

Experience with Blaise 5 Layouts and Templates

Sharon Johnson, Richard Squires, and Michael Johnson, U.S. Census Bureau

1. Abstract

Moving a current Blaise 4 survey instrument to Blaise 5 involves much more than simply converting existing code to run in Blaise 5. With the emergence of touch laptops, tablets, and phones as viable interviewing devices, one must consider these different form factors and how they will impact CAPI survey data collection. The move to Blaise 5, with all of the new features, appears to be the appropriate time to redesign our CAPI data collection instruments so that they take advantage of, and work well with, touch screen devices.

The Blaise Authoring team has been researching, learning, and testing many of the new Blaise 5 layout and template features in order to determine a reasonable approach for moving forward into Blaise 5. This paper will discuss some of the findings, issues, and challenges we have discovered in this effort. The paper will also review the latest settings we have defined for our Master Template, some of the custom templates that we have identified for our surveys, and how we implemented some of the Blaise 4 features into Blaise 5.

2. Overview

The Blaise Authoring team has been researching, learning, and testing many of the new Blaise 5 layout and template features at the Census Bureau in order to determine a reasonable approach for moving forward with Blaise 5. This paper will discuss some of the findings, issues, and challenges we discovered along the way. The paper will also review the latest settings we have defined for our Master Template, some of the custom templates that we have identified for our surveys, and how we implemented some of the Blaise 4 features into Blaise 5.

During the research effort, the Blaise Authoring team took the opportunity to develop the 2010 Census short form from scratch using the Blaise5 software. This allotted the research team to familiarize themselves with the additional enhancements and bug fixes that were provided with the more recent releases of Blaise 5. We discovered that using the Blaise Resource Editor to modify the layout presented a lot of flexibility when designing the look and feel of an instrument. This document identifies some of the capabilities that we discovered in the implementation process, and documents the experience based on this implementation.

2.1 Development Approach

Initially, we considered using the current Blaise 4 layout presentation which incorporates the Infopane and the Formpane as the Census Bureau standard. After further consideration and research of industry standards, we decided to use the Blaise 5 design concept. The design approach focused on the use of a touch screen device and the use of a laptop keyboard. With this approach in mind, we considered the use of buttons, dropdown lists, and hyperlinks for a successful execution of the questionnaire.

Once an initial prototype was developed, a user testing session was held with some developers to collect feedback. Changes were implemented, and then the process was iterated. At the conclusion of three of these sessions, the process was repeated for three iterations with the sponsors/end users.

Our comparison is with two different methods of collecting the same information. We developed both, a person-based and a topic-based solution. Each version incorporates different types of features made available by Blaise 5.

2.2 Design Approaches Implemented

The following suggestions were presented and implemented into the prototype.

- Allow 2-3 questions per screen.
- Highlight the active field when in a table.
- Auto advance on questions with radio buttons to speed up flow of navigation when either the radio button or the text is selected.
- Display a status bar with minimum information (must include the respondent's name). Additional case information is made available through the "Info tab".
- Provide the ability to use the arrow keys to move between questions if using a keyboard.
- Implement Signals involving multiple questions on different pages.

3. Blaise 5 Software Features Researched and Incorporated

3.1 Toggle Visibility

This feature is useful in the display of certain graphics depending on if it is to be used by the person administering the questionnaire. For example, the Help icon is displayed only if the field has help available. Another example for use of this feature is if a question allows for Don't Know or Refusal as an answer option.

3.2 Tooltip

This feature provides the capability of hovering over a field to display additional help instructions. Since this feature is not available on mobile operating systems (no cursor), we have not determined if this will be a common feature offered for all Census surveys.

