

Understanding and Applying Multilabel Formats



RTSUG
Jan. 14, 2004

Presented By:

Andrew H. Karp



Sierra Information Services, Inc.

19229 Sonoma Highway #264

Sonoma, California 95476 USA

707 996 7380

SierraInfo @ AOL.COM

<http://www.SierraInformation.com>

1

Using MULTILABEL Formats

- New to Version 8, MULTILABEL formats allow you to specify both a
 - Primary label
 - Secondary label
 - To values of variables upon which the format is to be applied
 - Not all SAS PROCs support multilabel formats
 - Only PROCs MEANS, SUMMARY, TABULATE and support MLF's
 - Those that do not (e.g., PRINT, FREQ) utilize only the primary label

2

A Simple Example

- Using MLFs in PROC MEANS to analyze the ages of persons on a data set
 - Children
 - Adolescents
 - Adults

3

A Simple Example

```
data ages;
length name $ 25;
input name $ age @@;
datalines;
amy 5 fred 10 susan 11 tina 9 john 15 stephanie 16
thomas 7 ingrid 19 hannah 16 irving 20 dianne 21 mike 18
mitch 27 maryanne 24 mort 22 debbie 8 peter 2 pamela 30
;
run;

options nonumber nodate;
proc print data=ages;
title 'Using Multilabel Formats';
title2 'Data Set';
run;
```

4

Using Multilabel Formats

Data Set

Obs	name	age
1	amy	5
2	fred	10
3	susan	11
4	tina	9
5	john	15
6	stephanie	16
7	thomas	7
8	ingrid	19
9	hannah	16
10	irving	20
11	dianne	21
12	mike	18
13	mitch	27
14	maryanne	24
15	mort	22
16	debbie	8
17	peter	2
18	pamela	30

Data Set

5

A Simple Example

```

proc format;
value agefmt
low-19 = '0 to 19'
20-high = '20 and older';
run;

```

Creating a "regular" Format

```

proc means n mean median data=ages;
class age;
format age agefmt.;
var age;
title2 'Proc Means with AGEFMT Applied';
run;

```

Associating a "regular" (single label) Format to a Variable

6

A Simple Example

Using Multilabel Formats

Proc Means with AGEFMT Applied

The MEANS Procedure

Analysis Variable : age				
age	N Obs	N	Mean	Median
0 to 19	12	12	11.3333333	10.5000000
20 and older	6	6	24.0000000	23.0000000

7

A Simple Example

```
proc format;  
value age2fmt (multilabel)  
  0-12 = '1) Child'  
  13-19 = '2) Adolescents'  
  0 - 19 = '3) Children & Adolescents'  
  20 - high = '4) Adults';  
run;  
proc means n mean median data=ages;  
class age/mlf; * <== apply the mlf;  
format age age2fmt.;  
var age;  
title2 'Proc Means with Multilabel Format AGE2FMT Applied';  
run;
```

Creating a
Multilabel Format

Applying the
Multilabel Format
in the CLASS
Statement

8

A Simple Example

The MEANS Procedure

Analysis Variable : age				
age	N Obs	N	Mean	Median
1) Child	7	7	7.4285714	8.0000000
2) Adolescents	5	5	16.8000000	16.0000000
3) Children & Adolescents	12	12	11.3333333	10.5000000
4) Adults	6	6	24.0000000	23.0000000

9

A Simple Example

```
proc format;
value age3fmt (multilabel)
  0-12 = 'Child'
  13-19 = 'Adolescents'
  0 - 19 = 'Children & Adolescents'
  20 - high = 'Adults';
run;
proc means n mean median data=ages;
class age/mlf; * <= apply the mlf;
format age age3fmt.;
var age;
title2 'Proc Means with Multilabel Format AGE3FMT Applied';
run;
```

10

A Simple Example

Using Multilabel Formats

Proc Means with Multilabel Format AGE3FMT Applied

The MEANS Procedure

Analysis Variable : age				
age	N Obs	N	Mean	Median
Adolescents	5	5	16.8000000	16.0000000
Adults	6	6	24.0000000	23.0000000
Child	7	7	7.4285714	8.0000000
Children & Adolescents	12	12	11.3333333	10.5000000

11

A Work Around

- The next section shows a fairly simple “work around” that can assist in applying MLFs

12

Using MULTILABEL Formats

- Example:
 - create a MLF (Multilabel Format) for the two digit variable in ADVREPT.TRANSACTIONS representing credit card type.
 - This Data Set contains 336,000 observations, each representing a single credit card transaction.
 - Used in the “Advanced Reporting with the Output Delivery System” seminar

