Part 4 - Blaise 4.8 to Blaise 5 Specification Conversion

Rhonda Ash - Survey Research Center, University of Michigan

1. Overview

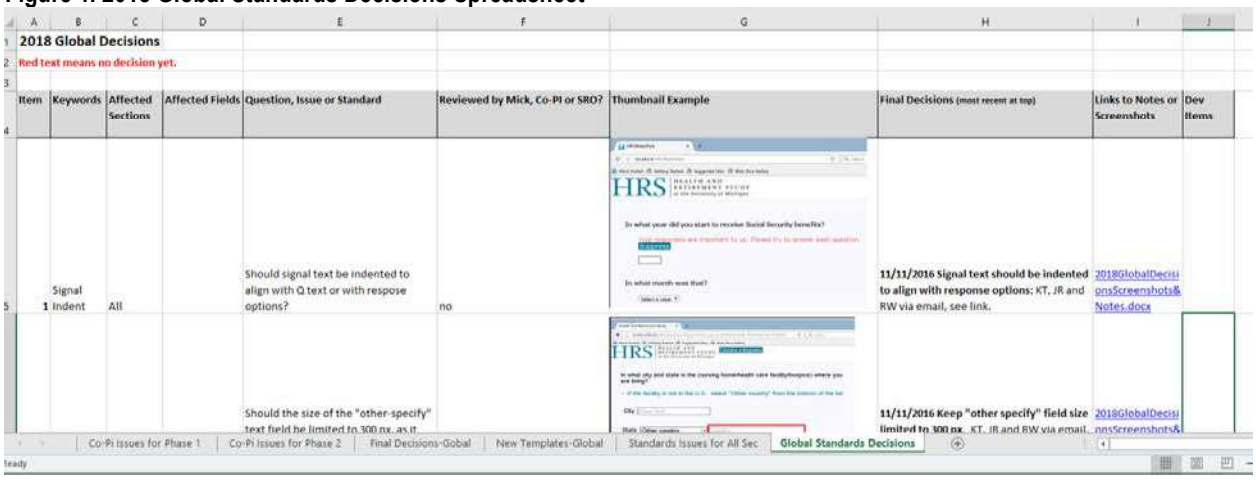
The Health and Retirement Study (HRS) is a national longitudinal survey on the health concerns and economics of aging and retirement. The HRS has, in the past, utilized a computer assisted interview (CAI) for biennial interviews of one to three hours in length given to around twenty thousand participants. Several years ago, HRS made a decision to convert to Blaise 5 so that we would be able to take advantage of the multi-mode features that are available.

There were many decisions involved in the conversion from Blaise 4.8 to Blaise 5. Some of the major decisions involved: the effects of language, mode and role changes; dropdown list vs. look up tables; signals vs. checks; how to handle grouping; and the handling of interviewer instructions. This was a complicated process that required hours of discussion before the first beta application. Following the beta-application, there were hours of testing, changing, re-testing, more discussion, changing, and re-testing again. We started with one of the "easiest" sections and spent 3 months until completion. Through that exercise we worked through most of the standards that would be used in later sections. The purpose of this paper is to describe some of the major changes that were needed to convert the HRS from Blaise 4.8 to Blaise 5, and outline some of the challenges and obstacles that were encountered along the way.

2. Decisions to be made

Very quickly we found that there were many opinions about how questions or fields should be presented and what effect multi-mode presentation would have on each screen. With a longitudinal survey, several decisions had to involve Co-PI input to ensure the content of the subject matter was not altered. In order to organize the vast number changes that needed to be considered, a chart was made to track decisions as they were made. There were oftentimes several decisions made on a single issue. All changes were tracked in an Excel spreadsheet that ended up being several tabs and many pages long. This spreadsheet proved to be a useful tool to track decisions and review from time to time (Figure 1).

Figure 1. 2018 Global Standards Decisions Spreadsheet



2018 Global Decisions

Red text means no decision yet.

