CodeBuilder2 – An Initial Coding Tool

Michael D. A. Egan, Operations Research and Development Division (ORDD),
Statistics Canada

1. Introduction

In February 2003, an informal post-development review of a survey determined that authors had spent too much time working on the static text during development. Other authors, working on other surveys, agreed. Looking at the specifications and the code, it struck us that a programmatic approach was available.

As a result, we created a set of Manipula scripts that we called *CodeBuilder*: it quickly became a favourite tool of developers. We added some functionality in the summer of 2004, resulting in *CodeBuilder2*. The name *CodeBuilder* will be used throughout this paper, for the sake of simplicity.

2. Genesis

The creation of an application to collect data for a survey is a multi-divisional project at Statistics Canada. Over the last several years, there have been many reviews of how these applications are created (in varying levels of formality). We are always looking for better ways to do things.

"Authoring" (the act of creating Blaise code from a specification) is an intra-divisional task, so improving this aspect of development is an internal initiative. When we look for an internal solution, we want to develop something that: has no impact on any of our clients; uses our existing software mix, and; can be created "in our free time."

"Specifying" is an extra-divisional task. Different client divisions have different histories of specifying, and have different levels of sophistication. Some clients have a metadata system that produces specification documents and other outputs; others use Microsoft Word. Some clients produce one document per block; most produce two documents per block (one per official language). While ORDD has no input on how clients create their specifications, we have documented a standard that defines the structure of specifications.

To summarize, we had an internal initiative to reduce the amount of time an author spent on static text during initial authoring. Our solution could have no impact on how clients create specifications, and so would actually have to handle any style of specifications.

2.1 The Burden of Static Text

Just how burdensome is static text? How can something which doesn't change be a drag on development? To answer these questions, one needs to know how text gets from the specifications to the code.

Normally, each block is specified through two documents: one for English, one for French. (Things do not go smoother when the client specifies both English and French in one document.) In the Blaise code, English and French mingles and weaves. Each field has English and French definitions; each answer item in a TYPE has English and French text. The following table is a mock-up of two specification documents (English, then French) and the associated code.

Q1	**QTextE11** **QTextE12** 0-90 Q2 **QTextE21** ITextE21 <1> ATextE11 <2> ATextE12 <3> ATextE13 <4> ATextE14	Q1	**QTextF11** **QTextF12** 0-90 Q2 **QTextF21** ITextF21 <1> ATextF11 <2> ATextF12 <3> ATextF13 <4> ATextF14	Q1 "Q1@|@|^piSubjectName @/@/@BQTextE11a QTextE11b@B @/@/@BQTextE12@B" "Q1@|@|^piSubjectName @/@/@BQTextF11a QTextF11b@B @/@/@BQTextF12@B" : 0..90 Q2 "Q2@|@|^piSubjectName @/@/@BQTextE21@B @/@/@SITextE21@S" "Q2@|@|^piSubjectName @/@/@BQTextF21@B @/@/@SITextF21@S" : (V01 (1) "@SATextE11@S" "@SATextF11@S", V02 (2) "@SATextE12@S" "@SATextF12@S", V03 (3) "@SATextE13@S" "@SATextF13@S", V04 (4) "@SATextE14@S" "@SATextF14@S")

(SOD standards require that the subject's name be displayed at the top of the screen on every question. This does not need to be specified, and is handled by "@|@|^piSubjectName". QTextE11a and QTextE11b show what would happen if the specified paragraph was long. ORDD coding standards place a limit of 104 bytes on a line of code. We would never put the type definition in the FIELDS section: they would be placed in the TYPE section. It is in the FIELDS section here so that you may have an underrstanding of the work required to manipulate text strings for enumerated types.)