Figure 1. Example of Toggle Visibility and Tooltip

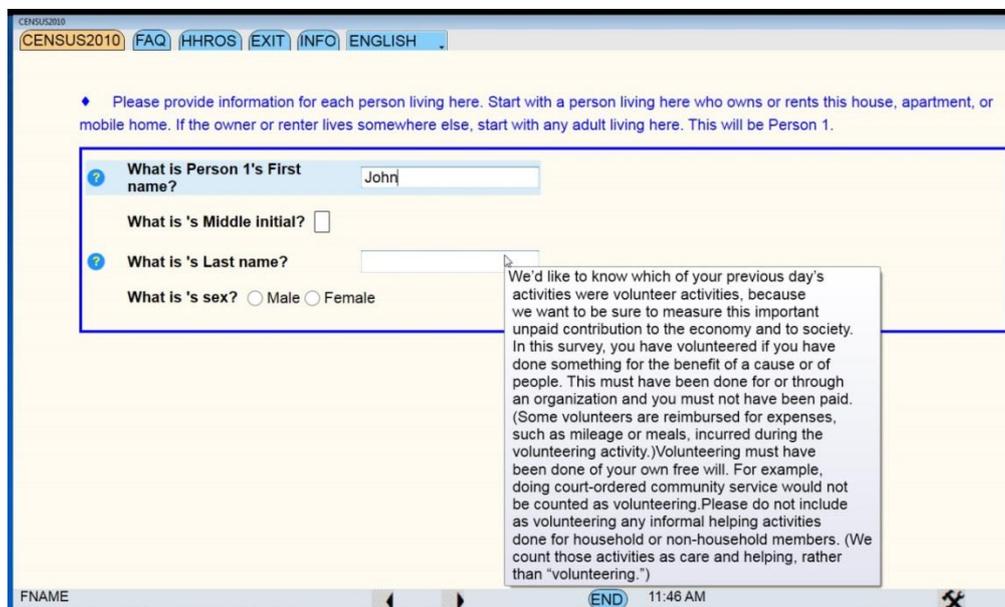


Figure 1 - This screen demonstrates both the Toggle visibility with the use of the Question Mark to represent help, and the Tooltip that displays a large amount of text that can be used for the help feature.

To incorporate the Tooltip functionality, first add Tooltip to ROLES. Then add tooltip to the field that will use the feature. See sample source code below.

Listing 1, Sample Tooltip Source Code

```

ROLES          = HELP "example of help",
                Tooltip "The ToolTip role provides a popup hint when the
                        user hovers over a control."

PHONE ENG "<newline><B>What is your telephone number? We may call if
           we don't understand an answer."
           ESP "<newline><B>TELÉFONO DEL HOGAR." /"Phone number:"
           Tooltip "Enter 10 digit phone number (no dashes)": TPHONE
    
```

3.3 Templates

Several templates are customized to design the layout that is currently being used. These templates have been implemented and tested using Blaise 5.04 b875.

In summary, we customized 21 templates in order to accommodate the design specifications received from the Sponsor and Developer inputs that include:

Table 1. Templates

Master Templates:

Template Name	Description
CensusStandardGrid	To accommodate Grid Style Structure
CensusStandard	To accommodate 2-3 questions per screen

Custom Templates:

Template Name	Description
InfoStatusPage1	Called when the INFO tab is selected
EndPage (EXIT)	Allows option to exit or return to survey
TOOLS	Accommodates tools used during the interview
RosterPage	Presents household roster information

Intro Page Template:

Template Name	Description
CensusDefault	Welcome screen with logo (picture)

Receipt Page Template:

Template Name	Description
CensusPrint1	Presented after the interview is completed. Thank you screen also provides option to PRINT. Show Household members interviewed, case status and Length of Interview.

Field Pane Templates:

Template Name	Description
Vertical1, Quantity1, Horizontal1, QuestionTextOnly	These were modified, so that the error message does not show directly under the field (since we are now showing standard error message location at the top.)

Data Value Templates:

Template Name	Description
DropDownList_age	Setup specifically for age drop down, made smaller and shows watermark "Enter age 0-120".
AnswerList	Answer list Area
DropDownList	Drop Down List Area

Category Templates:

Template Name	Description
Set	Set Category Code and Text
Enumeration	Enumeration Category Code and Text
RadioButton	Category Radio button
Enumeration FAQ	AutoEnter set to True for easy Navigation

Language Templates:

Template Name	Description
Default1	Specialized look for Language Button

Parallel Templates:

Template Name	Description
Default1	Specialized look for Parallel Tabs

3.4 Questions

We use the “highlight active question” feature while navigating through the questionnaire. In addition, for questions with a radio answer list, we make the answers selectable and change the color when advancing to better identify the answer that has been selected.

3.5 Edits/Errors

For displaying edit/error message information, we did not want the text to expand the screen or to incorporate scrolling to display errors. Based on these requirements, the following were considered: The use of a pop-up box similar to Blaise 4. However, this functionality is not available due to its incompatibility across multiple platforms.

- Creating our own pop-up. However, the software does not support creating an “overlay” screen that can sit on top of the Master Template.
- Using a box, reserving the right side of the screen and have it display the active error. However, this impacts the question/response texts on the screen and pushes them to the side
- Removing excess lines at the top of the screen, and reserving one line in that area for error message.

We decided to remove excess lines at the top of the screen, and reserve one line in that area for error messages. This works okay and allows for expanding as necessary.

3.5.1 Date of Birth (DOB) Question

Multiple approaches were investigated for the date of birth question.

1. An attempt to start the year range at a value within reason.

The current Date dropdown provided in Blaise 5 starts with the current year – 120 years. So for 2016, the list starts at 1896. Feedback provided to the committee requested that this date be “seeded”. In other words, start at a different place (e.g. 1950), based on the population being interviewed to prevent a lot of scrolling. This is currently not supported in Blaise 5, so alternative approaches were considered.

2. Before the DOB question, an age dropdown was provided that allowed for up through 120 years of age.

- Based on the age value provided, an approximate start year was determined and used for picking from a drop-down list.

- Month and day fields were pre-filled with January 1 and a signal applied if the default value was not changed. We ultimately decided not to prefill the date field.

