

Alternate Blocks in the CPS-ASEC instrument

Roberto Picha U.S. Census Bureau

16th International Blaise conference, Beijing China 2015



U.S. Department of Commerce
Economics and Statistics Administration
U.S. CENSUS BUREAU
census.gov

Outline

CPS ASC History

Prototyping

Implementing

Issues/Solutions

Final thoughts



CPS History

Instrument in
production

- Using Blaise since 2007
- Blaise 4.8.1 build 1403
- Different supplement each month
 - ASEC runs for three months
- Measures employment rate
 - ASEC collects income data and benefit programs



CPS ASEC

ASEC supplement Redesign

- Research based on cognitive test
- Paper based approach
- Runs in February, March and April

New requirement

- Order of sections based on
 - Low income, Seniors, for member 15y.o.
- Person questions
 - Income Source
 - Amount questions for each income source



Alternate order of groups

- Twenty three sections
- Four groups
- Three paths

Table 1. Alternate Order of Income Source Questions

| Default | | Low Income | | Seniors | |
|---------|---|------------|--|---------|---|
| Order | | Order | | Order | |
| 1 | Unemployment and Workers Compensation | 1 | Unemployment and Workers Compensation | 2 | Disability |
| 2 | Disability | 7 | Public Assistance / TANF | 3 | Social Security |
| 3 | Social Security/SS for Children | 8 | Food Stamps (SNAP) | 4 | Supplemental Security Income (SSI) |
| 4 | Supplemental Security Income (SSI) | 9 | WIC | 5 | Veterans |
| 5 | Veterans | 10 | School Lunches | 6 | Survivors |
| 6 | Survivor Benefits | 11 | Public Housing | 13 | Pensions |
| 7 | Public Assistance / TANF | 12 | Energy Assistance | 14 | Annuities |
| 8 | Food Stamps (SNAP) | 2 | Disability | 15 | Retirement Accounts – Withdrawals or distributions |
| 9 | WIC- no amount collection | 3 | Social Security | 16 | Other Income Earning Assets (outside of retirement) |
| 10 | School Lunches- no amount collection | 4 | Supplemental Security Income (SSI) | 17 | Property Income |
| 11 | Public Housing- no amount collection | 5 | Veterans | 1 | Unemployment and Workers Compensation |
| 12 | Energy Assistance | 6 | Survivor Benefits | 7 | Public Assistance / TANF |
| 13 | Pensions | 13 | Pensions | 8 | Food Stamps (SNAP) |
| 14 | Annuities | 14 | Annuities | 9 | WIC |
| 15 | Retirement Accounts (within) – Withdrawals or distributions | 15 | Retirement Accounts – Withdrawals or distributions | 10 | School Lunches |
| 16 | Other Income Earning Assets (outside of retirement) | 16 | Assets (outside of retirement) | 11 | Public Housing |
| 17 | Property Income | 17 | Property Income | 12 | Energy Assistance |
| 18 | Education Assistance | 18 | Education Assistance | 18 | Education Assistance |
| 19 | Child Support | 19 | Child Support | 19 | Child Support |
| 20 | Alimony | 20 | Alimony | 20 | Alimony |
| 21 | Financial Assistance from friends or relatives | 21 | Financial Assistance from friends or relatives | 21 | Financial Assistance from friends or relatives |
| 22 | Other Income I (as in ASEC now) | 22 | Other Income I (as in ASEC now) | 22 | Other Income I (as in ASEC now) |
| 23 | Other Income II (as in ASEC now) | 23 | Other Income II (as in ASEC now) | 23 | Other Income II (as in ASEC now) |

Prototyping

Two prototypes were conducted as proof of concept

- Prototype to show the routing of alternate questions.
- Expanding the prototype to a bit larger scale.



