Coping with people who just won't stay put: The Use of Blaise in Longitudinal Panel Surveys

Colin Setchfield, Office for National Statistics, UK

1. Introduction

The UK's Office for National Statistics (ONS) has conducted the General Household Survey (GHS) since the survey's inception in 1971. It is a multi-purpose continuous national survey of people living in private households, comprising a household section (where all questions are addressed to a household reference person) and individual sections (where questions are asked individually of all resident adults in the household aged 16 and over).

In 2005, in order to meet new requirements for producing comparable data for the European Union's Survey of Income and Living Conditions (EU-SILC), these were incorporated into GHS[1]. The result was the implementation of a rotating four-year panel design, with first follow-up interviews in 2006. The sample would be interviewed in each of the three subsequent years after it was first included; each of the four years is referred to as a 'Wave'. (In order to replenish the sample, a quarter of the sample is replaced each year.)

Other panel surveys conducted by ONS – such as its Labour Force Survey (LFS) – always returned to the original sampled addresses at each subsequent wave, and interviewed the residents there: irrespective of whether they were the interviewees at the previous occasion. Contrariwise, those interviewed at Wave 1 on GHS were defined as original sampled members. It was them, and not necessarily the address at Wave 1, in which successive waves would be interested.

These original sample members (OSMs - the household members at Wave 1) remain part of the sample for the remaining waves, unless they die, move into an institution or move abroad. If they move elsewhere within Great Britain, and are resident there at a later wave, they are interviewed at their new address.

At successive waves, any other people who move in with the original sampled members from Wave 1 (irrespective of address) also become eligible for interview. These later additions only remain eligible so long as they reside at an address where at least one original sampled member from Wave 1 also resides.

A method of feeding data forward across to the next wave had been previously established at ONS, and with minor adaptation this was applied to the GHS to become the standard ONS approach for rotation[2]. It was, however, this tracking of people's movements and the subdividing of households which had split that became the main challenge in developing the Blaise datamodel.

---

[1] The merged survey now uses the acronym GSL though in this paper the more familiar GHS has been retained throughout.

[2] "Rotation" and "feeding forward" are interchangeable terms with ONS. They both refer to the transfer of data collected at one wave into the next subsequent wave's datamodel.

This paper details both the ONS method of rotating data, and also the filtering method to change the household array across waves: used in order to cater for the removal and addition of household members and forcing removed members into new household cases for interviewing or processing. Its conclusion will review the impact of such complicated code, particularly in the light of organisational changes experienced within ONS.

2. Feeding forward data: establishing a standard ONS approach

2.1 Background

Feeding data forward from one wave to another was already used within ONS on its Labour Force Survey (LFS) prior to the redevelopment of the GHS. The LFS method was relatively straightforward. The data from the previous wave were read into the individual cases when created, prior to their release into the Field. Consequently, they were automatically populated in the Fieldpane within the interview grid even before interviewing began. This method predated ONS's use of Manipula, and was dependent on previously-used Clipper programs. It was therefore not a basis for feeding data forward on GHS.

2.2 Use of External database

A method for rotation using Blaise, was introduced by Tim Burrell for the introduction of the Annual Population Survey (APS) in 2004, having previously been used on a number of ad hoc surveys. Rather than pre-filling the questionnaire with data from the previous wave; the data were provided as an External database, allowing rotation to be processed within the datamodel itself.

All data that needed to be fed forward to the following wave were to be placed in a single file. Feeling that, in some situations, there may be capacity issues if the resulting file proved too large, it was felt that it would, in these instances, be appropriate to subdivide the file and use more than one External file. On the APS, data were divided into two files: one holding household-level information and another holding person-level information, though this was mainly to meet the processing needs of the LFS.

On GHS a single External file was provided combining both household-, and person-, level information for the whole sample. Nevertheless, in order to optimise the External file, only those fields whose data were to be fed forward were included.

Originally, consideration had been given to an option to "pick and mix" individuals from different addresses, should someone move from one sampled address to another: hence every record in the sample was provided within the External file. Writing Blaise code to facilitate this merging of data was deferred, to allow time to investigate frequencies of this happening. With no such cases observed, this option was dropped, requiring interviewers simply to re-enter the data of a sampled person who should happen to move in with other sampled members. As a result, the External database needs only to include the one related record from the sample rather than the full record. Though this change has yet to be implemented on the GHS, a single-record External file is being used for ONS's next new panel – the Household Asset Survey.

