

# Changing a Static Condition to a Dynamic Data-Driven Field with SAS®

Misty Johnson

State of Wisconsin Department of Health Services  
WIILSU North 2013 Milwaukee, WI



# Outline

Function of the SAS program

Problem

Quick fix solution

Better Solution: Data-Driven Variable

Other gems in code

Conclusion

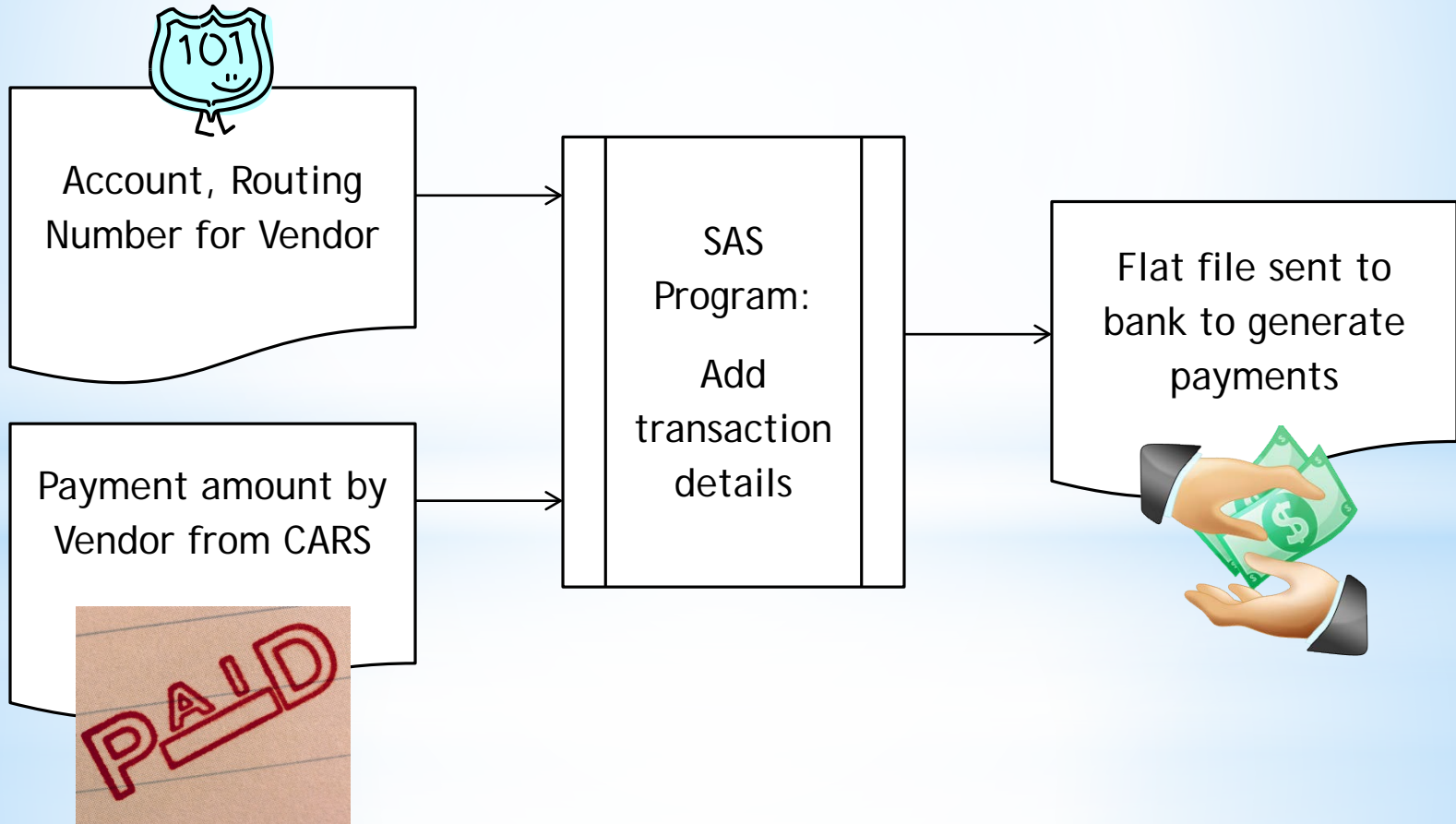


# Background

- State of Wisconsin Department of Health Services (DHS)
  - 600+ provider contracts
  - Annual contracts > \$670 million
- Community Aids Reporting System (CARS)
  - Holds contract & expense information
  - Reporting system, not a payment system
  - Monthly COBAL, SAS jobs calculate payments
    - Interface file ➡ payments

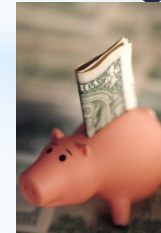


# Function of SAS Program



# Problem

- All Direct Deposit / Automatic Clearing House (ACH) payments went into checking accounts
- Vendor wished to have ACH deposited to savings account instead