Prototyping simple datamodel

DATAMODEL Inst "Routing questions"

FIELDS

B, C, D, E: STRING

PATH : (D "default", L "low Income", S "Seniors")

RULES

PATH

IF PATH = D **THEN**

B

C

D

ELSEIF PATH = L **THEN**

C

B

D

ELSE

B

D

C

ENDIF

E

 [Error] INST2.bla(135,6): Field already displayed, or display order conflict



Prototyping complex datamodel

DATAMODEL Inst
FIELDS

Path : (Default "B,C,D,E", LowIncome "C,B,D,E", Seniors "B,D,C,E")

BLOCK BlkB ; **BLOCK** BlkC ; **BLOCK** BlkD ; **BLOCK** BlkE

BLOCK BOrderOne { default order }

FIELDS B : BlkB; C : BlkC; D : BlkD; E : BlkE
ENDBLOCK

BLOCK BOrderTwo { low Income order }

FIELDS C : BlkC; B : BlkB; D : BlkD; E : BlkE
ENDBLOCK

BLOCK BOrderThree { Senior order }

FIELDS B : BlkB; D : BlkD; C : BlkC; E : BlkE
ENDBLOCK

BLOCK BMaster {master copy of the answers (no matter which order)}

Fields B : BlkB; C : BlkC; D : BlkD; E : BlkE
ENDBLOCK

AUXFIELDS

OrderOne : BOrderOne

OrderTwo : BOrderTwo

OrderThree : BOrderThree

DoOnce : 0..1

FIELDS

Master : BMaster



Prototyping complex (cont'd)

RULES

```
Master.KEEP {keep the master set of answers}  
path  
DoOnce.KEEP
```

```
IF DoOnce = EMPTY THEN  
  IF path.ord = 1 THEN  
    OrderOne.B := Master.B  
    OrderOne.C := Master.C  
    OrderOne.D := Master.D  
    OrderOne.E := Master.E  
  ELSEIF path.ord = 2 THEN  
    OrderTwo.C := Master.C  
    OrderTwo.B := Master.B  
    OrderTwo.D := Master.D  
    OrderTwo.E := Master.E  
  ELSEIF path.ord = 3 THEN  
    OrderThree.B := Master.B  
    OrderThree.D := Master.D  
    OrderThree.C := Master.C  
    OrderThree.E := Master.E  
  ENDIF  
  DoOnce := 1  
ENDIF
```

```
IF path.ord = 1 THEN  
  OrderOne  
  Master.B := OrderOne.B A  
  Master.C := OrderOne.C  
  Master.D := OrderOne.D  
  Master.E := OrderOne.E  
ELSEIF path.ord = 2 THEN  
  OrderTwo  
  Master.C := OrderTwo.C  
  Master.B := OrderTwo.B  
  Master.D := OrderTwo.D  
  Master.E := OrderTwo.E  
ELSEIF path.ord = 3 THEN  
  OrderThree  
  Master.B := OrderThree.B  
  Master.D := OrderThree.D  
  Master.C := OrderThree.C  
  Master.E := OrderThree.E  
ENDIF
```



Implementing

Breaking Sections into small blocks to accommodate the order of blocks

Some dependencies were disabled

This led to additional task not anticipated

Modifying parameterization to blocks

Rearranging parameters

Read information across sections, generating GP's for simplicity

Layouts



Issues during implementation

During system test

- Missing data
 - The execution of rules caused some missing data
 - Flag variables
 - Usually array data
- Data loosing when exiting interview early



Solutions during implementation

System test and how to save data

- Applying KEEP statements
 - Running out of time
- Converting Auxfields blocks to Fields
 - This eventually caused some concerns
 - Incremented size of metafile up to five times



Issues during production

Master Control

- Running Manipula Scripts
 - Recycling cases
 - Re-interview writing SCIF data

Data Output

- Duplicate data
 - House level
 - Person Level



Solutions during production

Master control Access Violation

- Using a greater version of Manipula
 - Blaise 4.8.4 build 1861

Data Output

- Duplicate data existed due to the conversion of auxfields to field blocks
 - Fourth instance block for person level
 - 49th instance block person level



Data dup linking HH

- To determine which line was the correct line of data to be utilized was as follows.
 - Looking in the blkASEC block (file A.81) to determine the pointer value for BINCSOR_AMT. In the following example is at position 36-40.

| inst.A81 | | x | | | | | | | | | | | | | |
|----------|----------|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 10 20 30 40 50 60 70 80 90 | | | | | | | | | | | | | |
| 1 | OW000001 | 1 | 1 | | 1 | 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | OW000002 | 1 | 1 | 1 | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 3 | | | | | | | | | | | | | | | |