13

Using MULTILABEL Formats

```
proc format library=advrept;  
  value $cardf (multilabel)  
  'VC' = 'Visa Classic'  
  'MC' = 'Master Card Classic'  
  'VP' = 'Visa Platinum'  
  'MP' = 'Master Card Platinum'  
  'MG' = 'Master Card Gold'  
  'VG' = 'Visa Gold'  
  'VC','VP','VG' = 'Total, Visa'  
  'MC','MG','MP' = 'Total, Master Card';  
run;
```

Primary Label

Secondary Label

14

Using MULTILABEL Formats

```

options fmtsearch=(advrept) nonumber nodate nocenter;
proc freq data=advrept.transactions;
tables cardtype;
format cardtype $cardf.; ←
title 'Advanced Reporting';
title2 'PROC FREQ OUTPUT';
run;
    
```

Since PROC FREQ does not support the use of multilabel formats, only the primary label is used when format \$CARDF. is applied in this task.

15

Using MULTILABEL Formats

Advanced Reporting
PROC FREQ OUTPUT

The FREQ Procedure

Credit Card Type

Result of using a multilabel format with a PROC FREQ, which does not support this capability. Only the primary labels are utilized.

cardtype	Frequency	Percent	Cumulative Frequency	Cumulative Percent
Master Card Classic	52798	20.32	52798	20.32
Master Card Gold	86611	33.34	139409	53.67
Master Card Platinum	34746	13.38	174155	67.04
Visa Classic	52733	20.30	226888	87.34
Visa Gold	14665	5.65	241553	92.99
Visa Platinum	18221	7.01	259774	100.00

16

Using MULTILABEL Formats

```
proc means data=advrept.transactions
  maxdec = 2 mean sum p50 nway;
format cardtype $cardf. charge_amount dollar16.2;
class cardtype/mlf; ←
var charge_amount;
output out=x mean= sum= p50=/autoname;
run;

proc print data=x;
title2 'Multilabel Formats: Data Set Created by PROC MEANS';
run;
```

17

Using MULTILABEL Formats

Advanced Reporting
Multilabel Formats: Data Set Created by PROC MEANS
The MEANS Procedure
Analysis Variable : charge_amount Transaction Charge Amount

Credit Card Type	N Obs	Mean	Sum	50th Pct1
Master Card Classic	52798	515.30	27206890.11	515.16
Master Card Gold	86611	515.05	44609348.78	515.77
Master Card Platinum	34746	522.78	18164543.35	523.77
Total, Master Card	174155	516.67	89980782.24	517.42
Total, Visa	85619	513.63	43976349.40	512.39
Visa Classic	52733	513.32	27068908.04	511.67
Visa Gold	14665	509.72	7475032.87	507.06
Visa Platinum	18221	517.67	9432408.49	518.86

18

Using MULTILABEL Formats

Advanced Reporting
Multilabel Formats: Data Set Created by PROC MEANS

Obs	cardtype	_TYPE_	_FREQ_	charge_amount_ Mean	charge_amount_ Sum	charge_amount_ P50
1	Master Card Classic	1	52798	\$515.30	\$27,206,890.11	\$515.16
2	Master Card Gold	1	86611	\$515.05	\$44,609,348.78	\$515.77
3	Master Card Platinum	1	34746	\$522.78	\$18,164,543.35	\$523.77
4	Total, Master Card	1	174155	\$516.67	\$89,980,782.24	\$517.42
5	Total, Visa	1	85619	\$513.63	\$43,976,349.40	\$512.39
6	Visa Classic	1	52733	\$513.32	\$27,068,908.04	\$511.67
7	Visa Gold	1	14685	\$509.72	\$7,475,032.87	\$507.06
8	Visa Platinum	1	18221	\$517.67	\$9,432,408.49	\$518.86

19

Using MULTILABEL Formats

- A Work-Around if the Formatted Values are not portrayed correctly
 - A new format is created. Each value label has a number in the first column, so that they are portrayed in the desired order.
 - Two Version 8 enhancements are used in the PROC MEANS task
 - **DESCENDTYPES**: orders the observations in the output data set by descending value of _TYPE_, so that _TYPE_ = 0 is the last observation
 - **CHARTYPE**: converts the default numeric variable _TYPE_ to a character variable.