On GHS, with waves a year apart from each other, a separate Keeping-in-Touch Exercise (KITE) was provided between waves in order to update contact information when

applicable. One option would have been to use the KITE data as a second External file at the subsequent wave in order to supplement or update the data from the previous wave. Instead it was decided that GHS data would be fed forward into the KITE, so that at the subsequent wave only one External (the KITE database) was required, thereby reducing the number of dependencies.

In preference to reading the External file at each block in the questionnaire, a separate holding block (QRotate) was provided, which read the External database of the previous data and to which the data were copied across. The External file was read only once; from this holding block, the fed-forward data could be manipulated and assigned to other blocks within the questionnaire, without further reference to the External.

2.3 Methods of feeding forward data

Three methods of feeding forward data were available.

- Automatic feeding of data into Fieldpane
- Showing fed-forward data in Infopane as an interviewer instruction
- Showing fed-forward data in Infopane as part of the read-out text of the question

Dependent interviewing – using previously-collected data either to remind the respondent of his or her previous answers or to probe for changes since then – is widely used in panel surveys. It helps to reduce respondent burden, as well as having beneficial effects on interview length, and reports have shown evidence for improved data quality.

<u>Automatic feeding of data into Fieldpane</u>

This was the method that had been utilised on the LFS. As that survey grew over the years, this method helped to contain an exponential growth in questionnaire length at subsequent waves. Such was the extent that this continues to be the method of choice for the LFS even when it becomes part of the Integrated Household Survey in 2008.

It was felt, however, that requiring interviewers to re-enter answers was more methodologically sound (Jäckle (2006) – see below). Therefore, apart from at a few fields in GHS, this method is restricted in ONS to LFS alone and is not seen as the standard approach.

One of the dangers of placing data from the previous wave in the Fieldpane was interviewers unintentionally passing over whole sections of questions, simply by accidentally pressing one of the navigation buttons on the laptop (such as <Page Down>).

As mentioned above, this method was employed in a few instances on the GHS. These were (1) where data were unlikely to have changed apart from in the most exceptional circumstances, or (2) where a filtering question could be used to check whether the previous situation had changed. This first category comprised the questions on sex, date of birth, and also household accommodation, though allowing an overwrite option in case the details had changed or had been recorded incorrectly before. The second category comprised questions on occupation and qualifications. For these, a filter question was introduced at subsequent waves, because of the level of information associated with them.

The filter question provided the full information on occupation or qualifications held, asking whether the person held exactly the same job as before or whether she had obtained any additional qualifications since last wave. Depending on the respondents' answer at the filter question, the ensuing questions on the topic were either automatically filled and passed over or the interviewer was routed back through them.

Figure 1: Method and Blaise Code for feeding data into Fieldpane

Method:

In the FIELD section,

- add the Empty attribute to the relevant field(s) (to allow field to be shown as blank on deleting previous data before entering new data)

In the RULES section,

- name the field
- add a condition to apply only if that Field is empty
- (for Fields related to individuals) add a secondary condition to apply only if the person is not a new household member
- add a final condition to apply only if data exists within the holding block (QROTATE)
- instruct Blaise to assign the value of the fed-forward data into the live Field

Example Code:

```
Sex
IF Sex <> RESPONSE THEN
   IF QNames.QBNames[i].AxNewPer=No THEN
      IF QRotate.QBRotate[i].RSex =RESPONSE THEN
         Sex:=QRotate.QBRotate[i].RSex
      ENDIF
   ENDIF
ENDIF
```

<u>Showing fed-forward data in Infopane (as question text or instruction)</u>

There were concerns that overuse of the option of automatically feeding data into the Fieldpane could lead to a deterioration in data quality. The fear was that interviewers could become too reliant on it and not check sufficiently whether the information had changed since last call.

Consideration was given to using variable text fills within the question wording, in order to reference the answer from the previous wave before asking the question itself. For example, the question, "How many bedrooms do you have?" at a subsequent wave would read, "Last time, you said you had four bedrooms; is this (still) correct?"

Within ONS, variable text fills are used sparingly. For us, it has always been important to keep Blaise code easily read and intelligible; an excessive use of text fills has been seen as counter to this. Similarly, there is a concern that an over-reliance on text fills could impact on the focus and professionalism of interviewers, encouraging them to read the questions without considering the context.