Figure 2. Example of DOB Question, Error Icon, and Error Screen

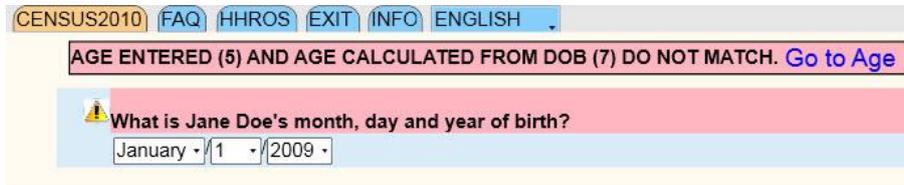


Figure 2 - Example of DOB question, error icon, and error screen. A link is provided in the error area that allows navigation to the Age field for correction.

- An error icon is next to the question to more easily identify what needs attention since more than one error may occur on a page.

3.6 Navigation

We tried several different options for the advancing between pages. As we consider touch screen, real estate, and moving current development to mobile devices in the future, it was decided to use the arrow icons (translatable pictures feature) instead of the previous and next buttons. The up and down arrows on the keyboard did not work initially. We were finally able to implement the up and down arrow navigation by using shortcut events CtrlDown (down arrow) for NextField, and CtrlUp (up arrow) for Previous Field. The shortcut is set at the MasterPage and will work wherever the MasterPage is used.

To implement the shortcut key assignments, you have to:

- In the resource editor, select your master template(s), the very first row (Master Page Template), select the events tab.
- Click on the icon for edit shortcuts.
- Add OnCtrlUp (keystroke) and add OnCtrlDown(keystroke).
- For OnCtrlUp, assign the action Previousfield, for OnCtrlDown, assign the action Nextfield.

3.7 Swiping

We used the Blaise 5 default of SwipeLeft = NextPage and SwipeRight=PreviousPage. During our testing, we found that while using one particular tablet, the longer Swipe Left and Right options would actually move us 2 pages forward or backward. We only discovered this happening on this device, other devices this feature worked fine.

3.8 Parallel Tabs

The parallel tabs were originally developed as just rectangular boxes. We modified these so they displayed as round edged buttons with some spacing between, which allowed for easier selection. The final design kept only the rounded top edges with spaces in between. The language selector was placed in the same area, but with a different look and feel. (See Figure 2 above)

3.9 Help/Flashcards

Initial viewing focused on the help being presented similar to the way it is in Blaise 4, using an external file. The belief is that when help is executed it should be consistent.

The syntax used in Blaise 4 doesn't work in Blaise 5. We were using an internal hyperlink that required a local channel to the local service that is running on the laptop. There is a separation between the Blaise program running and the server service.

For Flashcards, we were able to successfully use an individual .pdf file for each of the screens that use a flashcard. However, this required us to generate a lot of .pdf files and didn't seem very efficient. Our initial thought was to use one .pdf file and pass a parameter to the appropriate destination in the file. We could not successfully implement this as Blaise 5 was not accepting parameters.

- Alternatively, we implemented the external call for Help using .chm. We are exploring the use of the Tooltip, question help, and html files to display for help since .chm will not function properly on mobile devices. Not all help text needs to call an external browser window; as a result, some fields incorporated the use of the ToolTip option to display help text including a large amount of information. (See Figure 1, above for an example of ToolTip help.)

If help is available for a question, the help icon is positioned to the left of the question text. Both Help and Flashcard display content will be accessible by touching a clickable link near the question.

3.10 External Lookup Tables

We added an external State lookup table to our solution. To accomplish this, we did the following:

Using the Blaise4to5 converter, convert the stateslist.bla to stateslist.blax.

Listing 2. External Lookup Table Source Code

```
STATESLIST.BLAX
DATAMODEL
  SECONDARY
    StatesName = STATEText
  FIELDS
    STATEABREV : STRING[2]   {pos 1 - 2 }
    STATEText  : STRING[40]  {pos 3 - 42 }
  ENDMODEL
```

Next, we convert the stateslist database from Blaise 4 to Blaise 5 using the data converter for use in the code. Add the StatesList to USES.