20

Using MULTILABEL Formats

```
proc format library=advrept;
  value $card2f (multilabel)
    'VC' = '5Visa Classic'
    'MC' = '1Master Card Classic'
    'VP' = '7Visa Platinum'
    'MP' = '3Master Card Platinum'
    'MG' = '2Master Card Gold'
    'VG' = '6Visa Gold'
    'VC','VP','VG' = '8Subtotal, Visa'
    'MC','MG','MP' = '4Subtotal, Master Card';
run;
```

21

Using MULTILABEL Formats

```
proc means data=advrept.transactions
  NOPRINT descendtypes chartype;
format cardtype $card2f. ;
class cardtype/mlf;
var charge_amount;
output out=x(rename=( _FREQ_ = trans_count))
  mean= sum= p50=/autoname;
run;
```

22

Using MULTILABEL Formats

Advanced Reporting
 Multilabel Formats: Data Set Created by PROC MEANS
 A Work-Around for Problems with MLFs

Obs	cardtype	_TYPE_	trans_ count	charge_ amount_ Mean	charge_ amount_ Sum	charge_ amount_ P50
1	1Master Card Classic	1	52798	515.302	27206890.11	515.160
2	2Master Card Gold	1	86611	515.054	44609348.78	515.770
3	3Master Card Platinum	1	34746	522.781	18164543.35	523.765
4	4Subtotal, Master Card	1	174155	516.671	89980782.24	517.420
5	5Visa Classic	1	52733	513.320	27068908.04	511.670
6	6Visa Gold	1	14665	509.719	7475032.87	507.060
7	7Visa Platinum	1	18221	517.667	9432408.49	518.860
8	8Subtotal, Visa	1	85619	513.628	43976349.40	512.390
9		0	259774	515.668	133957131.64	515.730

23

Using MULTILABEL Formats

- Now that the observations in the output data set are ordered correctly, all we need to do is
 - Strip off the first column of the variable CARDTYPE
 - Replace the blank space where _TYPE_ = 0 with <<< GRAND TOTAL >>>
 - These are both accomplished in the Data Step on the next page.

24

Using MULTILABEL Formats

The LENGTH programming language function returns the byte-length of a character variable. In the Assignment Statement creating CARD, the SUBSTR function starts at the second byte of CARDTYPE and the length of the substring is the byte length of CARDTYPE minus 1.

```
* clean up card type;
data x2(drop=cardtype _type_);
  set x;
  * _TYPE_ is character;
  if _type_ = '1' then do;
    card = substr(cardtype,2) ;
  end;
  else do;
    card = '<<< Grand Total >>>';
  end;
run;
```

25

Using MULTILABEL Formats

```
proc print data=x2;
  format trans_count comma12. charge_amount_mean
    charge_amount_sum charge_amount_p50 dollar16.2;
  id card;
  title3 'Result of Work-Around for MLFs';
run;
```

26

Using MULTILABEL Formats

Advanced Reporting

Multilabel Formats: Data Set Created by PROC MEANS

Result of Work-Around for MLFs

card	trans_count	charge_amount_ Mean	charge_amount_ Sum	charge_amount_ P50
Master Card Classic	52,798	\$515.30	\$27,206,890.11	\$515.16
Master Card Gold	86,611	\$515.05	\$44,609,348.78	\$515.77
Master Card Platinum	34,746	\$522.78	\$18,164,543.35	\$523.77
Subtotal, Master Card	174,155	\$516.67	\$89,980,782.24	\$517.42
Visa Classic	52,733	\$513.32	\$27,068,908.04	\$511.67
Visa Gold	14,665	\$509.72	\$7,475,032.87	\$507.06
Visa Platinum	18,221	\$517.67	\$9,432,408.49	\$518.86
Subtotal, Visa	85,619	\$513.63	\$43,976,349.40	\$512.39
<<< Grand Total >>>	259,774	\$515.67	\$133,957,131.64	\$515.73

27

Multilabel Formats

- Benefits
 - Creates multiple levels of groupings/subgroupings
 - Can avoid tedious data step coding or merging of multiple data sets
 - Flexible
 - Easy to create
 - Easy to modify

28

Multilabel Formats

- Limitations
 - In Version 8
 - NOTSORTED option in PROC FORMAT is disabled when MULTILABEL option is also specified in the VALUE Statement.
 - This issue will be addressed in Release 9.1
 - MLFs only used by three BASE SAS Procedures
 - MEANS
 - SUMMARY
 - TABULATE

29

Example

- Analytic Situation
 - Two reports are generated by an analyst at a credit card company
 - Analysis of credit card charges aggregated/summarized by
 - Charge Amount Decile
 - Charge Amount Quintile
 - Credit Card
 - » American Express
 - » Discover
 - » 3 VISA “Products”
 - » 3 MasterCard “Products”

30

Example

- Goal
 - How can we show analyses of charges at different combinations of credit card types?