To get the text for one field, an author would:

- Insert the field-id and header information in the code in the Control Centre;
- switch to the English specification;
- find, highlight and copy the English text;
- switch to the Control Centre;
- paste the English text into the code;
- add font-controls (@B, @S, @U) and align the code;
- switch to the French specification;
- find, highlight and copy the French text;
- switch back to the Control Centre;
- paste the French text into the code;
- add font-controls (@B, @S, @U) and align the code;
- insert the field definition (0..90 or tYesNo, for example).

This takes four or five minutes per field for normal question text. If there were 25 fields in the block, you're looking at two hours of copying, pasting and formatting. In a smallish questionnaire, where there are 500 questions, you're looking at one solid week of copying, pasting and formatting.

We did not want to spend that amount of time on static text. We wanted a program that would handle the default field definitions (field-id, standard header), insert the English and French text, handle font controls and format the code properly.

2.2 The Benefit of Standards

It was fairly easy to write the code that creates the FIELDS statements, and that is because of our coding standards and SOD presentation standards. ORDD standards require certain parts of a field definition to be in certain areas, and that is the sort of thing a computer program can do extremely well; repeatedly; without getting bored.

SOD presentation standards require that text which the interviewer must speak aloud be displayed in bold-black, while optional text and interviewer instructions are to be displayed in bold-sapphire. There are a few other colours that are used on rare instances. *CodeBuilder* assumes that any text should be bold-black. If a particular paragraph has a different requirement, the author just has to insert the appropriate font control in the *CodeBuilder* input file.

It was not possible to write code that could detect question text within the specification. How would the code differentiate between question text and field range descriptions, or programmer notes, or answer categories? That's something that the author can do with complete accurately. The way an author organizes the input files tells *CodeBuilder* what it is dealing with (field-id, question text, edit text, etc.).

2.3 Reduction, Not Elimination

Unfortunately, *CodeBuilder* does not reduce the amount of time an author spends on static text by 100%. It reduces it by about 80%. It takes a little time to organize the specifications, about one minute per field.

When using *CodeBuilder*, the author works on text at the block level, not at the field level, and not at the answer/item level. The organization work is largely deleting text that isn't needed, which can be done very quickly. What would have taken two hours can now be done in about 20 minutes.

A completely programmatic solution would require specifications that clearly identify their various component parts (English text, French text, answer text, flow instructions, etc.). This level of functionality describes a metadata system, not a couple of Manipula scripts.

3. When *CodeBuilder* is Used

CodeBuilder is an **initial** coding aid: it is used one time, when we do initial coding.

CodeBuilder is not an **iterative** coding aid: it cannot be used to make updates to a few text items in a few places. After initial coding is done, we will continue to update specific text items the way we always have done: testers will record a text problem; specifications will change, if necessary; authors will update the specific word or phrase in the code.

