

The SAS Macro Facility:

...an Introduction

Chuck Mohamed
CM Solutions Inc.
Calgary, October 19th, 2004

What is a SAS macro ?

- It depends ...
- Macro variable
- Complete programs written as macros
- Tool to extend and customize SAS

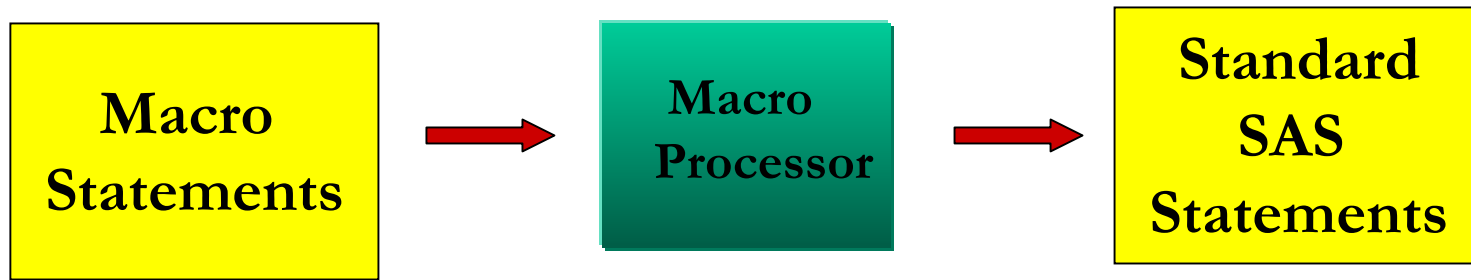
Why use Macros?

1. Dynamically create code at execution time
2. Conditionally execute data steps and procs
3. Communicate across SAS steps
4. Generalize and standardize SAS code
5. Hide complex code that can be invoked easily
6. Extend SAS functionality
7. ...

Standard SAS Process

- Program is compiled and immediately executed.
- When using the Macro facility there is an extra step
- SAS needs to pass your macro statements to the macro processor before compiling

How does SAS execute macros



**Macro statements given to processor
BEFORE the compiler**

Need to resolve macros first

Macro Language Components

- **Macro Variables**
- **Macro Programs**

Macro Variables

- Always store character strings
- Follow SAS naming rules
- Are NOT the same as data step variables
- Stored in memory in a macro symbol table
- Start with &

Macro Variable - examples

- Assign text constants
 - %let type =Medium;
 - %let type = 'Medium';
 - %let repdate = October 2004;
 - %let repdate = October 2004 ;

Quotation marks are stored as part of the macro value
Leading and trailing blanks are not stored

Macro variables – examples

- Assigning numeric variables
 - `%let a = 75;`
 - `%let b = 150;`
 - `%let c = 150 + 75;`
 - What will the value of c resolve to?

Macro variables

- Assign expressions
 - Result is text
 - `%let a = %eval (210 + 40);`
 - `%let a =%eval (210.0/40.0);`
 - `%let a =%sysevalf(210/40);`

Macro variables

- Assign expressions;
 - `%let total_amt =%eval(150+75);`

```
%let A=150;
```

```
%let B=75;
```

```
%let operator=+;
```

```
%put The result of &A &operator &B is %eval(&A  
&operator &B).;
```

Macro variables

- Automatic variables
 - Created by macro processor
 - Contain info about the environment
 - Reference as normal (ie `&sysdate` `&sysbuff`)
 - Begin with “sys...”