Item	Keywords	Affected Sections	Affected Fields	Question, Issue or Standard	Reviewed by Mick, Co-PI or SRO?	Thumbnail Example	Final Decisions (most recent at top)	Links to Notes or Screenshots	Dev Items
1	Signal Indent	All		Should signal text be indented to align with Q text or with respose options?	no		11/11/2016 Signal text should be indented to align with response options: KT, JR and RW via email, see link.	2018GlobalDecisionsScreenshots&Notes.docx	
				Should the size of the "other-specify" text field be limited to 300 px, as it			11/11/2016 Keep "other specify" field size limited to 300 px. KT, JR and RW via email,	2018GlobalDecisionsScreenshots&	

Co-Pi Issues for Phase 1 | Co-Pi Issues for Phase 2 | Final Decisions-Global | New Templates-Global | Standards Issues for All Sec | Global Standards Decisions

3. Making Changes to the Specifications

3.1 A Migration Tool

We chose to write a migration tool (BlaiseSource) to help in the conversion process. This proved valuable as we were able to make changes quickly when the "standards" direction changed, and it did so often in the 3 year process. Much time was saved by allowing this program to make alterations like translation from '@' to <tag>, changing language tag (CorEng, PrxSpn vs. Eng, Spn), and adding banners.

As we went through the design and decision process we were able to add and alter different parts of the code using the BlaiseSource program. The procedure list grew rapidly (GENERATEIWERSELFTAGS, ADDROLES, ADDLANGUAGES, MOVELABTOROLE, FINDZEROS) and we found that we needed to be able to import code from other sources and alter specs for cleanup purposes (IMPORTNEWDESCRIPTORS, LISTDEFINEDTYPES, REMOVECOMMENTS, REPLACELANGUAGECONDITIONS).

Because of the multi-mode application, we needed to insert a new Web language into the specifications (GENERATEIWERSELFTAGS). Alterations to signals and help text all had to be re-tooled for a self-interview as well (REMOVEF1HELPNOTICE).

3.2 The Blaise Compiler

Changes had to be made to the Blaise compiler so it would correctly interpret the specifications. For example, an 'empty' or 'zero' value is evaluated differently in Blaise 4.8 and in Blaise 5. Reviewing compiler warnings after the compilation was needed (Warnings about 'empty' and Warnings about 'zeroes') (FINDZEROS). This needed to be done on a field by field bases to ensure what the correct intent was.

With the Blaise version changing so often we had to pay attention to the warnings that were presented when compiling. See Figure 2 for examples of commonly seen warnings.

Figure 2. Common Compile Warnings

```
12:48:49 : WARNING: Generated Page 1 (Layoutset Interviewing1) contains 2 Layout errors
12:48:49 : WARNING: Generated Page 1 (Layoutset Interviewing1, Parallel PRIMARY) contains 1 Layout error
12:49:41 : WARNING:     Self Reference: Field Relations of Block Type BA.BA_Relations is put on route with
generated parameter 'Relations.A023_PWSpPAlive' that contains an instance of the same type
12:49:21 : WARNING:  ==> Warning Line 3941 (51) - The result of the expression is always 'true'. Use = RESPONSE
instead. : C:\SVN\production\2018HRS\source\HRS18_T.incx
12:49:20 : WARNING:  ==> Warning Line  692 (32) - The meaning of this expression differs from Blaise 4. It is only
'false' when the left hand side has been assigned the value '0'. : C:\SVN\production\2018HRS\source\HRS18_S.incx
```

3.3 Blaise Version Upgrades and Changes

We found that with this large of a survey and the changes in the Blaise program (sometimes weekly) we had to test most features with each release in order to ensure they were not affected by the version upgrade. In order to efficiently test the areas of greatest concern, we made a mock survey (waterpark) that contained most of the areas we were concerned with (dropdowns, screen formatting, and selected grouping). This process sped up the testing and gave us a way to notify Stats if a problem was found.

3.4 Specification Upgraded for Language

Back in 2002 when HRS moved to Blaise 4.3, HRS designed the specifications to use language to control mode and role attributes. This allowed HRS to maintain only one set of specifications for the whole survey. HRS had not only a Core, Proxy and Exit questionnaire in English, but each counterpart in Spanish as well. The main advantage to this being that only one location need be updated when there was a change. See Figure 3 for the language breakdown in the 2016 wave.

Figure 3. Language and Roles Breakdown in the 2016 Wave

```
2016
LANGUAGES =
        CORENG "SELF - English", CORSPN "SELF - Spanish",
        PRXENG "PROXY - English", PRXSPN "PROXY - Spanish",
        SPPENG "SP PROXY - English", SPPSPN "SP PROXY - Spanish",
        EXTENG "EXIT - English",     EXTSPN "EXIT - Spanish"
```

With the new features of Blaise 5 we were able to break down mode, roles and language. Mode is whether the interview is self-administered or interview-administered, language is whether the interview is conducted in English or in Spanish, and roles are whether the interview is a core, proxy, spouse proxy or exit interview (Figure 4). We were still able to give one location to be updated when there was a change needed, but were now taking advantage of the new multi-mode fracture in Blaise 5.