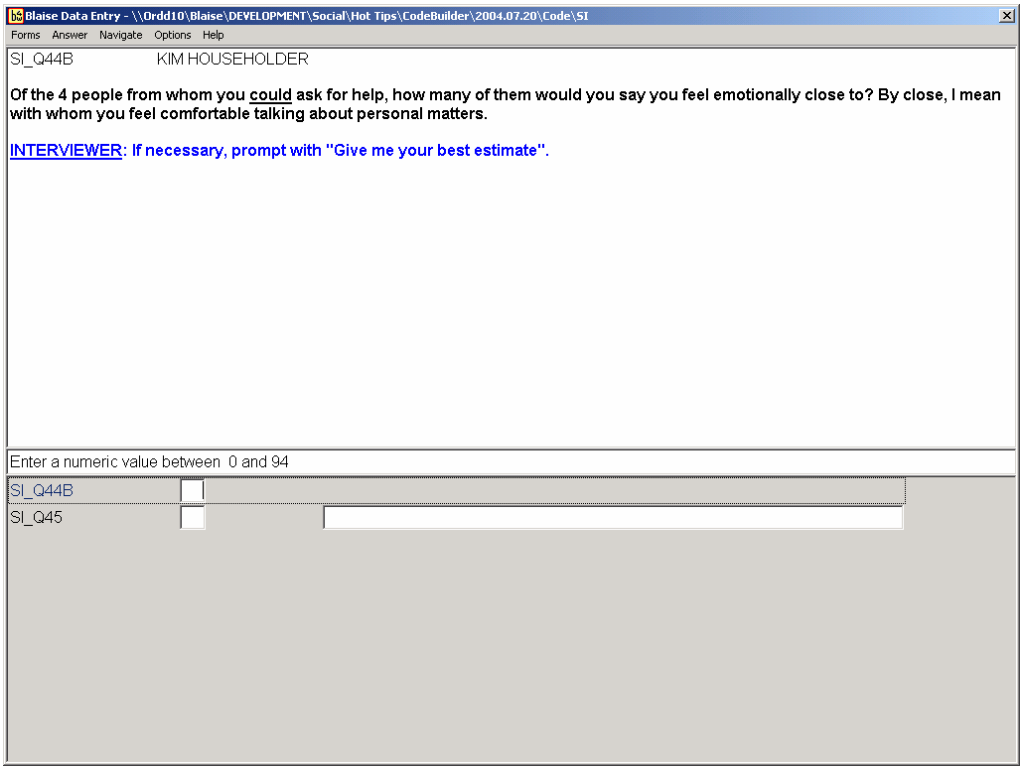
While "static text" is not supposed to change, it does. Despite all of the pre-coding reviews, mistakes will work their way into the code. There will be situations where testing will show that the specified text doesn't address all circumstances. Clients and sponsors will change their minds about the text they want. It is even possible that an author will make a mistake while using *CodeBuilder*! It is important to note that, while static text does change, it doesn't change a lot, and it is far faster to fix specific problems than it is to re-run *CodeBuilder*.

