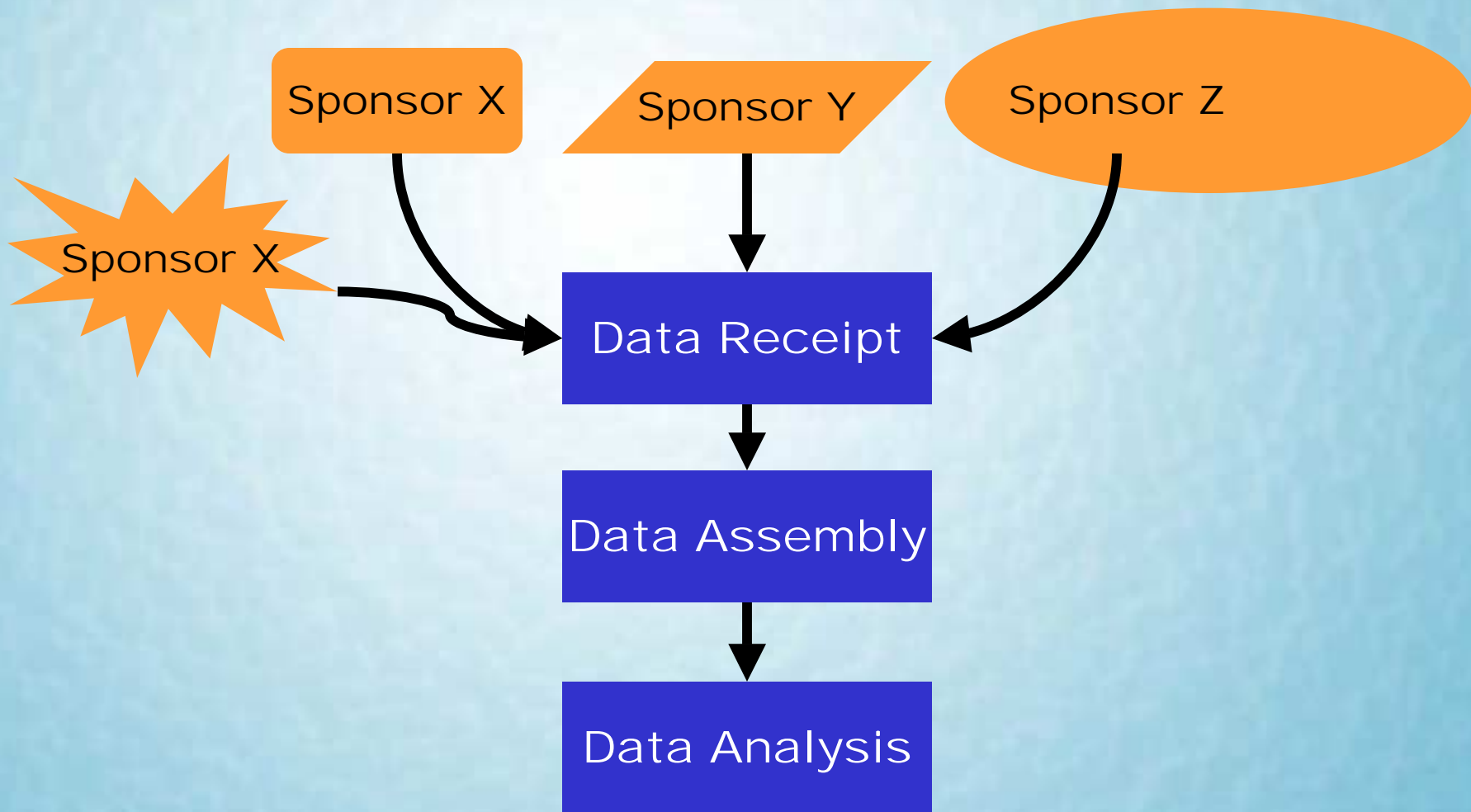

Rapid Assessment of Data Set Structures

Phil Vecchione, BS
Cognigen Corporation

Receiving Data at Cognigen



Common Questions When Receiving New Data

- What variables are located in multiple datasets?
- Have variables been added or deleted since the previous transmission?
- Does a specific variable exist?
- How complete is the data from a given dataset?