Figure 4. Mode, Roles, and Language Breakdown in the 2018 Wave

```
2018
MODES =
  SELFADMIN DESCRIPTION ENG "TAKEN BY RESPONDENT,
  IWERADMIN DESCRIPTION ENG "ADMINISTERED BY INTERVIEWER,
LANGUAGES =
        ENG "ENGLISH",
        SPN "SPANISH"
ROLES =
          COR, PRX, SPP, EXT,...
```

This meant that most of the specifications had to undergo a major overhauling. Every field where there was a difference between the presentation of ‘interviewer’ (Figure 5) and ‘self’ (Figure 6) mode needed to have that defined in the specifications.

Figure 5. Interview-Administered Mode Specifications and Screen Display

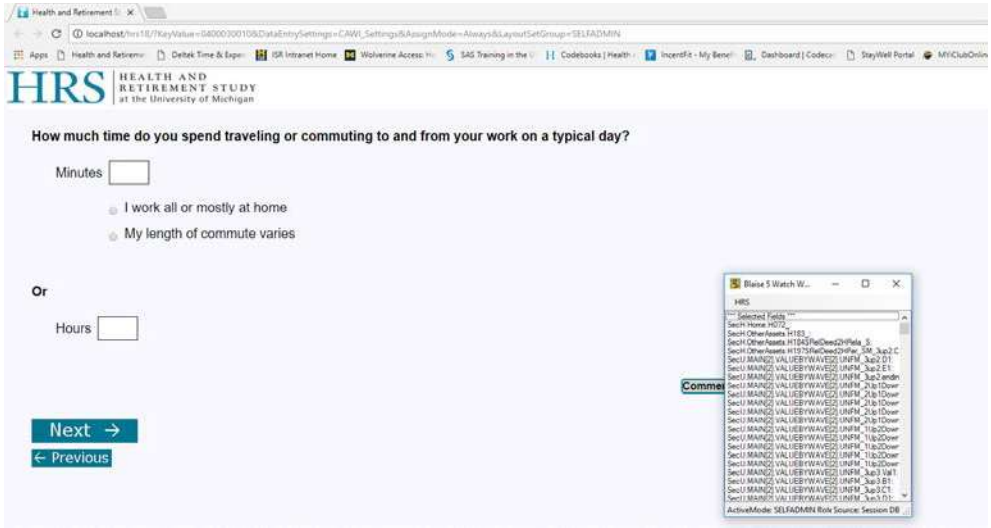
Interviewer mode

```
{CR} ENG "How much time do you spend traveling or commuting to and from your work on a typical day?
        <newline><newline><INST><img source=diamond>Enter 996 if R works all or mostly from home.
        <newline><img source=diamond>Enter 997 if R reports that the length of commute varies.</INST>
        <newline><newline><INST>Minutes:</INST>
        <newline>OR
        <newline>Hours:"
```

Figure 6. Self-Administered Mode Specifications and Screen Display

Self-Administered Mode

```
SELFADMIN
  ENG "HOW MUCH TIME DO YOU SPEND TRAVELING OR COMMUTING TO AND FROM YOUR WORK ON A TYPICAL DAY?"
    PRELAB ENG "MINUTES"
    PRELAB SPN "MINUTOS"
        TEMPLATES "SPECIALANSWER"
```



4. Switching the Specifications to SELFADMIN Mode

When considering how we could convert our interview-administered survey to a web friendly self-administered survey, our goal was to create a layout that was friendly to someone who had never taken a survey online, while also maintaining the interviewer-assisted application. By leaving the default as IWERADMIN, we were able to add to the current specs and change the text to adjust for SELFADMIN. This feature, along with major manipulation in the BLRD file, let us show one set of text for a Self-administered and another for the Interviewer-administered (Figure 7).