SAS

File Edit View Tools Data Solutions Window Help

Enter a Command

VIEWTABLE: Sasdata.Cars

	Model	Country	Type	Weight	TurningRadius	Displacement	Horsepower	GasTank
1	Acura Integra	Japan	Small	2700	37	112	130	13.2
2	Acura Legend V6	Japan	Medium	3265	42	163	160	18
3	Audi 100	Other	Medium	2935	39	141	130	21.1
4	Audi 80	Other	Compact	2670	35	121	108	15.9
5	Audi 90	Other	Compact	2790	35	141	130	15.9
6	BMW 325i	Other	Compact	2895	35	152	168	16.4
7	BMW 535i	Other	Medium	3640	39	209	208	21.1
8	Buick Century	USA	Medium	2880	41	151	110	15.7
9	Buick Electra V6	USA	Large	3350	43	231	165	18
10	Buick Le Sabre V6	USA	Large	3325	42	231	165	18
11	Buick Riviera V6	USA	Medium	3465	41	231	165	18.8
12	Buick Skylark	USA	Compact	2640	39	151	110	13.6
13	Cadillac Brougham V8	USA	Large	4285	44	307	140	25
14	Cadillac De Ville V8	USA	Large	3545	43	273	180	18
15	Cadillac Eldorado V8	USA	Medium	3480	42	273	180	18.8
16	Chevrolet Astro V6	USA	Large	4025	42	262	150	27
17	Chevrolet Beretta	USA	Compact	2655	38	133	95	15.6
18	Chevrolet Camaro V6	USA	Sporty	3110	41	191	140	15.5
19	Chevrolet Camaro V8	USA	Sporty	3320	41	305	170	15.5
20	Chevrolet Caprice V8	USA	Large	3855	42	305	170	25
21	Chevrolet Cavalier	USA	Compact	2485	38	133	95	13.6
22	Chevrolet Corvette V8	USA	Sporty	3280	42	350	250	20
23	Chevrolet Lumina	USA	Medium	3195	42	151	110	17.1
24	Chevrolet Lumina APV V6	USA	Large	3630	42	191	120	20
25	Chrysler Imperial V6	USA	Medium	3570	43	202	150	16
26	Chrysler Le Baron Coupe	USA	Medium	2975	39	153	150	14
27	Chrysler Le Baron V6	USA	Compact	3065	41	181	141	16
28	Chrysler New Yorker V6	USA	Medium	3450	42	202	147	16
29	Dodge Caravan	USA	Large	3385	42	153	100	20
30	Dodge Colt	Japan	Small	2270	32	90	81	13.2
31	Dodge Daytona	USA	Sporty	2885	38	153	100	14

Results Explorer

Log - (Untitl... Editor - Unti... Results Vie... Output - (Un... VIEWTABL...

NOTE: Table has been opened in browse mode.

C:\chucks stuff\sas session

Macro variables

```
Proc print data=sasdata.cars noobs;  
  where type = "&type";  
  var model type weight horsepower;  
  title "Specifications for &type cars as of &reptime";  
Run;
```

Resolves to

```
Proc print data=sasdata.cars noobs;  
  where type="Medium";  
  var model type weight horsepower;  
  title "Specifications for Medium cars as of October 19, 2004";  
Run;
```

Macro programs

- A second SAS programming language for string manipulation
- A program that writes a program
- Begins with a %macro *macroname* statement
- Ends with a %mend *macroname*
- Pass macro parameters to macro program

SAS
File Edit View Tools Data Solutions Window Help

Enter a Command

VIEWTABLE: Sasdata.Cars

	Model	Country	Type	Weight	TurningRadius	Displacement	Horsepower	GasTank
1	Acura Integra	Japan	Small	2700	37	112	130	13.2
2	Acura Legend V6	Japan	Medium	3265	42	163	160	18
3	Audi 100	Other	Medium	2935	39	141	130	21.1
4	Audi 80	Other	Compact	2670	35	121	108	15.9
5	Audi 90	Other	Compact	2790	35	141	130	15.9
6	BMW 325i	Other	Compact	2895	35	152	168	16.4
7	BMW 535i	Other	Medium	3640	39	209	208	21.1
8	Buick Century	USA	Medium	2880	41	151	110	15.7
9	Buick Electra V6	USA	Large	3350	43	231	165	18
10	Buick Le Sabre V6	USA	Large	3325	42	231	165	18
11	Buick Riviera V6	USA	Medium	3465	41	231	165	18.8
12	Buick Skylark	USA	Compact	2640	39	151	110	13.6
13	Cadillac Brougham V8	USA	Large	4285	44	307	140	25
14	Cadillac De Ville V8	USA	Large	3545	43	273	180	18
15	Cadillac Eldorado V8	USA	Medium	3480	42	273	180	18.8
16	Chevrolet Astro V6	USA	Large	4025	42	262	150	27
17	Chevrolet Beretta	USA	Compact	2655	38	133	95	15.6
18	Chevrolet Camaro V6	USA	Sporty	3110	41	191	140	15.5
19	Chevrolet Camaro V8	USA	Sporty	3320	41	305	170	15.5
20	Chevrolet Caprice V8	USA	Large	3855	42	305	170	25
21	Chevrolet Cavalier	USA	Compact	2485	38	133	95	13.6
22	Chevrolet Corvette V8	USA	Sporty	3280	42	350	250	20
23	Chevrolet Lumina	USA	Medium	3195	42	151	110	17.1
24	Chevrolet Lumina APV V6	USA	Large	3630	42	191	120	20
25	Chrysler Imperial V6	USA	Medium	3570	43	202	150	16
26	Chrysler Le Baron Coupe	USA	Medium	2975	39	153	150	14
27	Chrysler Le Baron V6	USA	Compact	3065	41	181	141	16
28	Chrysler New Yorker V6	USA	Medium	3450	42	202	147	16
29	Dodge Caravan	USA	Large	3385	42	153	100	20
30	Dodge Colt	Japan	Small	2270	32	90	81	13.2
31	Dodge Daytona	USA	Sporty	2885	38	153	100	14