It was also felt that showing the fed-forward data as an interviewer instruction, and not read into the spoken text of the question, safeguarded against the respondent being unduly influenced by the answer that they gave at the previous wave.

Figure 2: Method and Blaise Code for displaying data in Infopane

Method:

In the AUXFIELD section,

- create an auxfield to display the variable text into which the previous data will be fed forward (it may also be necessary to create locals - LRData - for fed-forward data set up as enumerated Sets). For GHS, a single Auxfield was definied (AxRData) and was set back to empty after each use

In the RULES section,

- first set the Auxfield to blank (e.g. AxRData:='')
- if fed-forward data is set up as an enumerated Set and the Field name is not clear enough to be used within the variable text, define the text to be used for each option by using the Locals you have created
- (for Fields related to individuals) add a condition to apply only if the person is not a new household member
- add a secondary condition to apply only if data exists within the holding block (QROTATE)
- define the text to be displayed using the Auxfield (in combination with any Local if necessary)
- the toggle **@R** was been defined for use with instructions displaying fed-forward data, this formats the text in a dark-orange red.

Example Code:

```
LRData:=''
AxRData:=''
IF DVAge >=16 THEN
   IF QRotate.QBRotate[i].RMarStat = NevMarr THEN
      LRData := 'Single - Never Married <1>'
   ELSEIF QRotate.QBRotate[i].RMarStat = MarrLiv THEN
      LRData := 'Married and living with spouse <2>'
   ELSEIF QRotate.QBRotate[i].RMarStat = Separated THEN
      LRData := 'Married but separated from spouse <3>'
   ELSEIF QRotate.QBRotate[i].RMarStat = Divorced THEN
      LRData := 'Divorced <4>'
   ELSEIF QRotate.QBRotate[i].RMarStat = Widowed THEN
      LRData := 'Widowed <5>'
   ENDIF
   IF QNames.QBNames[i].AxNewPer=No THEN
      IF QRotate.QBRotate[i].RMarStat = RESPONSE THEN
         AxRData:='@RLast time Marital Status was recorded as ' +
         LRData +'@R'
      ENDIF
   ELSE
      AxRData:=''
   ENDIF
   MarStat
```

Jäckle (2006) warns against 'proactive dependent interviewing' – reminding respondents of previous answers – "because of the potential under-reporting of change over time".

Consequently, GHS adopted the alternative approach of 'reactive dependent interviewing' – using previous answers only to check information received. Data from the previous wave were displayed in the Infopane as an interviewer instruction only (not read out to the respondent): leaving the question text unaltered. This required the interviewer to re-enter the data each time. Though this would take more time to interview at the subsequent wave, it was felt that data quality was safeguarded.

3. The Filter Block: Disposing of and reallocating unwanted people

3.1 Background

The new Rotation method for feeding data forward across waves only addressed part of the longitudinal aspect of GHS. It was most suited to panel surveys where the respondents never moved from their original address or, if they did, they dropped from the sample.

GHS was different. It allowed for people to become eligible at later waves even though they had not been interviewed in previous waves. It allowed for people to retain their eligibility even if they moved address. It allowed for a single address at Wave 1 to spawn eligible households at multiple addresses across the country as people moved out of their original home into new homes with or without those they had previously lived with, and also with or without others with whom they had never previously lived.

To cope with these requirements, a Filter block (QTFilter) was developed in order that, before interviewing at the household, the current eligibility status of each member at the previous Wave could be recorded.

The Filter block, however, grew in function, also serving to (1) incorporate data on new contact details for 'Movers', (2) act as a controlling block for all multi-households created at subsequent waves, and (3) facilitate checks so that a case cannot be transmitted without all necessary additional households being opened.

3.2 Structure of Filter Block

In terms of its Fields, the structure of the Filter Block was relatively straightforward. For each person recorded at the previous interview, the following details were collected.