Figure 7. Interviewer Administered vs. Self-Administered Specification

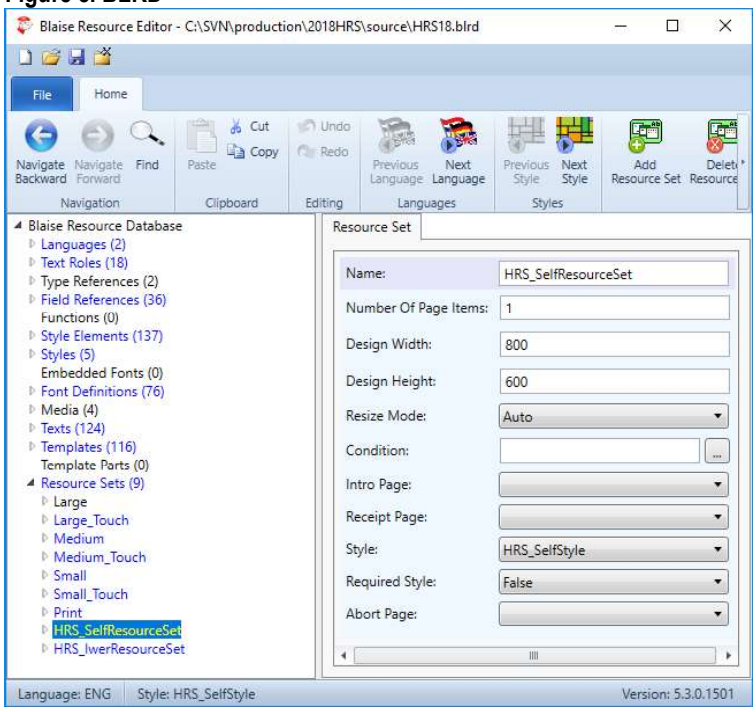
```
B126_ (B126)
  ENG "We'd like to know more about how old you were when you had the health condition(s) we just talked about.
<newline count=2><INST><img source=Diamond>Select continue.</INST>"
   {~08}

   /"Childhood Health Follow-ups Intro"
SELFADMIN
    ENG "We'd like to know more about how old you were when you had the health condition(s) we just talked about.
    <newline count=2><INST><img source=Diamond>Please select \"Next\" to continue.</INST>"
```

4.1 Managing Two Different Worlds in One Place - BLRD and Templates

Template control became very important. How would we work on both modes of the application at one time? – We split the BLRD into several slices by using layout sets (divide and conquer). This allowed both modes development and testing to go on together.
We had to quickly develop a naming convention so the layout sets could be worked on by two different programmers and merged back together in one BLRD (Figure 8).

Figure 8. BLRD



Using two layout sets to control mode caused the file size to more than double, which prompted concerns. The compile time extended to 1.5 hours. After talking to Stats about this, we started using the “Compile – pages” and “Compile – No pages” feature. The “Compile – pages”, (or FAST as we nicknamed it) took 1.5 hours to compile, but the survey responded in real time without generating the page as the survey was presented. Since there was such a time lag with the compile delay we also used the “Compile – No pages” (or SLOW as we nicknamed it). That allowed faster turnaround time for smaller tests. We would make many SLOW compiles throughout the day and a FAST once a day.

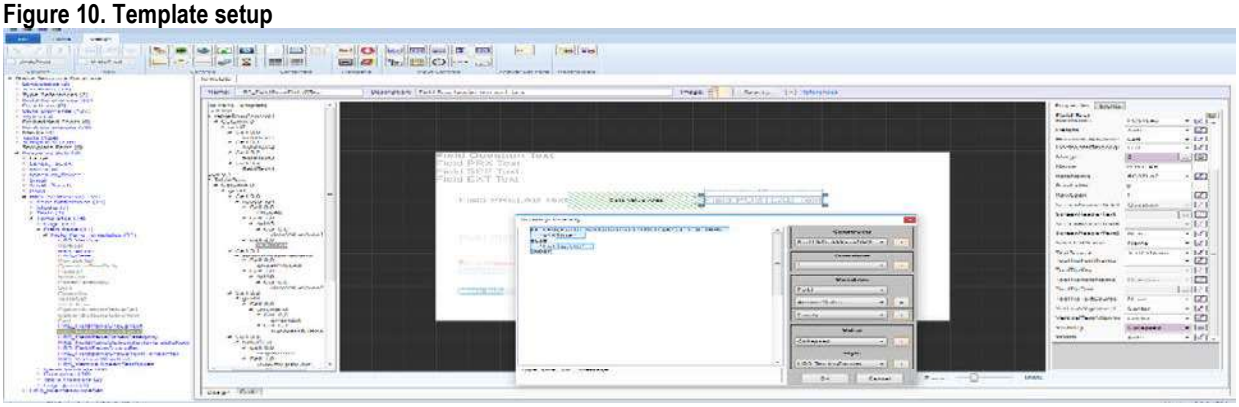
4.2 BLRD - Using Automated Templates and Layout Sets

There was a large learning curve to understanding controlling the BLRD features. We needed to know it well enough to automatically assign templates and mode based on field type or grouping. Without using this feature we might have never gotten our survey out to the field in Blaise 5. By using templates named to reflect the CAWI and CAPI mode, a naming convention was developed to allow automated usage of these template rules (FIELDQTEXTS, _GROUPTEXT, _GROUPTEXTWITHOR, _OTHERSPECIFY). After a template was developed and tested we could call it in the specifications and would know how it was going to present the screen. In the example in Figure 9, the template call in the code tells Blaise to use the attributes of the MaskLarge template for N014_, and nothing more is needed. Similarly, if we wanted to present a pre or post label on the screen at N014_ we would only need to specify that in the specs and the template would know to look to a role text of PRELAB and POSTLAB (Figure 9).