# A Static Macro “Variable”

```
%LET MTRANSCD = '22';    *ASSIGN STATIC TRANSACTION CODE;  
.....(SAS code) .....  
*CONSTRUCTION OF TEXT STRING IN A LATER DATA STEP;  
DATA DETAIL3;  
...  
RECORD3 = '6' || &MTRANSCD. || BANKNUM || ACCTNUM;
```

- Static value
- Declared at top of program
  - + Easy to find if you need to change it
  - In effect for all records (vendors)





# Sidebar Discussion

Why the quotes?

```
%LET MTRANSCD = '22';    *ASSIGN STATIC TRANSACTION CODE;  
.....(SAS code) .....  
*CONSTRUCTION OF TEXT STRING IN A LATER DATA STEP;  
DATA DETAIL3;  
...  
RECORD3 = '6' || &MTRANSCD. || BANKNUM || ACCTNUM;
```



# Sidebar Discussion

- Macro variable assigned with %LET:
  - Will **always** be text
  - Numerics → BEST12. format → converted to text (with spaces)
    - Use STRIP function to remove leading/trailing blanks

```
%LET MTRANS CD = 22;
```

```
RECORD3 = '6' || STRIP(&MTRANS CD.) || BANKNUM || ACCTNUM;
```

- Why use the quotes?
  - Define as literal
  - No type conversion → no spaces!

```
%LET MTRANS CD = '22';
```

```
RECORD3 = '6' || &MTRANS CD. || BANKNUM || ACCTNUM;
```

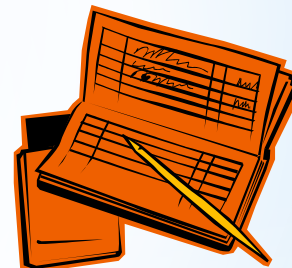




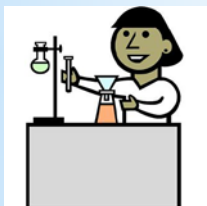
# Problem

How to best edit this macro variable?

- All ACH payments go into checking accounts



- One vendor wants savings account deposit, could be more in the future...



# Quick Fix with SAS

```
%LET SAVEACCT = ('123546', '555888'); *VENDORS WANTING SAVINGS ACCT DEPOSIT;  
%LET MTRANS SV = '32'; *TRANSACTION CODE FOR SAVINGS ACCT;  
%LET MTRANS CK = '22'; *TRANSACTION CODE FOR CHECKING ACCT;  
.....(SAS code) .....  
IF VENDNO IN &SAVEACCT. THEN  
    RECORD3 = '6' || &MTRANS SV. || BANKNUM || ACCTNUM;  
ELSE    RECORD3 = '6' || &MTRANS CK. || BANKNUM || ACCTNUM;
```

- Define “savings account vendors” as a macro variable
- Define various transaction values as macro variables
  - + No change to input data
  - Process change
    - new saving account vendor requires set-up in SAS
  - Requires a SAS programmer



# What Data do we Collect?

DEPARTMENT OF ADMINISTRATION  
EXECUTIVE BUDGET & FINANCE  
DOA-6622 (R06/2010)

## State of Wisconsin Electronic Funds Transfer Enrollment Application

Please fax the completed form to:

State Agency Name:	
Contact Name	
FAX No.	
Phone No.	

Type of Enrollment ☐ New ☐ Change Effective Date:

Vendor Information				Financial Institution Information		
Federal Identification Number				Depository Name		
Vendor's Name				Street Address		
Street Address				City		State Zip
City		State	Zip	Contact Person		
Contact Person				Phone		
Email Address				Fax		
Phone				Bank Routing Number/ABA (9-Digits)		
Fax				Account Number		
				Type of Account <input type="checkbox"/> Checking <input type="checkbox"/> Savings		
				<input type="checkbox"/> Check if the entire amount of the ACH payment is ultimately deposited to a financial institution outside the U.S.		



# Better Idea: Data Driven Variable

- Why collect data if you aren't going to use it?
- New variable in tab-delimited file (Excel<sup>®</sup>)

	A	B	C	D	E	F	G	H	I	J	K
1	OBS	IGIP	Muni_Code	Agncy_No	Vendor_Name	Last	First	Phone	Bank_Routing_No	Bank_Account_No	Trans_Code
2	2	1	01000	1	TREAS ADAMS CO	Smith	Mary	555-5554	123456789	456789	22
3	5	0	10000	10	TREAS CLARK CO	Jones	Mike	555-6666	99902111	100200	32
4	284	0	92003	100537	Mom's Child Care	Cleaver	June	555-7777	010101010	2223334444	22
5	285	0	59371	100552	Pleasantville School Distr	Bell	Lucille	555-9999	075075075	2202202200	22

22 = checking account  
32 = savings account

All names, contacts, routing numbers & account numbers shown above are fictitious and used for example only.