- Current Status, with categories defined as:
  - Resident here – eligible to be interviewed
  - Resident here – under the age of 16
  - Moved from Household 1 – Now resident locally; details known; can interview
  - Moved from Household 1 – Now resident elsewhere in Great Britain; details known; reallocate
  - Moved from Household 1 – Now resident at unknown address
  - Ineligible – Died since last call
  - Ineligible – Now in institution (for 6 months or more)

- Ineligible – Now resident abroad (for 6 months or more)
- Ineligible – Mover at GSK, new case already created; or, No OSM left (*GSK is the KITE questionnaire for GHS*)

- A Derived Variable Field recording whether the person was an OSM
- For those who have moved, two Fields: one asking whether the person moved to the same address as another person already coded, and the other to record the date a person moved
- For these movers, Fields are provided to record details of new address, telephone number and email, or these if details are not known a Field to record what action was taken by the interviewer in order
- For those who have become ineligible, two Fields: one to record the country to where a person has moved, and the other to record a date of death or committal to an institution (both only when appropriate)

The complicatedness of the block, however, lay within its routeing, particularly in terms of how it would be used as a "gate-keeper" to determine which individuals from the previous interview should continue to be interviewed using the same case and which should be assigned to new cases in order to interview them elsewhere or to code them off out of the sample.

3.3 Different category – different case

ONS interviewers and its Casebook system (for managing cases in the Field) were both used to dealing with split households. A case could be spawned in the Field to create a second, third or fourth version of it. It would retain the same Primary key elements of Area number and Address number, and would increment the remaining element – Household number – by one for each new case created. In usual circumstances, this was used to cope with multihousehold addresses; at subsequent waves, however, GHS was not usual.

After its first wave, GHS's sample was relatively fixed. Only OSMs of households at Wave 1 (and people they live with at the time of re-interview) would be eligible for interview. As such, no new multihousehold address detected at subsequent waves would be eligible, unless it comprised of at least one of the original household members.

The option of creating additional households at addresses was therefore used instead as a way of processing the people from last interview into their separate common categories. All people who continued to be resident at the same address would be included all together in one household (if there were any in this category, they would continue to be interviewed at Household 1). Those who had moved would be designated the next available household on the following basis.

1. Those who had moved locally within the same sampled area and therefore still able to be interviewed by the same interviewer. If more than one person moved into the same address, they would be interviewed as one household; if a person moved without any other OSM, they would be designated their own individual household.

2. Those who had moved outside of the sampled area, and therefore it was more effective to reallocate to another interviewer working closer to the new address. Again, those who moved together would be designated the same household.
3. Those who had moved to an unknown address.

Finally, all those who were no longer eligible for interview (those who had died, been institutionalised, or who moved abroad) would be designated a single household in order to be coded out of the sample. The same Final Outcome code applies to all household members who become ineligible and therefore separate households were not required for these.

Having categorised the previous residents into groups for interviewing or coding off, it was next necessary to determine whether any member of the group was an original household member. If none was, that group would no longer be eligible for interview.

Figure 3: Method and Blaise Code for designating individuals to separate households

Method:

In the FIELD section,

- create a field to record whether person is an OSM or living with an OSM
- create a field to record household number at which person should be processed
- create counters for number of Local Movers, Distant Movers, and Movers to unknown addresses,

In the RULES section,

- first field is simply a matter of assigning a value on the basis of the OSM field of the person or, if a mover, of the person with whom they moved
- compare Current Status field of person with the field of (if a mover) whether they moved with someone else and with the field above
- if person still eligible and resident at same address allocated to household 1
- set each mover counter to – 1
- count number of movers by category so long as they moved locally without anyone else
- on counting person, assign household on the basis it will be the sum of 1 plus the aggregate of all counters processed to that point in order

Example Code:

```
LGB:=(-1)
 {Local for counting the number of distant movers}
 LLocal:=(-1)
 {Local for counting the number of locals movers}
 LDK:=(-1)
 {Local for counting the number of movers to unknown addresses}
 FOR i:=1 TO 16 DO
    IF (QBFilter[i].CurStat=Resident OR QBFilter[i].CurStat=UnderAge) THEN
       QBFilter[i].MemHH:=1
       {Those still resident are defaulted to Household 1}
    ELSEIF (QBFilter[i].CurStat=MovedInt AND
   (QBFilter[i].MovedWth=EMPTY OR
    QBFilter[i].MovedWth=17)) THEN
       LLocal:=LLocal+1
       QBFilter[i].MemHH:=1+LLocal
       IF NResident=1 THEN
          QBFilter[i].MemHH:=QBFilter[i].MemHH+1
       ENDIF
```