Figure 9. Template Usage in the Specifications

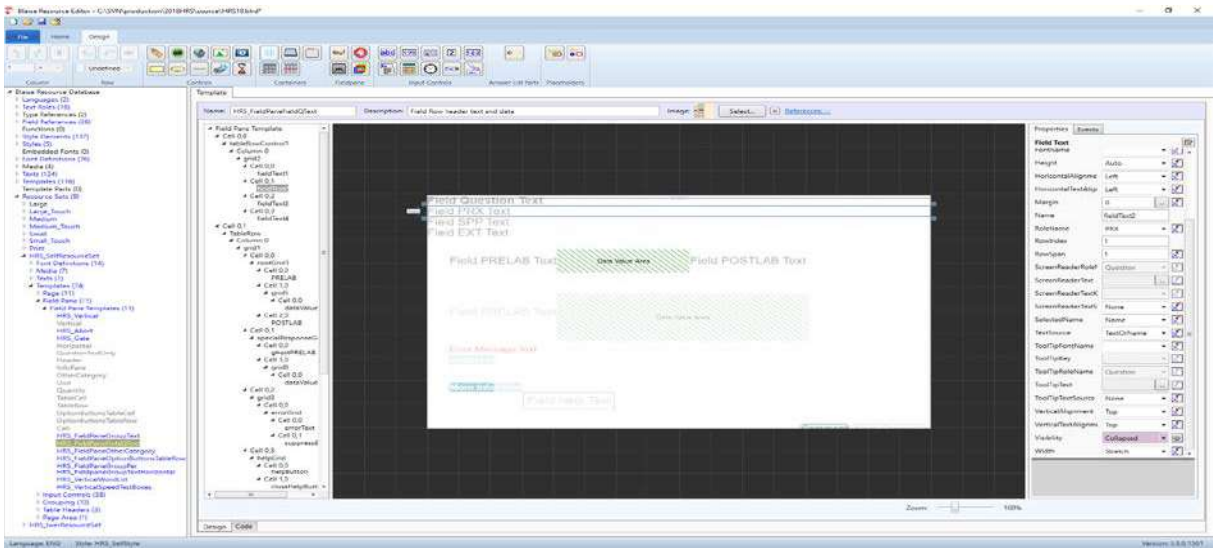
```
N014_ ("N014_")
   PRELAB ENG "$"
   PRELAB SPN "$"
   POSTLAB ENG ".00"
   POSTLAB SPN ".00"
 Templates "MASKLARGE"
```

Figure 10. Template setup



Another example of the BLRD control is using Expressions through the Visibility Property to control the cells of a template, allowing the display of one language role verses another (Figure 11).

Figure 11. Controlling the Display of Language Roles



Expressions code

```
  IF (IWType.EnumerationValue = 1 OR IWType.AnswerStatus = Empty) OR
((LEN(Field.GetRoleText('SPP')) = 0 AND LEN(Field.GetRoleText('PRX')) = 0) AND
LEN(Field.GetRoleText('EXT')) = 0) THEN
 'Visible'
ELSE
'Collapsed'
ENDIF
```

4.3 Grouping

A new feature to Blaise 5 is the concept of groups. Blaise help defines the purpose of the group section as: "Ties together related fields in order to allow a horizontal or tabular display and/or a special behavior on the screen." Of course, HRS has to add more complexity by having groups within groups (Figure 13).

In order to make templates for several different types of groups, the standards had to change and we had to think with a multi-mode brain. The BLRD was used for a lot of the control. These templates controlled the look and feel of several features, including the other specify screens, drop downs, tables, scales, and rosters. Again, we had to split these templates into the two modes to allow different presentations.