Results Explorer

Log - (Untitl... Editor - Unti... Results Vie... Output - (Un... VIEWTABL...

NOTE: Table has been opened in browse mode. C:\chucks stuff\sas session

Example

```
Ods html file='usergrp.htm';  
proc sort in=sasdata.cars out=carsout; by country horsepower;  
proc print data=carsout NOOBS;  
TTITLE Cars of the world by country and horsepower;  
var horsepower model weight horsepower;  
by country;  
run;  
Ods html close;
```

Example

```
%MACRO reportcars (sortseq=, sortvar1=, sortvar2=);  
proc sort in=sasdata.cars out=carsout; by &sortseq &sortvar1 &sortvar2;  
run;  
proc print data=carsout NOOBS;  
TITLE "Cars of the world by &sortvar1 and &sortvar2";  
var &sortvar2 model weight horsepower;  
by &sortvar1;  
run;  
%MEND reportcars;
```

How do we execute a macro program?

- General form is *%macroname*
- If parameters are used
 - *%macroname(parameter1=, parameter2=);*
- In our example
 - *%reportcars(sortseq=, sortvar1=country, sortvar2=horsepower);*

Other features

- CALL SYMPUT
 - Take value from data step and put it into the macro variable
 - Can be used across data steps and procs;

Converting SAS programs

- Are you repeating large sections of SAS code within your programs?
- Are you repeating the same task in many different programs?
- Do you need a standard series of steps to be performed for multiple users?

Troubleshooting techniques

- Make sure statements match such as %MACRO-%MEND, %DO-%END, %IF-%THEN-%ELSE.
- CALL SYMPUTS will not execute until a RUN or another step is encountered.
- Be careful about balancing single and double quotes.
- Use %LOCAL and %GLOBAL when clarification of scope is necessary.
- Use the SYMBOLGEN and MPRINT to trace what the macro is doing.
- Use %PUT statements during testing

Troubleshooting macros

- Printing Macro Expansions
- Macro expansions can be printed by turning on the mprint and symbolgen options:
options mprint symbolgen;
- Use **%put _all_;**

Where is this stuff kept?

- By default SAS compiles Macros in the WORK directory
- Save the macro in a library and %INCLUDE that saved macro when needed.
- Use SASAUTOS system option to specify location of SAS macros

Conclusion

- Macro variables and macro programs allow you to extend and customize the SAS system
- The use of macros in SAS code can streamline your programs and enable you to modularize your code.
- This presentation has been an introduction to some of the features of the SAS macro facility – there's a lot more to be explored!

References and further reading

Carpenter, Art. **Carpenter's Complete Guide to the SAS Macro Language**, Cary, NC, SAS Institute., 1998.

SAS Institute, Inc. **SAS Macro Language: Reference, Version 8**, Cary, NC, SAS Institute., 1999.

Questions ?

Chuck Mohamed

CM Solutions Inc.

(403) 680-4932

Chuckmohamed@shaw.ca