

Data Step Debugger in SAS Enterprise Guide What's old is new again

9 February 2018
TASS Interfaces

Charu Shankar
SAS Institute Canada



Agenda

1. Why the SAS data step debugger
2. What's old is new again - EG vs Data step debugger
3. Debugger Commands
4. Demo

1. Why the Datastep debugger

for novices...

- Learn flow of control in data step
- Helps manipulate variables
- Helps fix bugs

and for everyone...

- Spot Syntax, data, logical errors
- No more put statements or poring over the log
- No more staring at pages of code & thinking very very hard :)



1. Why the Datastep debugger for everyone

- Understand legacy code

```
/** Load WordPress Administration APIs */  
require_once( ABSPATH . 'wp-admin/includes/admin.php' );  
  
/** Load Ajax Handlers for WordPress Core */  
require_once( ABSPATH . 'wp-admin/includes/ajax-actions.php' );  
  
@header( 'Content-Type: text/html; charset=' . get_option( 'blog_charset' ) );  
@header( 'X-Robots-Tag: noindex' );  
  
send_nosniff_header();  
nocache_headers();  
  
/* This action is documented in wp-admin/admin.php */
```

2. What's old is new again-DM vs EG debugger

- Initializing the debugger
- Controlling Program Execution
- Suspending Program Execution



2. What's old is new again-DM vs EG debugger

Initializing the debugger

SAS Enterprise Guide

- Highlight DATAstep regions
- Injects LDEBUG option

Base SAS

- Add / DEBUGoption to DATA step

```
data percent(drop=i)/debug;  
  set orion.employee_donations;  
  array Contrib{4} qtr1-qtr4;  
  array Percent{4};  
  Total=sum(of contrib{*});  
  do i=1 to 4;  
    percent{i}=contrib{i}/total;  
  end;  
run;
```

```
proc print data=percent noobs;  
  var Employee_ID percent1-percent4;  
  format percent1-percent4 percent6.;  
run;
```

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

```
data percent(drop=i);  
  set orion.employee_donations;  
  array Contrib{4} qtr1-qtr4;  
  array Percent{4};  
  Total=sum(of contrib{*});  
  do i=1 to 4;  
    percent{i}=contrib{i}/total;  
  end;  
run;
```

The DATA Step Debugger window is open, showing the following code:

```
1 data percent(drop=i) / ldebug;  
2 set orion.employee_donations;  
3 array Contrib{4} qtr1-qtr4;  
4 array Percent{4};  
5 Total=sum(of contrib{*});  
6 do i=1 to 4;  
7   percent{i}=contrib{i}/total;  
8 end;  
9 run;
```

The Debug Console shows the following output:

```
DEBUG> delete break 2;  
Breakpoint 1 cleared at line 2
```

The Variable Value Watch table is as follows:

Variable	Value	Watch
Employee...	.	<input type="checkbox"/>
Qtr1	.	<input type="checkbox"/>
Qtr2	.	<input type="checkbox"/>
Qtr3	.	<input type="checkbox"/>
Qtr4	.	<input type="checkbox"/>
Recipients	.	<input type="checkbox"/>
Paid_By	.	<input type="checkbox"/>
Percent1	.	<input type="checkbox"/>
Percent2	.	<input type="checkbox"/>
Percent3	.	<input type="checkbox"/>
Percent4	.	<input type="checkbox"/>
Total	.	<input type="checkbox"/>

2. What's old is new again-DM vs EG debugger

Initializing the debugger

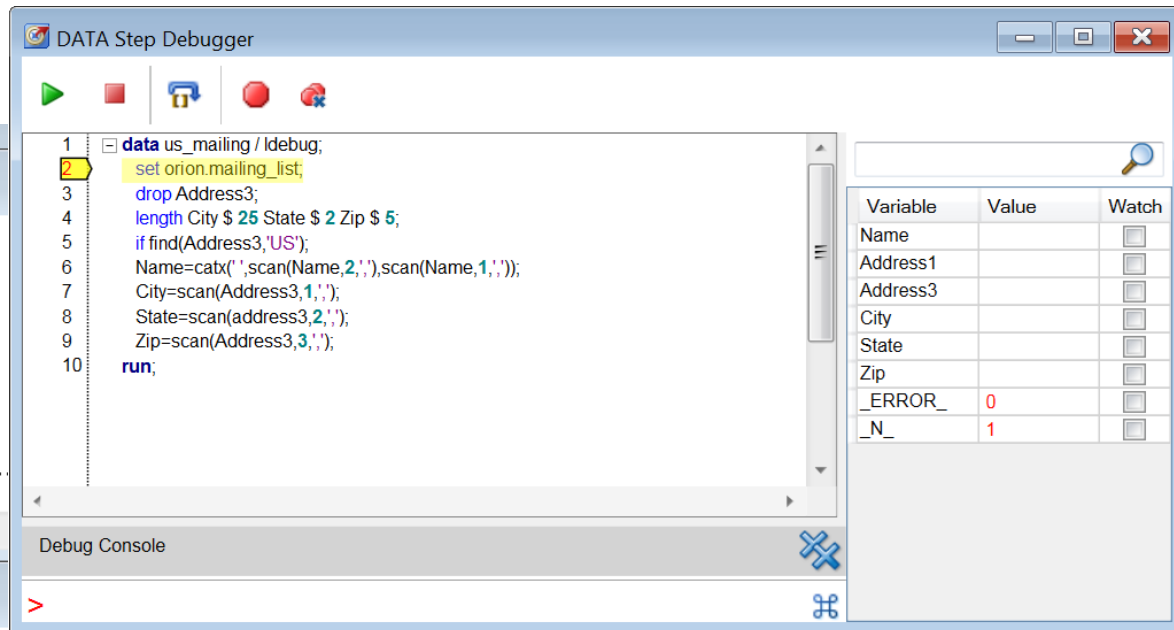
- Compile the step
- Display debugger window
- Pause execution

Is new again

What's old

```
DEBUGGER LOG
DATA STEP Source Level Debugger
Stopped at line 92 column 4
```

```
PROC DEBUGGER SOURCE
91 data percent(drop=i)/debug;
92 set orion.employee_donations;
93 array Contrib{4} qtr1-qtr4;
94 array Percent{4};
95 Total=sum(of contrib{*});
96 do i=1 to 4;
97     percent{i}=contrib{i}/total;
98 end;
99 run;
```



3. Debugger Commands

Command		Action
Step	F10	<ul style="list-style-type: none">• Execute one statement in DATAstep (by default)• Return control to the user
Jump		<ul style="list-style-type: none">• Move execution to specified location• Does not execute intervening statements• Return Control To User
Go	F5	<ul style="list-style-type: none">• Execute until breakpoint, watch variable changes, or DATAstep complete
Break	F9	<ul style="list-style-type: none">• Suspends execution at specified statement• Only suspends when that line is about to be executed
Watch		<ul style="list-style-type: none">• Suspends execution when value of a variable changes

4. Handy links and thank you ...

- [Using the data step debugger in SAS Enterprise Guide](#)
- [Step through your data step](#)
- [The data step debugger](#)

EMAIL Charu.Shankar@sas.com

SAS BLOG <http://blogs.sas.com/content/sastraining/author/charushankar/>

TWITTER CharuYogaCan

LINKEDIN [https://ca.**linkedin.com/in/charushankar**](https://ca.linkedin.com/in/charushankar)