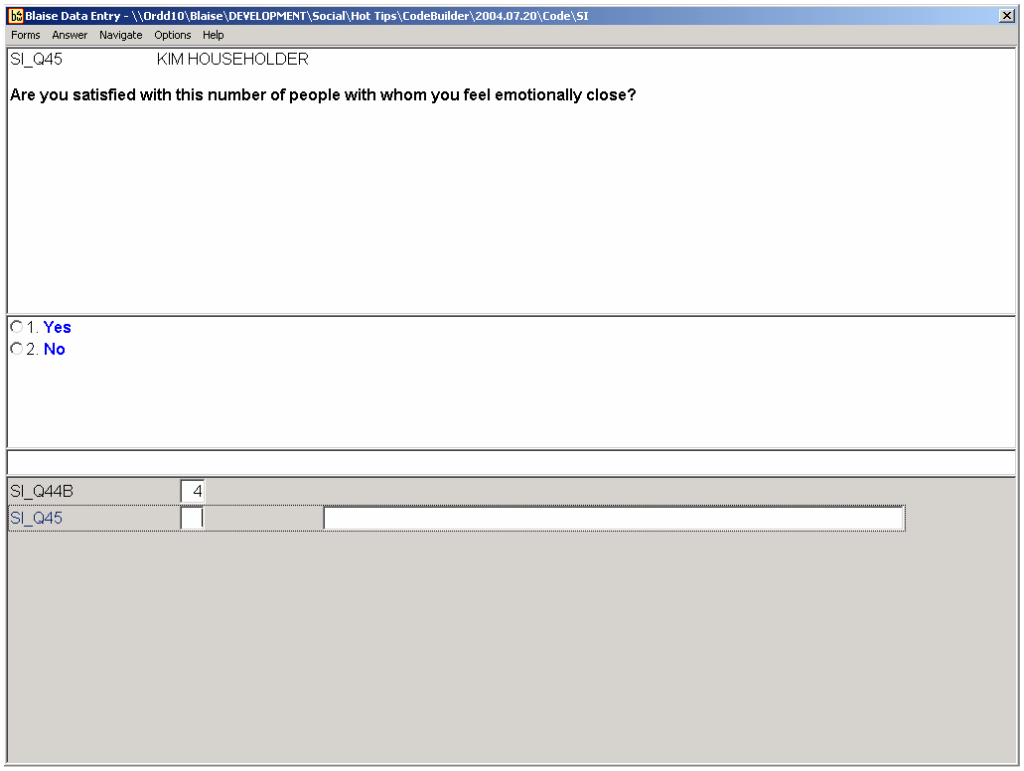
4. How *CodeBuilder* is Used

To better show how *CodeBuilder* works, some samples are included. First, we'll look at what the interviewer sees on a screen; then the specifications; then the *CodeBuilder* input files, and finally, the *CodeBuilder* output files.