Listing 3. External Lookup Table Source Code

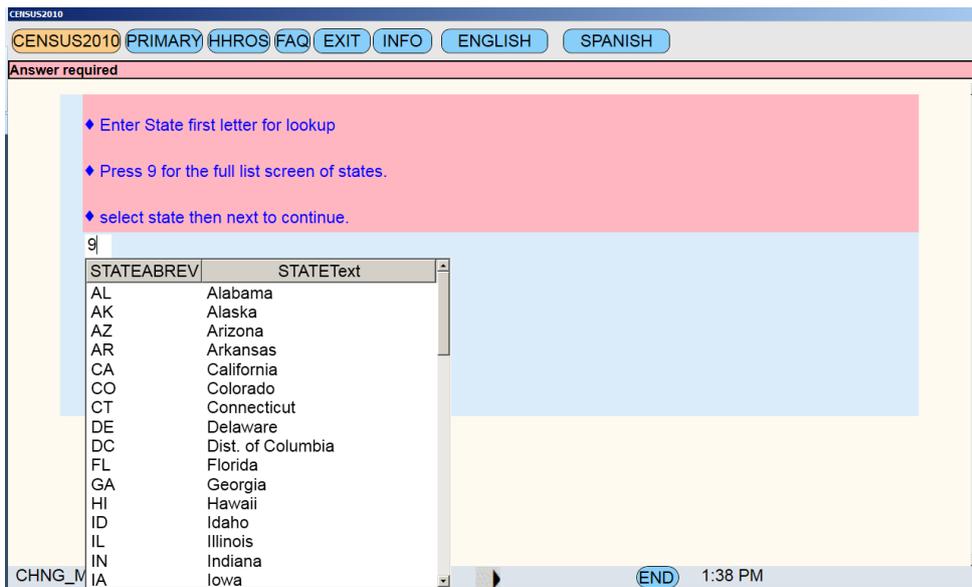
```
USES
  StatesList

{Lookup from statelist database file}

CHNG_MSTATE | STATESLIST.LOOKUP(StatesName).STATEABREV
IF CHNG_MSTATE = RESPONSE THEN
  MST:= CHNG_MSTATE
ENDIF
```

The above code calls the "stateslist" database when the field has an entry. The selection table pops up. The user can click on a selection which then closes the table. This puts the selected state abbreviation in the field.

Figure 3. Example of the State Lookup Table



Initially (5.0.4b875), it did not seem that the look-up was accessing the entries as we had intended based on the value we typed for searching. It was difficult to select with the touch screen due to size. To offer options, we made two fields. One field allowed the user to type the entire state abbreviation in to initiate the search.

The other field allowed for the look up. Typing a 9 on the first field goes to the second field. We also allowed for the entry of a '9' to bring up the entire State list. Alternatively, entering the 2-character state abbreviation displays the list of entries that start with that letter.

We found that using the arrow keys to go to the state and pressing enter caused problems. The selection table would start at the first state with the letters typed in. There was no option to go back in the list. It goes to the next item if there is not a match available for what you typed. For example: If you typed in "MA" for Maine, it would sometimes put in MA for Massachusetts. If you typed in "MD" for Maryland, the list would start at Michigan. The best way to get the item you desired was to type in the first letter and click with the mouse for the correct selection.

Selecting with the mouse click worked correctly each time. Initially, the display window only displayed about 10 rows. We inquired on the possibility of opening up the window so that it displays more of the list that is available from top to bottom. A later version of the software provided this capability.

3.11 Tables

Figure 4. Example of a Table Using Buttons

What is firstname4's Last name?

##	Status	First Name	MI	Last Name	Gender	Related
1	Add	firstname1	1	lastnamefamily	M	Reference person
2	Add	firstname2	2	lastnamefamily	F	Husband or wife
3	Add	firstname3	3	lastnamefamily	M	Biological son or daughter
4	Add	firstname4	4	lastnamefamily	F	Biological son or daughter
5	Dele	firstname5	5	lastnamefamily	F	Biological son or daughter

LNAME END 1:02 PM

The table in figure 4 demonstrates the use of a button in column “##”. After entering all of the information for a household member, the corresponding number button becomes active by turning Yellow. Once active, the button allows navigation to the demographics section for that person. The button color change to orange once partial demographics information is entered. A green button is displayed if the demographic information is complete for that household member. Displayed above the table is the current active field question text. The status bar, defined in the Master template, will display the Don’t Know and Refusal buttons when they are valid for the active field.

The use of tables was simple once we learned how to do the grouping. Initially we had trouble attempting to implement the following features:

1. **Changing the background color when a field was active.** This issue was solved in later versions. Our temporary fix was to use a watermark on the active field.
2. **Displaying the question text in a table.** This was temporarily within four master templates. This was also solved in a later version of Blaise. We started with multiple master templates, but with more knowledge, we were able to reduce this down to one template.
3. **Jumping issue when the answer was a different size.** The column would grow and shrink based on the answer in that column. This was solved by using the stretch setting and setting the size to the maximum size needed for the field.
4. **Displaying the entire answer of a drop down.** The display window didn’t seem to be wide enough. Alternatively, we put an abbreviated answer option at the front of the answer list.

Figure 5. Example of Table with Dropdown Content

The screenshot shows a web application interface for 'CENSUS2010'. At the top, there are navigation buttons: CENSUS2010, PRIMARY, HHROS, FAQ, EXIT, INFO, ENGLISH, and SPANISH. Below this is a section titled 'What the status' with a table. The table has the following columns: #, Status, First Name, MI, Last Name, Gender, and Related. The first row is highlighted in green. A dropdown menu is open for the 'Related' column of the first row, showing a long list of options including 'Reference person', 'Husband or wife', 'Biological son or daughter', and 'Deleted visitor'. The dropdown menu is truncated, with some options cut off at the bottom. At the bottom of the interface, there is a status bar with 'Status', navigation arrows, 'END', and '10:48 AM'.