New

# Impact Upon Process Change



Account, Routing  
Number for Vendor

SAS  
Program:

Add  
transaction  
details

Flat file sent to  
bank to generate  
payments

Payment amount by  
Vendor from CARS

PAID





# Impact Upon Process Change

- Minor change to input data file
  - + Follows form
- Minor process change
  - + "Type of Account" now entered
  - + No SAS programmer required

	A	B	C	D	E	F	G	H	I	J	K
1	OBS	IGIP	Muni_Code	Agncy_No	Vendor_Name	Last	First	Phone	Bank_Routing_No	Bank_Account_No	Trans_Code
2	2	1	01000	1	TREAS ADAMS CO	Smith	Mary	555-5554	123456789	456789	22
3	5	0	10000	10	TREAS CLARK CO	Jones	Mike	555-6666	99902111	100200	32
4	284	0	92003	100537	Mom's Child Care	Cleaver	June	555-7777	010101010	2223334444	22
5	285	0	59371	100552	Pleasantville School Distr	Bell	Lucille	555-9999	075075075	2202202200	22

22 = checking account  
32 = savings account

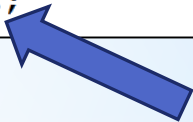




# Other Gems:

## %LET, Libname with Excel

```
%LET MCHEK      = '<FILEPATH>\EMV90318.DTA'; /* INPUT FLAT FILE FROM CARS */  
%LET MACHREC    = '<FILEPATH>\VO90318E.ACH'; /* OUTPUT INTERFACE FILE, ACH PAYMNT*/  
LIBNAME ACHFILE EXCEL "<Filepath>\VendorACH.xls" mixed=yes;
```



- Simple %LET statements for incoming/outgoing flat files
  - Payment information by vendor / interface file
- LIBNAME with EXCEL engine for Vendor file
  - Mixed option allows data types to be text & numeric in same column
    - Possible re-formatting by user?
  - Otherwise, SAS would discard rows of different type



# Other Gems:

## Literals, PUT statement

```
DATA ACHINFO
    (DROP=TRANS_CODE    RENAME=(TRANS_CODE=TRANS_CODE));
SET ACHFILE."ACH_Listing$"N;
* MAKE SURE TRANS_CODE IS A TEXT, SET TO DEFAULT OF 22 IF MISSING;
IF TRANS_CODE THEN TRANS_CODE=PUT(TRANS_CODE,2.); /*NEW VARIABLE*/
ELSE TRANS_CODE='22'; /*DEFAULT VALUE*/
RUN;
```

- Name literal required for \$ in Excel tab name
  - Sheet name may not follow SAS naming standards
- PUT statement ensures Transaction Code is text of length 2
  - Corrects possible numeric formatting in Excel by user
  - Default to Checking Account if no value
- Original Transaction Code dropped, new text version kept



# Other Gems:

## Misover, Input

```
*BRING IN PAYMENT DATA BY VENDOR FROM FLAT FILE;  
DATA CHECKS;  
INFILE &MCHEK. MISOVER;          /* EMVXXXXX.DTA - FLAT FILE FROM CARS */  
INPUT @ 7 MUNICD $ 5.  
      @ 22 PAYEE $ 22.  
      @ 54 AMOUNT COMMA13.0;  
RUN;
```

- Flat file of payments by vendor
- MISOVER option
  - Tells SAS to skip over any missing values at end of row
- INPUT statement
  - Tells SAS exactly where and formatting of raw data



# Use of Data-Driven Variable

```
DATA DETAIL3;                                /* MERGE PAYMENT FILE AND ACH TABLE BY MUNI CODE */
MERGE CHECKS ACHINFO;
BY MUNICD;
*RECORD3 = '6' || &MTRANSCD. || BANKNUM || ACCTNUM;      /* OLD CODE WITH STATIC VAR */
RECORD3 = '6' || TRANS_CODE || BANKNUM || ACCTNUM;      /* NEW CODE WITH DYNAMIC VAR */
/*... (CREATE OTHER VARIABLES NEEDED FOR INTERFACE FILE )... */
FILE &MACHREC. LS=125;                          /* OUTPUT INTERFACE FILE */
PUT
@ 3 RECORD3 $ 29.
/*... (OTHER VARIABLES NEEDED FOR INTERFACE FILE )... */
RUN;
```

- Merge payment, vendor info
- New line with dynamic data-driven variable
- Output written with FILE statement
  - Line-Size option for later upload to z/OS environment
- PUT statement writes variables in prescribed order, format



# Conclusions

- Change is inevitable!
- Write SAS programs that adapt to change
  - “Important” static values → macro variable
  - May need to find and change later
- Edit SAS programs with *minimal* process change
  - Always keep user in mind
  - Quick fix in SAS may not be the best option
- Allow the data to speak for itself



# Thank You!

Misty Johnson

State of WI Department of Health Services

[misty.johnson@wi.gov](mailto:misty.johnson@wi.gov)

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