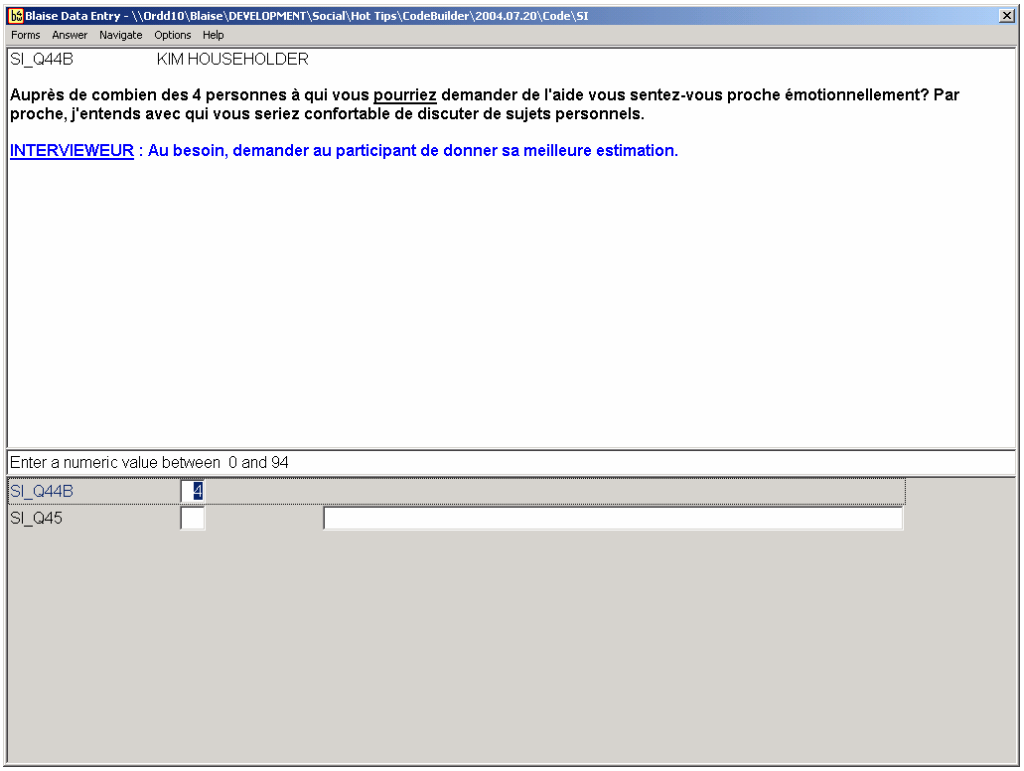
4.1 Interviewing Screens

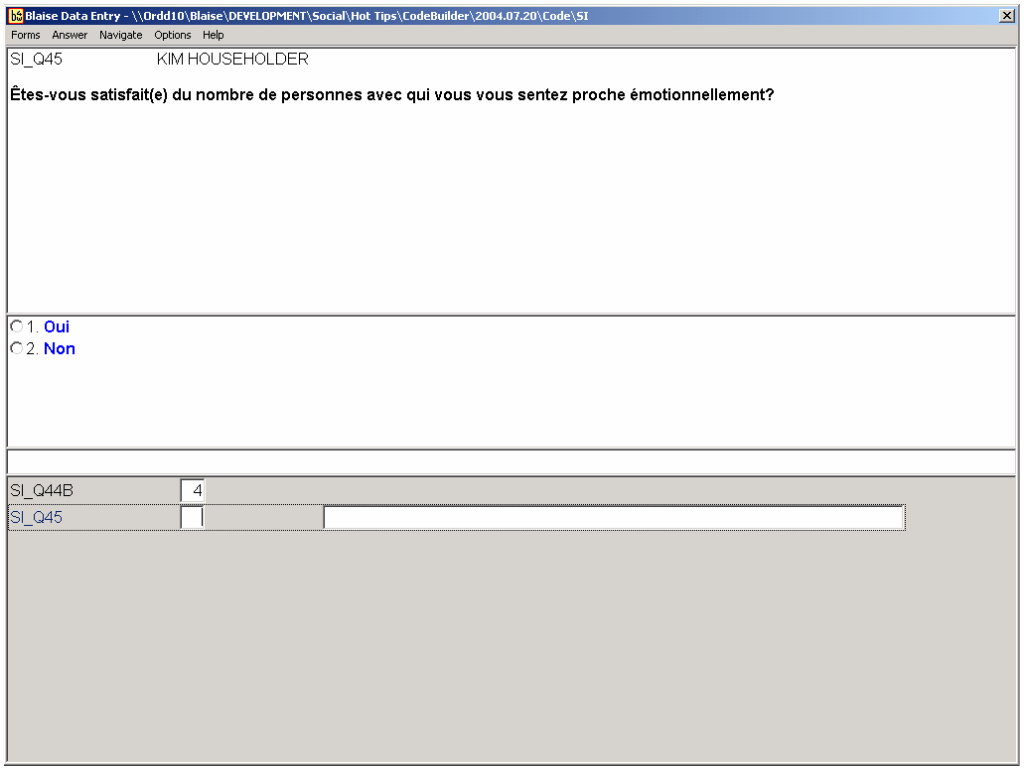
Here are two questions from one of our surveys, in English.





Here are the same two questions, in French.





4.2 Specifications

This is the English specification for those two questions.

SI_Q44B **Of the ^Numhelp people from whom you <u>could</u> ask for help, how many of them would you say you feel emotionally close to? By close, I mean with whom you feel comfortable talking about personal matters.**
<u>INTERVIEWER:</u> If necessary, prompt with "Give me your best estimate".

[Min: 0 Max: 94]

<98> Refused ……go to SI_Q46
<99> Don't know……go to SI_Q46

SI_Q45 **Are you satisfied with this number of people with whom you feel emotionally close?**

<1> Yes
<2> No
<8> Refused
<9> Don't know

As previously mentioned, SOD standards require interviewers to speak aloud text that is displayed in bold-black. ORDD specification standards say that text which is to be displayed in bold-black must be specified in bold. Text that is specified in regular font (such as the interviewer instruction) will be displayed in bold-sapphire font.

Next is the French specification for those two questions.

SI_Q44B **Auprès de combien des ^Numhelp personnes à qui vous <u>pourriez</u> demander de l'aide vous sentez-vous proche émotionnellement? Par proche, j'entends avec qui vous seriez confortable de discuter de sujets personnels.**
<u>INTERVIEWEUR</u>: Au besoin, demander au participant de donner sa meilleure estimation.

[Min: 0 Max: 94]

<98> Refus ……passez à SI_Q46
<99> Ne sait pas ……..passez à SI_Q46

SI_Q45 **Êtes-vous satisfait(e) du nombre de personnes avec qui vous vous sentez proche émotionnellement?**

<1> Oui
<2> Non
<8> Refus
<9> Ne sait pas

4.3 *CodeBuilder* FIELDS Input

In the boxes below, bold-red text represents text that the author has to add or modify during the preparation of the input files. Everything else has been copied directly from the specification.