#	Status	First Name	MI	Last Name	Gender	Related
1		firstname1	1	lastnamefamily	M	Reference person
2					F I	Husband or wife
3					M	Biological son or daughter
4					F I	Biological son or daughter
5					F I	Biological son or daughter

Figure 5 –The dropdown contents are larger than the answer in the table causing truncation of display. DK/RF will also display in the dropdown if they are valid answer options.

5. **Making the connection of rules and template variables.** We added field references. This linked the fields to the templates and the templates could associate the field value from the rules.
6. **Placing buttons on a table.** We were able to master this with assistance from “Statistics Netherlands”.

We found that setting up the layout setting is still time consuming. Using “promote” for each setting proved to consume allot of time. For example, we had a table with 18 items and used promote 5 times per item. This required 90 promote picks and if the design was changed, we had to do again. Another challenge we encountered is identifying the best way to handle different data types in the table.

Figure 6. Example of Templates

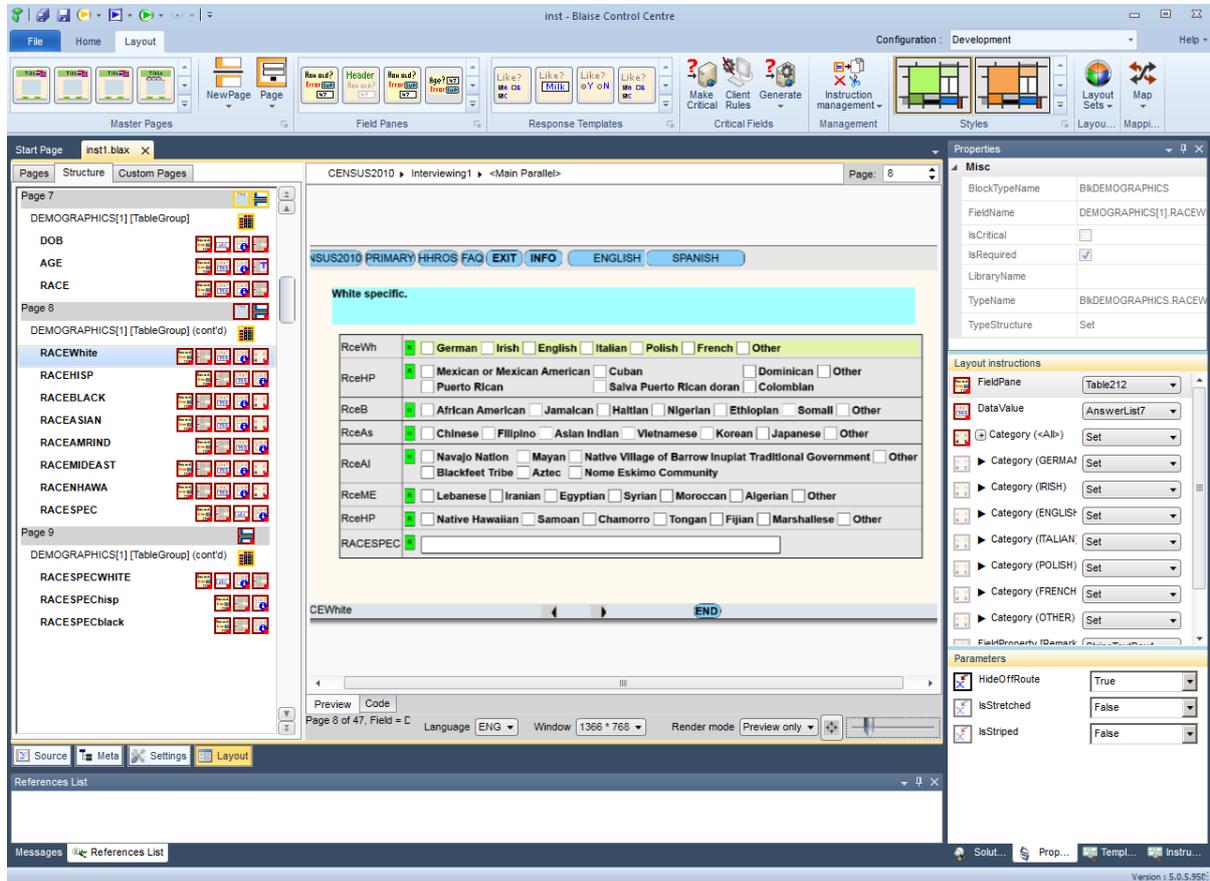


Figure 6 – Demonstrates the use of Templates on top of templates. There is a Master Page template, a table template, and 5 other templates used to display the row for RceWh.