31

The Multilabel Format Facility

```
proc format;
  value $multi2f (multilabel)
    'VC' = 'Visa Classic'
    'VP' = 'Visa Platinum'
    'VG' = 'Visa Gold'
    'VP','VG','VC' = 'All Visa Products'
    'MC' = 'MasterCard Classic'
    'MP' = 'MasterCard Platinum'
    'MG' = 'MasterCard Gold'
    'MC','MG','MP' = 'All MasterCard Products'
    'VP','VC','VG','MC','MG','MP' =
      'All Visa & MasterCard Products'
    'AX' = 'American Express'
    'DC' = 'Discover Card';
run;
```

32

The Multilabel Format Facility

FORMAT NAME: \$MULTI2F LENGTH: 30 NUMBER OF VALUES: 8		
MIN LENGTH: 1 MAX LENGTH: 40 DEFAULT LENGTH 30 FUZZ: 0		
START	END	LABEL (VER. 8.2 18OCT2002:07:47:10)
AX	AX	American Express
DC	DC	Discover Card
MC	MC	MasterCard Classic
MC	MC	All MasterCard Products
MC	MC	All Visa & MasterCard Products
MG	MG	MasterCard Gold
MG	MG	All MasterCard Products
MG	MG	All Visa & MasterCard Products
MP	MP	MasterCard Platinum
MP	MP	All MasterCard Products
MP	MP	All Visa & MasterCard Products
VC	VC	Visa Classic
VC	VC	All Visa Products
VC	VC	All Visa & MasterCard Products
VG	VG	Visa Gold
VG	VG	All Visa Products
VG	VG	All Visa & MasterCard Products
VP	VP	Visa Platinum
VP	VP	All Visa Products
VP	VP	All Visa & MasterCard Products

33

The Multilabel Format Facility

```

proc means noprint data=deciles descendtypes;
label charge_amount = 'Charge Amounts ($)';
format charge_decile multi1f.;
class charge_decile/mlf; * <== apply the mlf;
var charge_amount;
output out=data5 mean= sum= min= max=
/autoname autolabel;

proc print data=data5 label split = ' ';
title 'Example 1: Using a Multilabel Format';
run;

```

34

The Multilabel Format Facility

Example 1: Using a Multilabel Format

Obs	cardtype	_TYPE_	_FREQ_	Charge Amounts (\$)_Mean	Charge Amounts (\$)_Sum	Charge Amounts (\$)_Min	Charge Amounts (\$)_Max
1	All MasterCard Products	1	174155	516.671	89980782.24	3.43	1295.22
2	All Visa & MasterCard Products	1	259774	515.668	133957131.64	2.19	1295.22
3	All Visa Products	1	85619	513.628	43976349.40	2.19	1262.18
4	MasterCard Classic	1	52798	515.302	27206890.11	10.05	1256.23
5	MasterCard Gold	1	86611	515.054	44609348.78	14.07	1249.00
6	MasterCard Platinum	1	34746	522.781	18164543.35	3.43	1295.22
7	Visa Classic	1	52733	513.320	27068908.04	17.54	1257.39
8	Visa Gold	1	14665	509.719	7475032.87	2.19	1208.21
9	Visa Platinum	1	18221	517.667	9432408.49	16.94	1262.18
10		0	259774	515.668	133957131.64	2.19	

35

What Happened?

- The results shown on the previous page are correct, but...
 - The ordering of the rows in the report are “wrong”...or, are they?
 - The format labels are stored in “sort order”
 - In Version 8, PROC FORMAT's NOTSORTED option cannot be used with a Multilabel Format

36


A Work-Around for Version 8

- One way to overcome this limitation is to re-build the MLF and put numbers in each of the value labels so that the resulting format will be stored in the desired order.

37

A Work-Around for Version 8

```
proc format;  
  value $multi3f (multilabel)  
    'VC' = '07Visa Classic'  
    'VP' = '08Visa Platinum'  
    'VG' = '09Visa Gold'  
    'VP','VG','VC' = '10All Visa Products'  
    'MC' = '03MasterCard Classic'  
    'MP' = '05MasterCard Platinum'  
    'MG' = '04MasterCard Gold'  
    'MC','MG','MP' = '06All MasterCard Products'  
    'VP','VC','VG','MC','MG','MP' =  
      '11All Visa & MasterCard Products'  
    'AX' = '01American Express'  
    'DC' = '02Discover Card';  
run;
```



38

A Work-Around for Version 8

```
proc means noprint data=deciles descendtypes completetypes;
label charge_amount = 'Charge Amounts ($)';
format cardtype $multi3f.;
class cardtype/mlf PRELOADFMT; * <== apply the MLF;
var charge_amount;
output out=data7 mean= sum= max= min=/autoname autolabel;
run;

proc print data=data7 label split=' ';
title 'Using an MLF with the COMPLETETYPES and';
title2 'PRELOADFMT Options';
run;
```