Example Code (continued):

```
ELSEIF (QBFilter[i].CurStat=MovedGB AND
(QBFilter[i].MovedWth=EMPTY OR
QBFilter[i].MovedWth=17)) THEN
   LGB:=LGB+1
   QBFilter[i].MemHH:=1+LGB
   IF NResident=1 THEN
      QBFilter[i].MemHH:=QBFilter[i].MemHH+1
   ENDIF
   IF NMovedInt>0 THEN
      QBFilter[i].MemHH:=QBFilter[i].MemHH+NMovedInt
   ENDIF
ELSEIF (QBFilter[i].CurStat=MovedDK AND
(QBFilter[i].MovedWth=EMPTY OR
QBFilter[i].MovedWth=17)) THEN
   LDK:=LDK+1
   QBFilter[i].MemHH:=1+LDK
   IF NResident=1 THEN
      QBFilter[i].MemHH:=QBFilter[i].MemHH+1
   ENDIF
   IF NMovedInt>0 THEN
      QBFilter[i].MemHH:=QBFilter[i].MemHH+NMovedInt
   ENDIF
   IF NMovedGB>0 THEN
      QBFilter[i].MemHH:=QBFilter[i].MemHH+NMovedGB
   ENDIF
```

3.4 Concertinaing the household per case

The first part of the Filtering process had thus provided each household member from the previous interview with their current eligibility status and also designated a grouping or 'household' in which to process or interview that individual.

The second part of the Filtering process was then the feeding forward of data from last interview (discussed above), according to the eligibility status and the designated household. In addition to this, a block was required to list names of those persons who had joined the household since last interview. It was necessary for this filtering and rotating of data to be undertaken within the Blaise code prior to the questionnaire proper, in order that household arrays throughout the questionnaire were informed by the new household composition.

As detailed above, feeding forward data from one wave to another can be simply achieved by assigning data from an External database into the current datamodel. However, with the potential of previous residents leaving and new people joining, the assigning of data would require two local counters, so that (for example) the data of Person 5 from last interview could be credited to the same person even though she was now Person 2 in the current wave.

When reading data from the External file, the Blaise would focus only on those individuals designated to household number of the case it was processing. Those passed over in this case would be processed when a case had been spawned for the relevant household number.

Figure 4: Method and Blaise Code for only feeding forward for relevant persons within household

Method:

In RULES section,

- create a FOR..DO loop, arrayed to the maximum household size
- the Local controlling the FOR..DO loop, designates the record read from the External database
- create a second Local which will designate the record number in the current datamodel to which the data will be written; initially set this Local to zero (0)
- for the first element of the array, if the person is designated to the household number of the case, set second Local to 1, otherwise keep it at zero (0)
- for each subsequent element, if the person is designated to the household number of the case, the second Local is incremented by 1, otherwise it retains the value it previously had

Example Code:

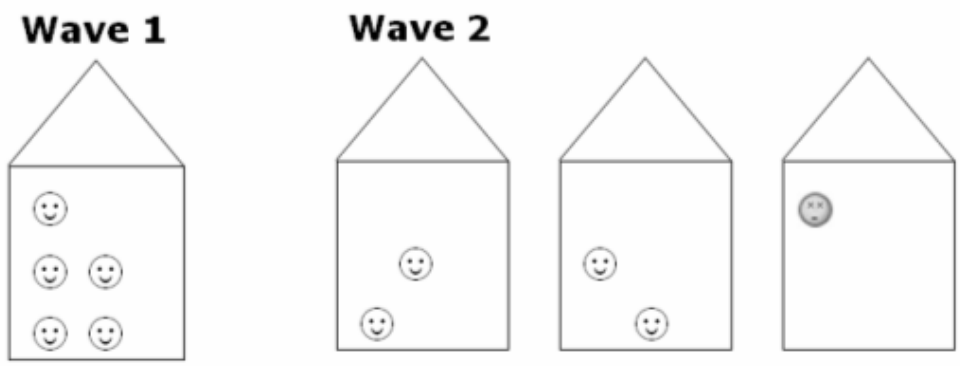
```
FOR i:=1 TO 16 DO
  IF (i= 1) THEN
    j:=0
    IF (QTFilter.QBFilter[i].MemHH=QID.HHold) THEN
      j:=1
    ELSE
      j:=0
    ENDIF
  ENDIF
  IF (i>1) THEN
    IF (QTFilter.QBFilter[i].MemHH=QID.HHold) THEN
      j:=j+1
    ELSE
      j:=j
    ENDIF
```

For example, in Figure 5 (below), at Wave 1 the sampled address had five resident eligible people who were interviewed, these are numbered 1-5 top to bottom, left to right. At the following interview (Wave 2), person 1 had died, persons 3 and 4 continued to remain resident at the original address, whereas persons 2 and 5 had moved out together to a new address. At Wave 2, the original household would now be divided for processing into the following groups or 'households'.