- This value matches up to the instance number in block blkincSor_Amt in file B.83

| inst.A81 | | inst.B83 | | | | | | | | | | | | x | | | | | | | | | | | | | | | |
|----------|----------|---------------------|---|---|---|---|---|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 10 20 30 40 50 60 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | OW000001 | 1 | 4 | 4 | 4 | 4 | 4 | 41 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | |
| 2 | OW000002 | 1 | 4 | 4 | 4 | 4 | 4 | 41 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- In this example, data was entered in the Veteran's blocks which are located in the Source A block, so the data output team needed to locate the Pointer value for A_Sor.

| inst.A81 | | inst.B83 | | | | | | | | | | | | x | | | | | | | | | | | | | | | | |
|----------|----------|------------------------|---|---|---|---|---|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 10 20 30 40 50 60 70 | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | OW000001 | 1 | 4 | 4 | 4 | 4 | 4 | 41 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | |
| 2 | OW000002 | 1 | 4 | 4 | 4 | 4 | 4 | 41 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Data dup linking HH (cont'd)

- Then data was matched to the instance number in block blkInc_Source_A.
- Example below shows two lines per case.
- The correct lines in this case are where the instance number is 4.

| | inst.A81 | inst.B83 | inst.A93 | x |
|---|----------|----------|----------|---|
| 1 | OW000001 | 3 | 3 | 3 |
| 2 | OW000001 | 4 | 4 | 4 |
| 3 | OW000002 | 1 | 1 | 1 |
| 4 | OW000002 | 4 | 4 | 4 |

- From here, the pointer number for BVET_PMT matched the instance number for blkVet_Pmt

| | inst.A81 | inst.B83 | inst.A93 | x |
|---|----------|----------|----------|---|
| 1 | OW000001 | 3 | 3 | 3 |
| 2 | OW000001 | 4 | 4 | 4 |
| 3 | OW000002 | 1 | 1 | 1 |
| 4 | OW000002 | 4 | 4 | 4 |

| | inst.A81 | inst.B83 | inst.A93 | inst.B04 | x |
|---|----------|----------|----------|----------|---|
| 1 | OW000001 | 31 1 2 | | 33 34 | |
| 2 | OW000001 | 41 1 2 | | 49 50 | |
| 3 | OW000002 | 11 | | 1 | |
| 4 | OW000002 | 41 | | 49 | |

Data dup linking Person.

- The same process was applied to get the pointer values for BVET_PMT_PERSON1
- Matching instance number for blkVet_Pmt_Person between file .B04 and file .B05

| | 0 | 10 | 20 | 30 | 40 | 50 | 60 |
|---|----------|----|----|----|----|----|----|
| 1 | OW000001 | 31 | 1 | 2 | | 33 | 34 |
| 2 | OW000001 | 41 | 1 | 2 | | 49 | 50 |
| 3 | OW000002 | 11 | | | | 1 | |
| 4 | OW000002 | 41 | | | | 49 | |
| 5 | | | | | | | |

| | 0 | 10 | 20 | 30 | 40 | 50 | 60 |
|---|----------|----|-------------------------------|----|----|----|----|
| 1 | OW000001 | 33 | 1123452123450199001UA61000101 | | | | ☐ |
| 2 | OW000001 | 34 | 21 11 0199001UA61000101 | | | | ☐ |
| 3 | OW000001 | 49 | 1123452123450199001UA61000101 | | | | ☐ |
| 4 | OW000001 | 50 | 21 11 0199001UA61000101 | | | | ☐ |
| 5 | OW000002 | 1 | 115 21 50199002UA61000101 | | | | ☐ |
| 6 | OW000002 | 49 | 115 21 50199002UA61000101 | | | | ☐ |
| 7 | | | | | | | |



Final thoughts

- Implementing significant changes to Production instruments can be really challenging
- Larger metafile caused some issues
- Returning back to Auxfields
 - Mapping the block with the use of keeps
 - Use of Manipula to synchronize the load or store of data
- Parallel production test
 - CPS ASEC using redesign vs old design
 - Data comparison about approach



Questions Answers

Roberto.v.picha@census.gov



U.S. Department of Commerce
Economics and Statistics Administration
U.S. CENSUS BUREAU
census.gov