Guidance is needed when it comes to adjusting for drop down list, button answers, select all that apply, enumeration, and other in a defined table. Alternatively, we made data value templates. We set the number of columns for each template. This optimized the display space for different answer size and the number of selections.

3.12 Languages

The languages indicator is displayed in default format near the parallel tabs on one prototype. Once a cell in the template is defined to activate the languages, incorporating this feature is easy. Using the Page Controls under the design tab, clicking on the language selector icon drops the parallel tab to the specified location. The languages are automatically picked as defined in the source code. A drop down list at runtime allows switching between the languages.

We looked at offering separate language buttons on one of the prototypes (see figure 5 above). The separate buttons makes it easier to switch between the two languages. However, that approach utilizes a lot more space in the tab area especially if a lot of languages are defined.

3.13 Status Bar

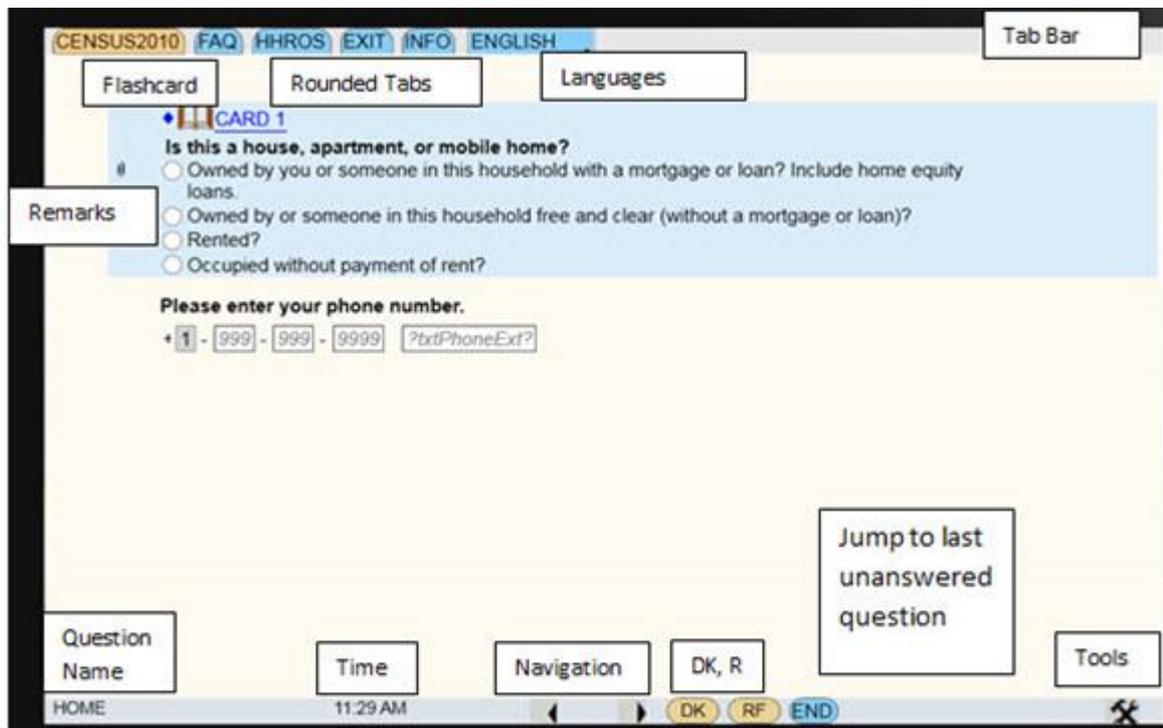
Once the information was defined to be placed on the status bar, it was important that these items were always displayed in the same place, and the displacer used only one line of real estate. Due to information presented in this area and the amount of real estate available, we decided to use front and back arrows instead of the default buttons with text. This helped reduce the “wordiness” of the status bar. Other, minimal data was also placed here (question name, respondent name, time, navigation, DK, R, and tools), with all other relevant data being accessible in either the INFO or HHROS parallel tab. (See figure 7 for an example of information in the status bar)

3.14 Phone Masking

We discovered that masking capabilities typically used for formatting phone numbers was not available yet, but would be implemented in the October 2016 release of Blaise 5. In the meantime, we created a template layout to handle the collection of the telephone number. (see figure 7). This involved the following steps.

1. Identify the field names (Country, Area, Prefix, Suffix, Ext)
2. Add the Type Reference in the BLRD for each of the fields. (Example: TPhoneArea String)
3. For each of the fields, add User Defined text. (Example: txtPhoneArea text:999)
4. Add a Template for the data fields.
5. Add a data value template in the Input Control Option.
6. Add a response value template. Use WaterMark to display format.
7. Add a table template in the Grouping option.
8. Define a Type for each of the fields.
9. Place all of the fields in a Group FullPhoneMask to collect the phone number value.
10. Define the format in the rules within the Group.
11. In the layout set, apply the PhoneMaskTemplate to the FullPhoneMask[TableGroup].

