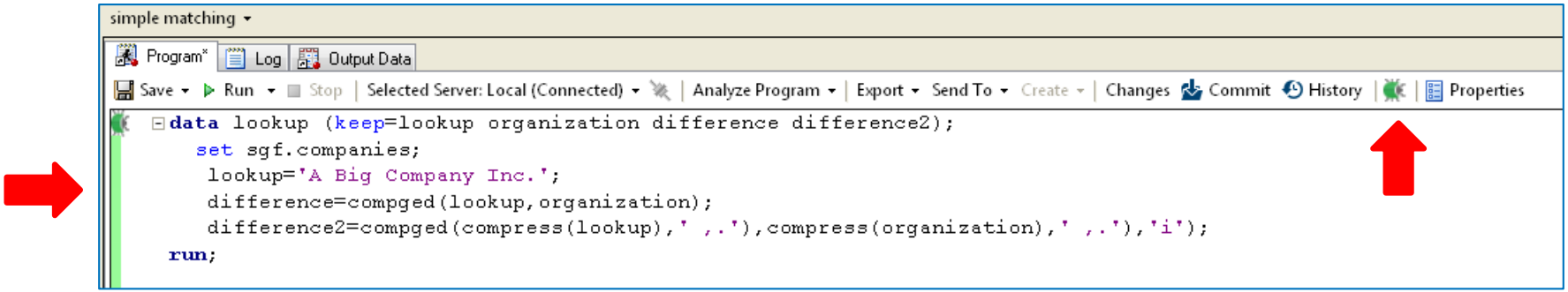
To use the Data Step Debugger in SAS Enterprise Guide.

Wish Granted!

The Data Step Debugger has been incorporated into SAS Enterprise Guide 7.13.

Enhancement #3: The Data Step Debugger available in SAS Enterprise Guide

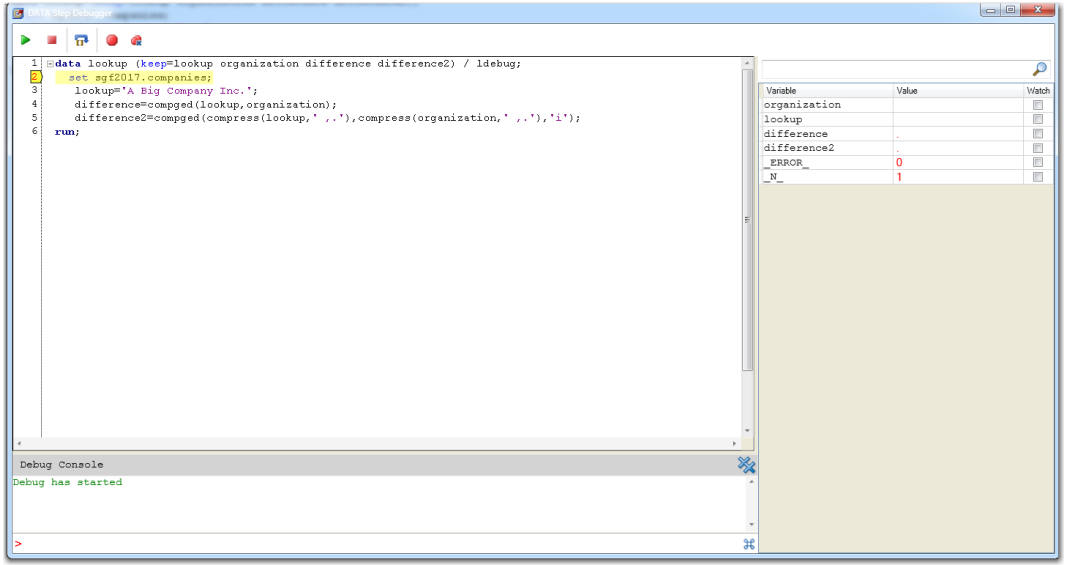
1. You can toggle the debugger on and off by clicking the “bug” toolbar button while viewing your program in the code node.



2. Once turned on, the Debugger highlights the data step to be analyzed.

Enhancement #3: The Data Step Debugger available in SAS Enterprise Guide (continued)

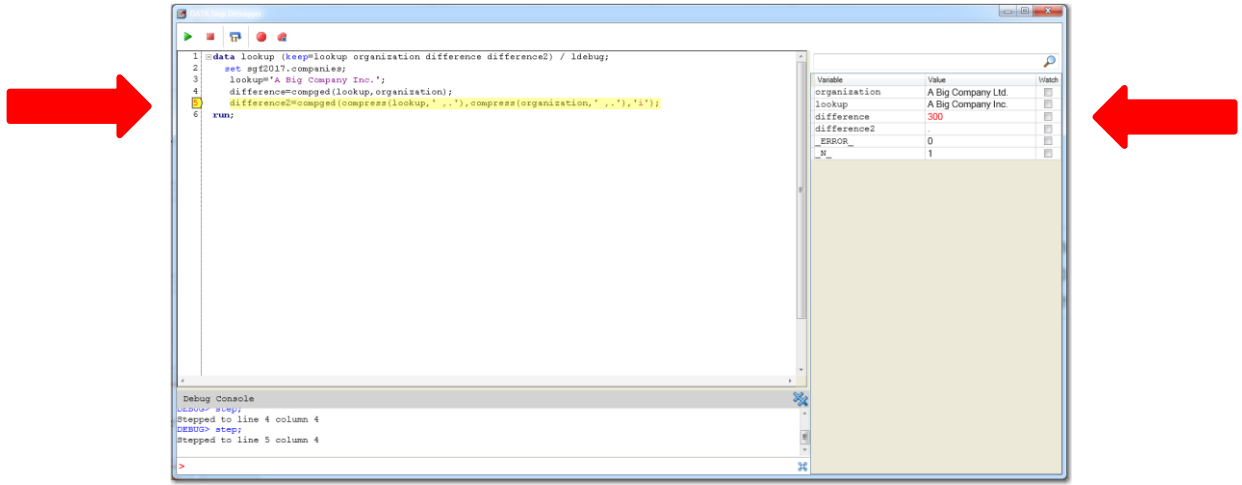
3. Click the bug on the left to open the debugger window with its various components.



4. Use F10 to step through the code or type commands next to the > at the bottom of the screen.

Enhancement #3: The Data Step Debugger available in SAS Enterprise Guide (continued)

5. The line about to be executed is highlighted in yellow. The assigned values appear on the right, with the most recently assigned value in red.



6. It's also possible to jump past lines of code to get to the "troublesome" section, set watchpoints and breakpoints, and much more.

Resources

The SGF paper lists numerous resources providing additional information about everything you've seen here.

- Special thanks go to
  - Maura Stokes (Senior Director, Advanced Analytics R&D, SAS),
  - Amy Peters (Principal Product Manager, SAS),
  - Kathy Wisniewski (Applications Analyst, the University of North Carolina at Chapel Hill), and
  - Merry Rabb (Manager, Public Health, RTI International)

  for their great ideas and excellent counsel.
- And special thanks go to YOU for making this a great ~~conference~~ user group meeting! Thank you!

Contact Information

Please contact me if you have any questions!

Lisa Horwitz

SAS Institute Inc.

SAS Campus Drive

Cary, NC 27513

Lisa.Horwitz@sas.com