COGNIGEN

How Do We Combat This?



The VAR Family

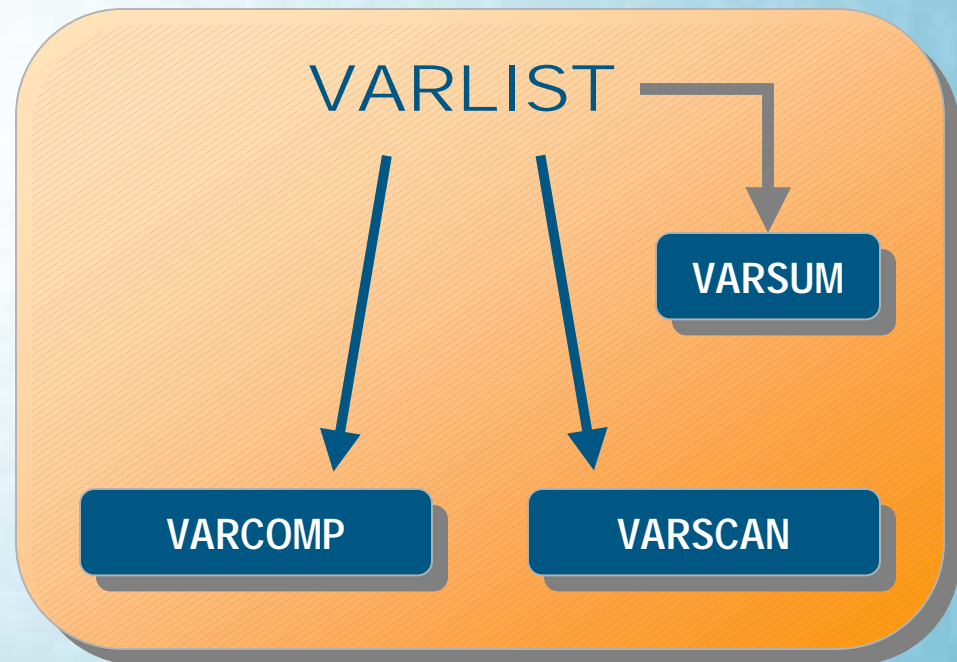
- A collection of macros that are designed to answer these issues
- All are derived from a similar program structure
- **The Crux: proc contents OUT = option**
- Used to create a dataset of variable names from all the datasets in a given library
- The dataset of variable names can be searched, summarized, analyzed

Why Not Use Data Dictionary Tables?

- At the time, I was not aware of them
- The Vcolumns table would work for creating a variable list for a given library
- Vcolumns only contains list of variable info
- Other DDT's have other info
- Proc Contents pulls this info together into one dataset
 - ◆ “One Stop Shopping”
- Easily expand functionality by using other Proc Contents variables rather than creating and merging other datasets from other DDT's

The Members of the VAR Family

- **Varlist** – Creates a list of all the variables from all the datasets in a library
- **Varcomp** – Compares the variable lists of two libraries and notes differences
- **Varscan** – Searches the variable list for a given keyword
- **Varsum** – Produces a summary table that shows the completeness of the data.



Meet the VAR Family

- Macro Call
- Function
- Use
- Mechanism
- Output

VARLIST

`%varlist(whatlib=libref, wherelib=libref, print=(0 | Null))`

- The parent macro of the VAR family
- **Function** – Creates a listing of variable names within a given library, and lists cases where a variable exists in more than one dataset
- **Use** – To map new libraries of data

```
Complete listing of Variables from:
Work.longev

MEMNAME  NAME      TYPE LENGTH LABEL          FORMAT
-----  -
BLODSAMP DAY       1      8   STUDY DAY
DRUGADMN DAY       1      8   STUDY DAY
EXAM     DAY       1      8   DAY
MEAL     DAY       1      8   STUDY DAY
QUESTION DAY       1      8   STUDY DAY
VITALS   DAY       1      8   DAY
FORMATS DEFAULT  1      5   DEFAULT
EXAM     DESCR1    2     40   DESC1
EXAM     DESCR2    2     40   DESC2
DEMOG    DOB       1      8   DATE OF BIRTH DATE

Variable Name:  BIRTHD
Label:  Date of Birth
-----
Data Sets:  DEMOG
            SUMMARY
```

