

# **Efficiencies with Large Datasets**

**Vancouver  
SAS Users Group**

**Peter Eberhardt  
Fernwood Consulting Group  
Inc.**

# Efficiencies with Large Datasets

## Agenda

- Making large datasets smaller
- Sorting Issues
- Matching Issues
- General Programming Issues

## What is large?

- Short but Wide
- Tall but Thin
- Tall and Wide
  
- Anything that stretches your resources

# Efficiencies with Large Datasets

## Why Worry?

- Limited Resources
  - Space
    - Disk space
    - memory
  - Time
    - CPU
    - ‘window of opportunity’
    - YOURS



What? Me worry?

## Efficiency

- Main Entry: **ef·fi·cien·cy**  
Pronunciation: i-'fi-sh&n-sE  
Function: *noun*  
Inflected Form(s): *plural -cies*  
**1** : the quality or degree of being efficient  
**2 a** : efficient operation **b** (1) : effective operation as measured by a comparison of production with cost (as in energy, time, and money) (2) : **the ratio of the useful energy delivered by a dynamic system to the energy supplied to it**

## Efficient

- Main Entry: **ef·fi·cient**  
Pronunciation: i-'fi-sh&nt  
Function: *adjective*  
Etymology: Middle English, from Middle French or Latin; Middle French, from Latin *efficient-*, *efficiens*, from present participle of *efficere*  
**1** : being or involving the immediate agent in producing an effect <the *efficient* action of heat in changing water to steam>  
**2** : **productive of desired effects**; *especially* : productive without waste

## Making Your Dataset Smaller

- SAS COMPRESS option
  - COMPRESS=YES
    - Compresses character variables
  - COMPRESS=BINARY
    - Compresses numeric variables
  - POINTOBS=YES
    - Allows the use of POINT= in compressed data
    - May increase CPU in creating the dataset



## Making Your Dataset Smaller

- LENGTH statement
  - SAS numbers stored as 8 bytes
    - Careful about loss of data
  - Character fields stored as length of their first reference
    - 8 character default

## Making Your Dataset Smaller

- LENGTH statement - WINDOWS

Significant Digits and Largest Integer by Length for SAS Variables

Length in Bytes	Largest Integer Represented Exactly	Exponential Notation	Significant Digits Retained
3	8,192	$2^{13}$	3
4	2,097,152	$2^{21}$	6
5	536,870,912	$2^{29}$	8
6	137,438,953,472	$2^{37}$	11
7	35,184,372,088,832	$2^{45}$	13
8	9,007,199,254,740,990	$2^{53}$	15

## Making Your Dataset Smaller

- LENGTH statement
  - Affects the way numbers are stored in the dataset.  
In memory all numbers are expanded to 8 bytes

## Making Your Dataset Smaller

- KEEP/DROP
  - Limits the columns read/saved
  - Dataset option
    - set vug.large (keep=r1 r2)
    - Can be used in PROCs
  - Data step statement
    - keep r1 r2

## Making Your Dataset Smaller

- TESTING
  - Sampling
    - RANUNI()
      - Uniform random distribution between 0 and 1
      - Approximate number of records
    - OBS=
      - Limits number of records
        - Exact number of records

## Sorting

- SORT options
  - NOEQUALS
  - SORTSIZE=
    - Be careful with SORTSIZE=MAX
  - TAGSORT
  - Indexing
    - Data Step
    - SQL
  - Don't sort
    - Sortedby dataset option

## Matching

- DATA Step Merge
  - Sorted data
- PROC SQL
  - No need to sort

## Matching

- **FORMAT Lookup**
  - Create a SAS format
  - Apply the format in a DATA step

# Efficiencies with Large Datasets

## Matching

- HASH Object
  - v9

## General Programming Issues

- Avoid 'redundant' steps
- Avoid extra sorting
- Clean up unused datasets
- Use IF .. THEN ... ELSE
- Consider VIEWS
- Use LABELS
- IF and WHERE

# Efficiencies with Large Datasets

## Review

- Making large datasets smaller
  - compress, length, keep/drop
- Sorting Issues
  - noequals, sortsize, tagsort, index
- Matching Issues
  - match/merge, SQL, formats (HASH object)
- General Programming Issues

# Efficiencies with Large Datasets

**Peter Eberhardt**  
**[peter@fernwood.ca](mailto:peter@fernwood.ca)**