39

A Work-Around for Version 8

Using an MLF with the COMPLETETYPES and
PRELOADFMT Options
A Work-Around

Obs	cardtype	_TYPE_	_FREQ_	Charge Amounts (\$)_Mean	Charge Amounts (\$)_Sum
1	01American Express	1	0	.	.
2	02Discover Card	1	0	.	.
3	03MasterCard Classic	1	52798	515.302	27206890.11
4	04MasterCard Gold	1	86611	515.054	44609348.78
5	05MasterCard Platinum	1	34746	522.781	18164543.35
6	06All MasterCard Products	1	174155	516.671	89980782.24
7	07Visa Classic	1	52733	513.320	27068908.04
8	08Visa Platinum	1	18221	517.667	9432408.49
9	09Visa Gold	1	14665	509.719	7475032.87
10	10All Visa Products	1	85619	513.628	43976349.40
11	11All Visa & MasterCard Products	1	259774	515.668	133957131.64
12		0	259774	515.668	133957131.64

40

A Work-Around for Version 8

```
data data8(drop=cardtype);
  set data7;
  card = substr(cardtype,3);
run;

proc print data=data8 split='_';
  id card;
  title 'Result of Work Around';
run;
```

41

A Work-Around for Version 8

Result of Work Around

card	TYPE	FREQ	Charge Amounts (\$)	
			Mean	Sum
American Express	1	0	.	.
Discover Card	1	0	.	.
MasterCard Classic	1	52798	515.302	27206890.11
MasterCard Gold	1	86611	515.054	44609348.78
MasterCard Platinum	1	34746	522.781	18164543.35
All MasterCard Products	1	174155	516.671	89980782.24
Visa Classic	1	52733	513.320	27068908.04
Visa Platinum	1	18221	517.667	9432408.49
Visa Gold	1	14665	509.719	7475032.87
All Visa Products	1	85619	513.628	43976349.40
All Visa & MasterCard Products	1	259774	515.668	133957131.64
	0	259774	515.668	133957131.64

42

Using the Same Value More Than Once in a Multilabel Format

```

proc format;
value $multi4f (multilabel)
  'VC' = '07Visa Classic'
  'VP' = '08Visa Platinum'
  'VG' = '09Visa Gold'
  'VP','VG','VC' = '10All Visa Products'
  'MC' = '03MasterCard Classic'
  'MP' = '05MasterCard Platinum'
  'MG' = '04MasterCard Gold'
  'MC','MG','MP' = '06All MasterCard Products'
  'VP','VC','VG','MC','MG','MP' =
    '11All Visa & MasterCard Products'
  'AX' = '01American Express'
  'DC' = '02Discover Card'
  'AX','DC' = '12Amex and Discover'
  'VP','MP' = '13Platinum Cards'
  'VG','MG' = '14Gold Cards'
  'VC','MC' = '15Classic Cards';
run;

```

43

Using the Same Value More Than Once in a Multilabel Format

Using the Same Value Multiple Times in an MLF

Obs	cardtype	_TYPE_	_FREQ_	charge_ amount_Sum
1	01American Express	1	0	.
2	02Discover Card	1	0	.
3	03MasterCard Classic	1	52798	27206890.11
4	04MasterCard Gold	1	86611	44609348.78
5	05MasterCard Platinum	1	34746	18164543.35
6	06All MasterCard Products	1	174155	89980782.24
7	07Visa Classic	1	52733	27068908.04
8	08Visa Platinum	1	18221	9432408.49
9	09Visa Gold	1	14665	7475032.87
10	10All Visa Products	1	85619	43976349.40
11	11All Visa & MasterCard Products	1	259774	133957131.64
12	12Amex and Discover	1	0	.
13	13Platinum Cards	1	52967	27596951.84
14	14Gold Cards	1	101276	52084381.65
15	15Classic Cards	1	105531	54275798.15
16		0	259774	133957131.64

44

Summary and Conclusions

- MLFs add a lot of power to SAS analysis and reporting capabilities in PROCs
 - MEANS
 - SUMMARY
 - TABULATE
- Work is underway in the BASE SAS R&D Group to overcome current limitations arising from having the NOTSORTED option disabled when the MULTILABEL option is also specified in a PROC FORMAT task.

45

Thanks for Attending!



Questions?

Comments?

46

Thanks!

- Thanks again for attending today's meeting
 - Questions?
 - Comments?
 - www.SierraInformation.com

47