Household 1:	Person 1 (Previously Person 3 last Wave) – Resident Person 2 (Previously Person 4 last Wave) – Resident
Household 2:	Person 1 (Previously Person 2 last Wave) – Mover Person 2 (Previously Person 5 last Wave) – Mover
Household 3:	Person 1 (Previously Person 1 last Wave) – Ineligible

The process just described will have processed two people from last Wave (household members 3 and 4) for Household 1. The issue that remained was how to ensure interviewers spawned new households in order to process the remaining people.

Figure 5: Pictogram for use with example (above) of how movers and ineligible persons are designated to separate households



3.5 The paradox of referencing the database as an External within its own datamodel

The concern was that, having interviewed those respondents still resident at the original address and coded off the case, there were no checks to ensure that an interviewer spawned new households in order to process the remaining respondents who had moved or become ineligible. Even if they remembered to spawn new households, a method needed to (1) apply the filtering from the first household and then (2) feed forward data from the previous interview appropriate to the people of these spawned households.

Spawned households would be stored as separate records within the same database as the original case. For one of these to be able to read the filter information of Household 1, the database of those cases needed to be used as an External file within the datamodel. This is possible within Blaise, and the metadata and data would simply need to be specified within the datamodel's Uses and Externals sections, as with any other External file.

The only problem this posed was that the datamodel could not be compiled until the *.BMI metadata file existed, which of course did not exist until the datamodel was compiled. The answer was to compile the datamodel twice over. For the first compilation, the references in the Uses and Externals sections to the datamodel's own database were commented out. Having compiled the datamodel minus these references, a *.BMI metadata file then existed which could be placed in the appropriate directory in order to recompile with the references previously commented out now commented back in.

With the option of the spawned household now able to read the record for Household 1, it was possible to assign the whole of the data from the filter block Household 1 to the spawned household, allowing to same method as above to be used to process the persons designated to that 'household'.

Figure 6: Method and Blaise Code for reading and feeding forward data from Household 1's Filter block

Method:
In RULES section,

- If Household number is greater than 1 (i.e. it is a spawned household), then read the case's database for the data line for Household 1
- Read the line, and keep and assign data from Household 1 to current household

Example Code:

```
ELSEIF QID.Hhold>=2 THEN
  IF CaseData.SEARCH(QID.Area,QID.Address,1) THEN
     IF CaseData.RESULT=0 THEN
        CaseData.READ
        CurStat.KEEP
        MovedWth.KEEP
        ROHM.KEEP
        InOut.KEEP
        DateMov.KEEP
        Countres.KEEP
        Actions.KEEP
        MemHH.KEEP
        CurStat:=CaseData.QTFilter.QBFilter[i].CurStat
        MovedWth:=CaseData.QTFilter.QBFilter[i].MovedWth
        DateMov:=CaseData.QTFilter.QBFilter[i].DateMov
        Countres:=CaseData.QTFilter.QBFilter[i].Countres
        DateDth:=CaseData.QTFilter.QBFilter[i].DateDth
        PID:=CaseData.QTFilter.QBFilter[i].PID
        KnowDet:=CaseData.QTFilter.QBFilter[i].KnowDet
        Add1:=CaseData.QTFilter.QBFilter[i].Add1
        Add2:=CaseData.QTFilter.QBFilter[i].Add2
        Add3:=CaseData.QTFilter.QBFilter[i].Add3
        Add4:=CaseData.QTFilter.QBFilter[i].Add4
        Distrct:=CaseData.QTFilter.QBFilter[i].Distrct
        PstTwn:=CaseData.QTFilter.QBFilter[i].PstTwn
        PostCd:=CaseData.QTFilter.QBFilter[i].PostCd
        SurePstC:=CaseData.QTFilter.QBFilter[i].SurePstC
        TeleNumb:=CaseData.QTFilter.QBFilter[i].TeleNumb
        EmailAdd:=CaseData.QTFilter.QBFilter[i].EmailAdd
        Actions:=CaseData.QTFilter.QBFilter[i].Actions
        MemHH:=CaseData.QTFilter.QBFilter[i].MemHH
     ENDIF
  ENDIF
ENDIF
```

Additionally, this ability to read its own database allowed for greater control in the back-end Admin section of the questionnaire. Counts of household member by category – resident, mover (divided into each subcategory type), and ineligible – were computed in the filter block.