VARCOMP

`%varcomp (newlib=libref, oldlib=libref)`

- **Function** – Compares variable lists from two libraries to determine if any variables have been added or removed
- **Use** – When an updated library is sent
- Calls `varlist (print=0)` to create list

```
VarComp Summary Report
-----

Number of Variables that are the same:  1128

Number of Variables that are new:       9

Number of Variables that are lost:      2
-----

                Variables Not In Previous Data Set
                (New Variables)

OBS      MEMNAME      NAME      LABEL

   1     ADE          DRUG
   2     ADE          PAGEDONE  PAGE WAS DONE
   3     DOSTUDY      DRUG
   4     EOS          DRUG
   5     MED_HX       DRUG
   6     OTHMED       DRUG
   7     SMEDACC      DRUG
   8     VITAL        DRUG
   9     VITAL        PAGENO    PAGE NUMBER

                        N = 9

                Variables In Previous Data Set but Not In New
                Data Set
                (Missing Variables)

OBS      MEMNAME      NAME      LABEL

   1     ADE          PTINIT    PATIENT INITIALS
   2     OTHMED       PTINIT    PATIENT INITIALS

                        N = 2
```

VARSCAN

`%varscan (whatlib=libref, keyword=text, field=[N,L,B])`

- **Function** – Searches a variable list for variable names and label names that contain a given keyword
- **Use** – When data first arrives to identify tables that have variables with common keywords (dose, conc, etc.)
- Calls `varlist (print=0)` to create list

```
B: Variable Name Search for Labels and Variables
      that sound like: dose
      From Library: work

Obs   MEMNAME   NAME   LABEL
1     CONMED    DOSE   DOSE
2     DRUGADMN  DOSE   COMPOUND DOSAGE

                        N = 2

B: Variable Name Search for Labels and Variables
      that contain: dose
      From Library: work

Obs   MEMNAME   NAME   LABEL
1     CONMED    DOSE   DOSE
2     DRUGADMN  DOSE   COMPOUND DOSAGE
3     CONMED    UNITS  DOSE UNITS

                        N = 3
```

VARSUM

`%varsum (whatlib=libref, dset=data set)`

- The workhorse of the VAR family
- **Function** – Creates a summary table for a given dataset, listing the variable names, labels, type, and the number of null and non-null values
- **Use** – On any table to determine if any key data is missing
- Cousin to Varlist
 - ◆ Does not call varlist, but has code derived from varlist

Variable Listing for data set: lib1.conc

Num Var	Label	Type	N	MISS	
1	PROT	Protocol	Character	884	0
2	CENTER	Center #	Character	884	0
3	SUBJECT	Subject #	Character	884	0
4	SAMPLE	SampleType	Character	884	0
5	DRDTRAW	Draw Date	Character	884	0
6	DRAWDT	Draw Date (SAS)	Numeric	884	0
7	VISIT	Visit Desc	Character	884	0
8	ANALYT	Analyte	Character	884	0
9	LAB	Analytical Lab	Character	884	0
10	EXTID	External ID #	Character	884	0
11	CONC	Concen.	Character	873	11
12	UNITS	Units	Character	884	0
13	COMENT	Lab Comments	Character	17	867

VAR Family: In the Field

- Newly arrived library of data: no Paperwork
 - ◆ Varlist
 - ◆ Varscan (keywords= conc, dose, age, sex)
- Updated library of data
 - ◆ Varcomp
 - ◆ Varsum on important tables (i.e., demog)
- Received Concentration File
 - ◆ Varsum to look at completeness

We're Expecting A New Arrival: VARLEN

- Based on VARLIST (print=0)
- This macro checks a library of data for variable names greater than 8 characters, and variable labels greater than 40 characters
- Used to check V8 datasets for V6 compatibility



Conclusions

- The VAR family gives us the ability to perform some automated checks on incoming data, and catch potential problems before data assembly occurs
 - ◆ It allows data assembly to be performed quicker
 - ◆ More time for data analysis
- VAR family makes it easy to create new family members by calling other members of the family in new macros



COGNIGEN

Thank You