Figure 12. Blaise Resource Database

Figure 13. Template Examples

Self Grouping Template Examples
HRS_OtherSpecifyDefault
HRS_OtherSpecifyStatelist
HRS_OtherSpecifyAddressInfo
HRS_OtherSpecifyPer
DropDownList

Iwer Grouping Template Examples
SRC_GroupText
SRC_GroupTextPer
SRC_GroupScale

Figure 13. Example of Group in Group code (outer group- blue, inner group green)

```
Group Group_F176_GROUPTEXTPER
{CR}ENG "How often do you get together with people in or near the facility just to chat or for a social visit?
<newline><newline><INST><img source=Diamond> If 'almost never' or 'never,' enter \"0\" at number of times, and select
\"Next\" to continue.</INST>"
            Templates "GROUPTEXTPER"

GROUP GROUP_F177_OTHERSPECIFYPER
            Templates "OtherSpecifyper"
FIELDS
F177_FreqGetToget ("F177_")
                        PRELAB ENG "Per"
                        PRELAB SPN "Por"
                        AlienArgs "SignalOverride"
                        Templates "OtherSpecifyper"
            /"NUM TIMES GET TOGETHER WITH PEOPLE- PER"
            : TTimeUnit5Alt
F178SFreqGetToget_S (F178S)
            ENG "Other (specify):"
            SPN "Otro (especifique):"
            /"NUM GET TOGETHER W/PEOPLE- PER- SPECIFY"
            : OPEN
rules
            F177_FreqGetToget
            IF F177_FREQGETTOGET = OTHERSPECIFY THEN
                                    F178SFreqGetToget_S
            ENDIF
EndGroup

Fields
F176_NumGetToget ("F176_")

            AlienArgs "SignalOverride"
            PRELAB ENG "Number of times"
            PRELAB SPN "Número de veces"
            /"NUM TIMES GET TOGETHER WITH PEOPLE"
SELFADMIN
   Templates "GROUPTEXTPER"
            :0..995

Rules
F176_NumGetToget
            NumberMask (NO,F176_NumGetToget, Amount_Masked)
IF (F176_NumGetToget = RESPONSE AND F176_NumGetToget <> 0 AND IWERADMIN) OR SELFADMIN THEN
            GROUP_F177_OTHERSPECIFYPE
ENDIF
IF F176_NumGetToget  = 0 THEN
            F177_FreqGetToget := EMPTY
ENDIF

Endgroup
```

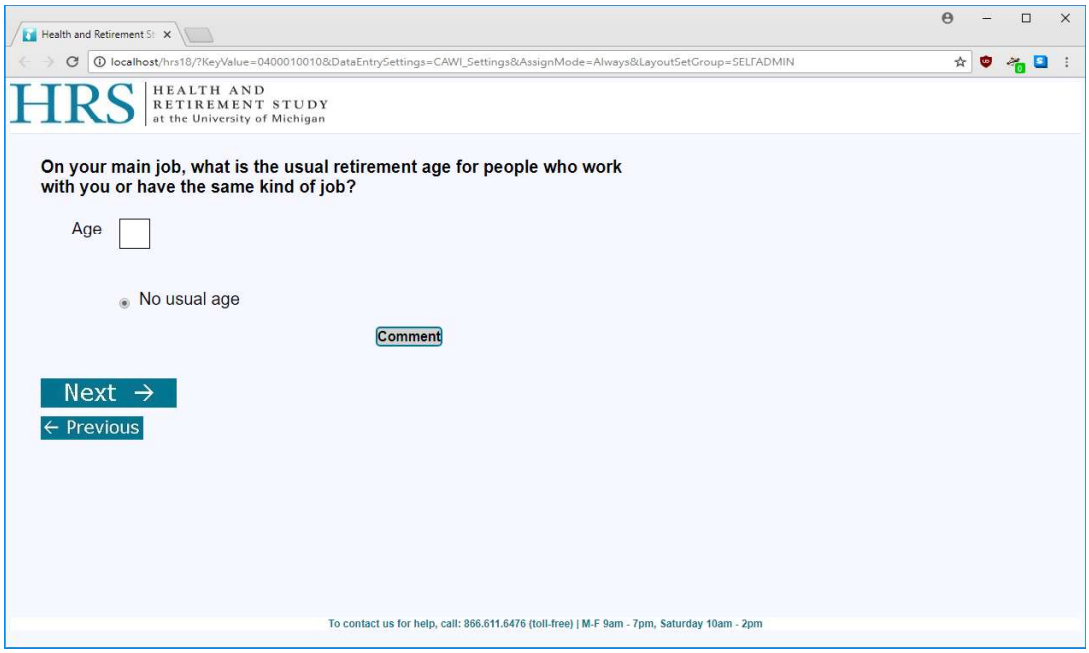
4.4 Special Answers

We used the special answer feature when there was a common answer over and above a range or codeframe. This allowed us to define data values that had special meaning other than the allowed range. We have an INCX file that holds all the "Special Answers" for HRS. (HRS18SpecialAnswers.incx). We applied the same naming convention to the answer types.
(SAV7A_V732 = "Special Answers", section V7A, Field V732). These "Special Answers" show on screen as code frames to be clicked if needed (Figure 14).