Not only did these allow for the status and outcomes of individuals and cases to be automatically computed in the Admin section of each case (original and spawned households), it allowed to check the status and outcome of related cases. This provided the option to disallow a household from being transmitted back to the office until households expected to be spawned (on the basis of the data in the Filter block) had been opened and completed.

It also allowed for messages to be displayed, against interviewers' cases within their Casebook Pending Tray, highlighting that additional households needed to be spawned and the total number expected.

4. Conclusion

Within ONS, the Blaise Mantra has always been, "Keep it simple". The Filter/Rotation method introduced on the GHS and now becoming standard for other longitudinal surveys at ONS is far from this sentiment.

Though each part of the method uses basic Blaise functionality, as a whole the complex code created problems for researcher-programmers not only in getting to grips with the processes being undertaken but (more so) in the impacts these had later in the questionnaire (particularly within the back-end Admin sections).

In the rotation/filter method of GHS, and in some of the developments considered and introduced for the new Integrated Household Survey (IHS), ONS is now reconsidering some of the boundaries it had previously placed on how it uses Blaise. In aspects like IHS's consideration of Prepare Directives, this was part of the continual review of the usefulness of aspects of Blaise's functionality not currently utilised.

On GHS, however, some of the features, such as checks between cases, would normally have been undertaken outside of Blaise. A Blaise solution was found, however, in order to provide a survey requirement in the light of time and people resources being unavailable from our usual Information Management providers.

With the complex inter-relationships across various blocks within this method, changes need to be thoroughly tested as, though they may perform the action required in the block in which they occur, the change may ricochet with unexpected results in later blocks using these variables. Minor changes in routeing or computation of a variable in the Filter block can, if not worked through blocks dependent on it, lead to incorrect outcomes being computed or false errors and messages being encountered by interviewers.

As with most Blaise writing, it is ultimately down to a balance between gains and losses. The cost for a survey in producing Blaise code for a complicated requirement may be too heavy for an organisation to bear. Manners and Green (1996) went so far as to state, "We contend that clever code … is a bad idea". Admittedly, this was in the context of "doing something in a non-standard way to shorten the code required", however their conclusion may still apply in this case, that, "If there is no discernible difference to interviewers and respondents in questionnaire performance between clever code which only programmers can understand and simple (even if redundant, lengthy) code which everyone can understand, one might argue that the latter type of code is more efficient."

ONS is currently undergoing organisational change, following the establishment of its Head Office at its Newport site in Gwent (South Wales). With all social survey posts transferred or transferring to this office by March 2008, the Blaise-knowledge pool of its new researcher-programmers is still relatively young and inexperienced. As Gatward (2007) highlights reliance on the expertise of a few (which more complicated code promotes) makes programming more susceptible to organisational change. In meeting its new challenges, ONS may need to reconsider its traditional approach on survey-specialist Blaise authors, working more in the hybrid model as detailed by Gatward.

From our experience, Blaise proves itself as a robust program capable of meeting the challenges that social surveys throw at it; the question that remains is the balance between what is possible and what is desirable.

5. References

Brown, A., Hale, A. and Michaud, S. (1998): Use of Computer Assisted Interviewing in Longitudinal Surveys, Chapter 10 of *Computer Assisted Survey Information Collection (Edited by M.P. Couper et al)*, pp.185-200.

Fiacco, A. (2007): The ONS Integrated Household Survey: The story so far. To be presented at the 11th International Blaise Users Conference, 2007, Annapolis, September 2007.

Gatward, R. (2007): Authoring Blaise Questionnaires – A Task for the Survey Specialist or an IT Programmer? To be presented at the 11th International Blaise Users Conference, 2007, Annapolis, September 2007.

Jäckle, A. (2006): Dependent Interviewing: A Framework and Application to Current Research, University of Essex, UK, Presented at Methodology of Longitudinal Surveys International Conference, 2006, Colchester, July 2006.

Manners, T. and Green, H. (1996): Blaise III: Who should do what? Internal Social Survey Division (ONS) paper, July 1996.