Figure 7. Example of Telephone Template and Other Features



3.15 Don't Know and Refusals

The consensus at the agency was that it was not a good idea to hide the Don't Know and Refusal if they were acceptable answer options. However, when they were available, we did not want them to expand the layout of the screen. As a result, we explored the option of not showing the DK and R in the answer list and use buttons on the screen to represent these options when they were valid answer options. To incorporate this feature, we had to add a parameter for the visibility at Templates | Input Controls | Special Answer Templates | Button. In the layout template, we set them to collapse.

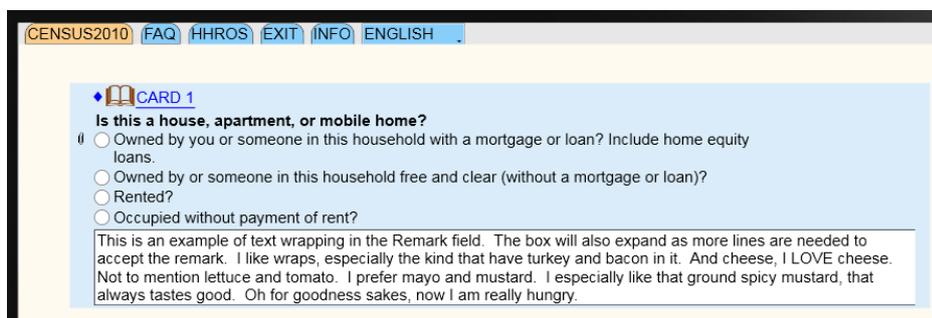
3.16 Remarks – (with the release of 505.b918)

We created a Remarks snippet that can be used and makes it easier than writing XML code required for each field that uses the feature. We think this feature should be available for every field in the questionnaire.

Initially, this feature presented us with a single line box. If a long remark was entered, it would just scroll off the screen to the right. We modified the appearance of this feature to allow for an expandable box with text wrapping. This required a modification to both the vertical and horizontal templates. The remark text box goes away with the advance to the next page. This refresh does not appear to happen on the same page even if you have the setting to hit the server after a changed question.

We could train our field representatives that a field will always have the ability to enter a remark. We noticed that standard Ctrl+M does bring up the box underneath the field with no positioning, but does not automatically send the focus to the Remarks entry area. You have to actually touch the remark area to start typing. You have to press Ctrl+M again for the remarks text box to go away unless you type something. The remarks field does not accept “Enter” as an entry.

Figure 8. Example of Multi-line Remarks



With touch in mind, we researched the possibilities of accessing the Remarks using a single touch motion without use of the keyboard. A Green “R” is displayed to allow touch for entry of a remark. Once a remark has been entered, the paperclip is displayed.

Figure 9. Example of Remarks Feature

DOB	<input type="text" value="R"/> <input type="text" value=""/> <input type="text" value=""/>
Age	<input type="text" value="R"/> <input type="text" value="Select a value"/>
Race	<input type="checkbox"/> White <input type="checkbox"/> American Indian or Alaska Native <input type="checkbox"/> Hispanic, Latino, Spanish <input type="checkbox"/> Middle Eastern or North African <input type="checkbox"/> Black or African Am. <input type="checkbox"/> Native Hawaiian or Other Pacific Islander <input type="checkbox"/> Asian <input type="checkbox"/> Some Other Race, ethnicity, or origin
Age	<input type="text" value="R"/> <input type="text" value="Select a value"/> <input type="text"/>

Age

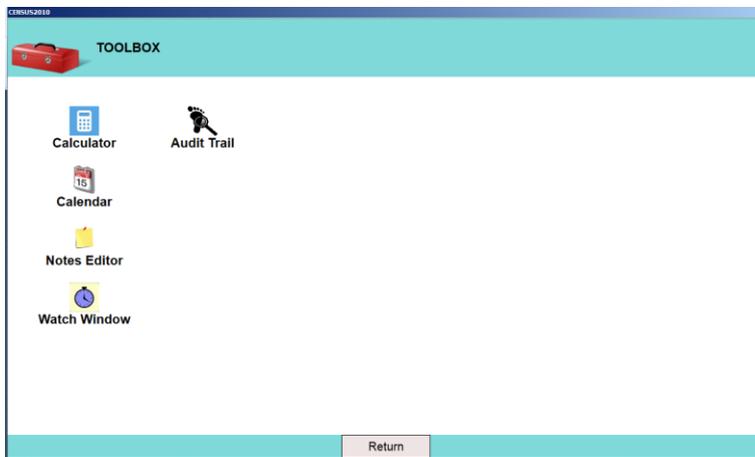
The remarks feature works well.

3.17 Tools

We asked about the possibility of adding a drop down picklist similar to the Language selector. We think this would be beneficial for providing a list for the various tools available as well as handle surveys with multiple parallel tabs incorporated for interviewing sections. The response received was that this kind of functionality, dropdowns, menu-bars in general will be available when Maniplus has been implemented.