Figure 14. Special Answers Specifications and Screen Display

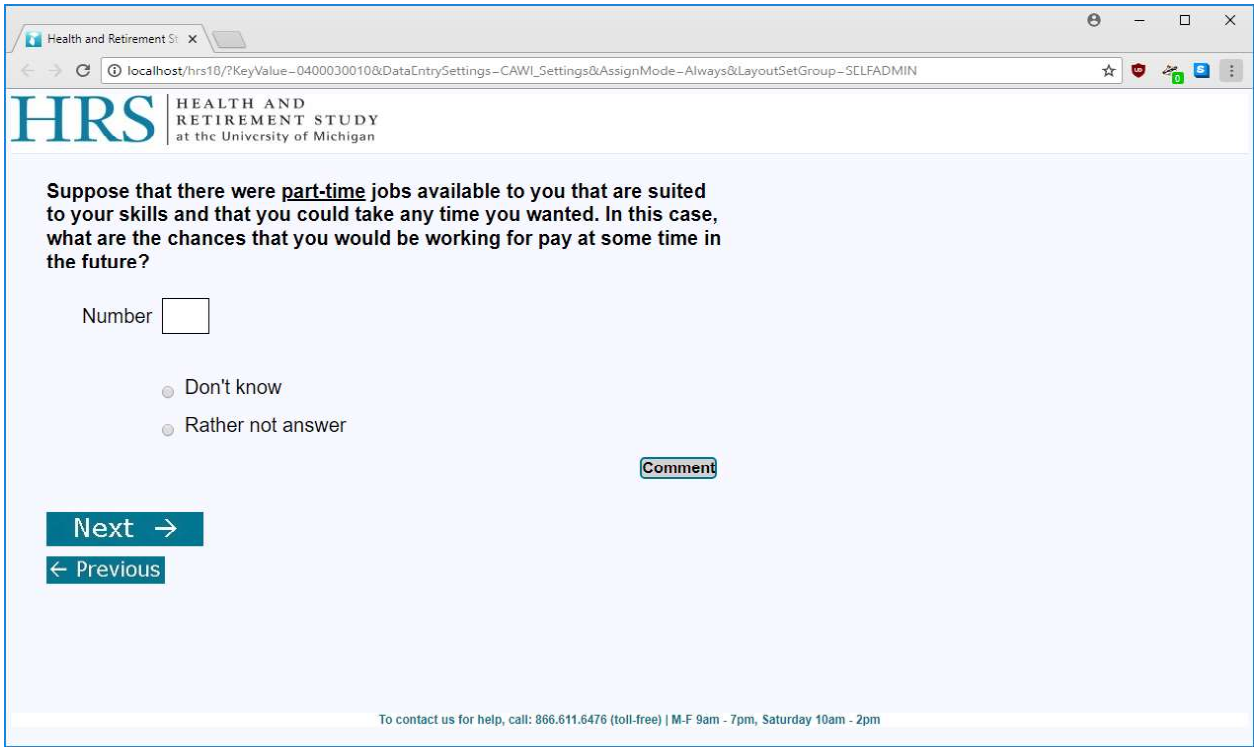
```
SPECIALANSWERS =
      SAJ564_95 ENG "No usual age" SPN "No hay una edad típica" {SP~18},
      NoAmount ENG "No amount" SPN "ninguna cantidad", {used in J529, J521}
      SAV7A_V732_996 ENG  "Works all or mostly at home " SPN "Trabaja todo o la mayoría del tiempo en casa" {SP~18}
      SELFADMIN ENG "I work all or mostly at home",
       SAV7A_V732_997 ENG  "It varies a lot" SPN "Varía mucho" {SP~18
                }SELFADMIN ENG "My length of commute varies" ,

 SPECIALANSWERSETS
    DefaultAttrib =
      SELFADMIN:  DONTKNOW, REFUSAL, EMPTY
      IWERADMIN: DONTKNOW, REFUSAL
 CAWINODKNORF =
      SELFADMIN: NODK, NORF, EMPTY
      IWERADMIN: DONTKNOW, REFUSAL
  CAWI_NO_RF =
      SELFADMIN: NORF, EMPTY
      IWERADMIN: DONTKNOW, REFUSAL
      SAJ564 = SELFADMIN: SAJ564_95, DONTKNOW, REFUSAL, EMPTY     SAV7A_V732  =
      SELFADMIN: SAV7A_V732_996,SAV7A_V732_997, DONTKNOW, REFUSAL, EMPTY
```



For a self-interview mode, we added the special answers for DK (Don't know) and RF (Refused) to several fields to allow the self-interview the option (Figure 15).