Alternatively, a tools icon was placed on the far right edge of the status bar (see Figure 7). This is a link, that when pressed directs focus to a custom page. The custom page displays all of the tools that have been made available and are identified below.

Figure 10. Example of Tools Available via the Tools Icon



3.17.1 Basic Tools

Based on use in the Blaise 4 instruments, some basic tools were incorporated. These tools included the calculator and calendar. In addition, an attempt was made to incorporate the Audit Trail, Watch Window, and a Notes Editor.

3.17.2 Audit Trail

An external program was written to access the audit trail data from the SQLite database "AuditTrailData.db".

3.17.3 Watch Window

Utilizing Software from University of Michigan, a C# project was created to demonstrate how to use the API's to generate a Watch Window Application, resembling the Blaise 4 Watch Window. A Windows Presentation Foundation (WPF) uses the same services to secure the applications. The goal was to run the Blaise 5WatchWindow.exe when the browser of the Wpf DEP is active for the instrument having a primary key.

Example:

```
Blaise5WatchWindow.exe -KeyValue:100 -InstrumentID:5ebfd908-c4c0-4959-bf0b-663829c287e2
```

Then you can choose fields from a field selector. The values of these fields will be displayed in the window; possibly changing whenever the rules of the interview are executed.

4. Features that worked well

While considering the redesign of CAPI instruments and future of touch screen devices, we found a number of features that worked well:

4.1 Buttons

Work well and are very flexible. They can be used for navigation, running routines, and have the ability to assist with implementing the look and feel of the questionnaire.

4.2 Fonts

Appear to be unlimited to how these can be defined. However, a lot of existing Census instruments use Wingding fonts, which are not currently supported. We understand this will be available once Blaise 5.1 is released.

4.3 Layouts

Once set, adjustments are easier.

4.4 Tooltip

Offers flexibility in amount of text allowed as well as positioning on the screen.

4.5 Watermarks

The Watermarks can be easily defined. They can also be used for hints, and prefilling help on how data should be entered. To save real estate, this feature can be used for our interviewer instructions.

4.6 Multimode

We like that this is supported. This feature can be pretty powerful. We are exploring this more based on devices and potential navigation.

5. Challenges Encountered

We, of course, encountered some challenges along the way. The following are some attempts that we made without success:

1. **Sub screens** - If we were allowed the option to create another screen smaller than normal, then we think this will allow us some additional flexibility. This could be used for studies that have a large amount of parallel tables, an “Info Tab Screen”, or even a table with a dropdown list containing a large number of answer options. This is similar to what is displayed when multiple languages are available.
2. **Numeric Dropdown** - We noticed that the dropdown feature is only available for character fields. We think a dropdown for numeric will also be useful. This can be used on touchscreen to select number of people in a household.

These are some features we feel can be useful, but didn't fully work for us.

5.1 Use of .pdf for help

We found that we really needed the ability to pass parameters from the command line for this to work as we would like.

5.2 Print

We thought this was a neat feature, but found that we could not use the feature outside of the Blaise 5 software. For example, we could not use this feature from a scripting language.

5.3 Tables

We would like to know the best way to display a table with multiple data types and the behavior of tables in a group.

5.4 Answer Required

We think it would be helpful if we could personalize the "Answer required" to include field name, so it would say "Answer required for FNAME". The "Answer required" is a system message when a field is defined with the "EMPTY" parameter. We would like to modify so it shows "Answer required for" + Field.Name

5.5 Fills

We could not use field references/fills in system texts.

6. Future Research

We think we could use the ability to add a drop-down box in the template.

We are still looking forward to the masking feature for use with phone and date of birth. We also need the ability to pass parameters into executables.

The ability to use the print option from a script and produce the questionnaire output may be helpful.

New releases often bring software improvements. Updating Help contents could prove most helpful as we explore the new capabilities.

7. Conclusion

The Blaise Authoring Team Layout Group began researching options in June, 2015. We began using Blaise Version 5.0.3 Build 810. In the early stages, we ran into bugs, but being new to the software, often thinking that we were not setting things up correctly, it was time consuming to work our way through the process and conduct the research. During the course of our investigation, a number of enhancements and bug reports were submitted. The support provided was excellent. We received quick turnaround on our feedback, and a number of suggestions and enhancements have been implemented in subsequent versions.

We have made a lot of progress in our understanding of Blaise 5 layouts and templates. We have a draft Master Template that we plan to use and maintain moving forward. This template will be modified as new features are made available with the new releases of Blaise 5 software and as additional decisions are made on implementing a Blaise 5 survey.

8. Acknowledgements

The author would like to acknowledge the work and knowledge of Richard Squires, Michael Johnson, and Daniel Moshinsky as their assistance on the Census Form Prototype instruments was invaluable in researching and testing the Blaise 5 features and identifying the challenges that were presented.

The views expressed in this paper are those of the authors and not necessarily those of the U.S. Census Bureau.