Figure 15. Don't Know and Refused Options for a Self-Administered Survey



Like the DK and RF these special values are assigned into the data and data processing will convert them to values for public viewing.

4.5 Checks and Signals

A decision that was made early on was that the Self-Administered mode had to allow the respondent to leave fields empty whereas the Interviewer-Administered mode did not. This meant that we had to adjust the signal and check logic so they were not presented with a warning if a field was left empty in self mode. Additionally, in order to reduce respondent burden, most checks became signals (Figure 16).

Figure 16. Specifications for a Self-Administered Signal

```
Signal F008_YRMADIED <= piINITA501_CurDateYear
 or Selfadmin
"Year entered is greater than today, if correct press suppress"
```

We also wanted to be able to change the text of the Signal/Check between modes (Figure 17).

Figure 17. Specifications for Differing Text Between Self-Administered and Interview Administered Signal

```
SIGNAL F002_MaAge IN [60..120] OR F002_MaAge = NONRESPONSE INVOLVING (F002_MaAge)
    "Values between 18 and 59 or between 96 and 120 are unlikely - please check"
SELFADMIN
    "Just to confirm, you entered ^{F002_MaAge}. If this is correct, select \"Next\" to continue.
            Otherwise please make the correction before continuing."
```

4.6 Instructions

In addition to changing signal text, instructions throughout the instrument had to be removed or softened between interviewer and self modes. See Figure 18 for an example of such an instruction.

Figure 18. Specifications for Differing Text Between Self-Administered and Interview Administered Instructions

```
F196_F036_ParST
("F036.5_")
    {CR}ENG "In what state do they live?
    <newline><newline><INST><img source=Diamond> Type 'OT' to enter 'other country'.</INST>"
  Selfadmin
    {CR}ENG "In what state do they live?
    <newline><newline><INST><img source=Diamond>If they do not live in U.S., select 'Other country'
   from the bottom of the list.</INST>"
```

4.7 Codeframes

Much like instructions and signal text, modifications were also made to codeframes so that self interviews referenced the first person and interviewer mode referenced the third person (Figure 19).

Figure 19. Specifications for Differing Text Between Self-Administered and Interview Administered Codeframes

```
MOTHMOVEDINWITHR (1)
   {CR}ENG "Mother moved in with R"
   SELFADMIN {CR}ENG "My mother moved in with me" ,
   RMOVEDINWITHMOTH (2) {CR}ENG "R moved in with mother
   SELFADMIN {CR}ENG "I moved in with my mother" ,
```

4.8 Help Text

Because a self-interview does not have an interviewer present to answer questions, we added more help text for the self mode. Help text for self-interviews was specialized to remove any reference to the “R”, was softened and addressed directly to the respondent (Figure 20).

Figure 20. Specifications for Differing Text Between Self-Administered and Interview Administered Help Text

```
HELP ENG "There can be some complicated ownership arrangements here. Simple ones are owns or rents, but the
farm could legally be a separate business that the R (or someone else) might own.  In this section, we are concerned
with the house and the immediately surrounding land; if it's owned by a business, it isn't a home owned by the R."

SELFADMIN
HELP ENG "In this section, we are concerned with the house and the immediately surrounding land;
if it's owned by a business, please don't count the home as owned by you, personally."
```

5. Conclusion

Decisions had to be made, designs changed, and tools had to be altered to allow the multi-mode longitudinal survey to transform to the Blaise 5 world. HRS still maintains one set of specifications that include not only multiple languages (ENG, SPN) and roles (Core, Proxy, Exit), but now includes Mode (IWER, SELF) as well. Programming the HRS survey is much like building a labyrinth with its many layers that all have to work together. This transition was a challenge that took several years to accomplish. With a lot of help from the Blaise team, along with the Survey Research Operations and HRS staff, we are now out in the field collecting surveys both with Interviewer assisted and self modes.