The text for the first paragraph of SI_Q44B has been truncated because it was too long to fit on one line. In the actual input file, all of the text for it is on one line. *CodeBuilder* recognizes each new line in the input file as a new paragraph in the question text. A blank line is used as a field delimiter. The type definitions are recorded in the French input file because our code places the field definition after the French text.

```
SI_Q44B
Of the ^NumHelp people from whom you @Ucould@U ask for help, how many of them would you say you feel…
@S@UINTERVIEWER@U: If necessary, prompt with ""Give me your best estimate"".

SI_Q45
Are you satisfied with this number of people with whom you feel emotionally close?
```

```
SI_Q44B
Auprès de combien des ^NumHelp personnes à qui vous @Upourriez@U demander de l'aide vous …
@S@UINTERVIEWEUR@U : Au besoin, demander au participant de donner sa meilleure estimation.
:   0..94

SI_Q45
Êtes-vous satisfait(e) du nombre de personnes avec qui vous vous sentez proche émotionnellement?
:   tYesNo
```

4.4 Sample – FIELDS Output

```
SI_Q44B          "SI_Q44B@|@|^piRespName
                 @/@/@BOf the ^Numhelp people from whom you @Ucould@U ask for help, how many of them
                 would you say you feel emotionally close to? By close, I mean with whom you feel
                 comfortable talking about personal matters.@B
                 @/@/@S@UINTERVIEWER@U: If necessary, prompt with ""Give me your best estimate"".@S"

                 "SI_Q44B@|@|^piRespName
                 @/@/@BAuprès de combien des ^Numhelp personnes à qui vous @Upourriez@U demander de
                 l'aide vous sentez-vous proche émotionnellement? Par proche, j'entends avec qui
                 vous seriez confortable de discuter de sujets personnels.@B
                 @/@/@S@UINTERVIEWEUR@U : Au besoin, demander au participant de donner sa meilleure
                 estimation.@S"

                 :   0..94

SI_Q45           "SI_Q45@|@|^piRespName
                 @/@/@BAre you satisfied with this number of people with whom you feel emotionally
                 close?@B"

                 "SI_Q45@|@|^piRespName
                 @/@/@BÊtes-vous satisfait(e) du nombre de personnes avec qui vous vous sentez
                 proche émotionnellement?@B"

                 :   tYesNo
```

4.5 *CodeBuilder* TYPE Input

I'm introducing a new sample field at this point (row 1 below), with a longish answer list, to demonstrate how *CodeBuilder* handles TYPE text. Again, bold-red text in the input files indicate text that the author has to type during the formatting of the file; everything else was brought in during the paste from the specification.

SI_Q03 **What kind of information did you receive?**
INTERVIEWER: Mark all that apply.

<01> How to find housing
<02> How to look for a job
<03> How to access medical care services
<04> How to get language training
<05> How to find education or training
<06> How to get foreign education credentials or work experience assessed
<07> How to receive support for basic needs and services (e.g. food, clothes)
<08> How to obtain citizenship or a permanent resident card
<09> How to sponsor other immigrants
<10> How to contact immigration agencies for other business
<11> How to access other government services (excluding immigration issues)
<12> How to obtain financial and insurance products
<13> Other – Specify…….. go to SI_S03
<98> Refused ……..go to SI_Q06
<99> Don't know …….go to SI_Q06
Default Next Question: SI_Q04

@S tKindInfo
<01> How to find housing
<02> How to look for a job
<03> How to access medical care services
<04> How to get language training
<05> How to find education or training
<06> How to get foreign education credentials or work experience assessed
<07> How to receive support for basic needs and services (e.g. food, clothes)
<08> How to obtain citizenship or a permanent resident card
<09> How to sponsor other immigrants
<10> How to contact immigration agencies for other business
<11> How to access other government services (excluding immigration issues)
<12> How to obtain financial and insurance products
<13> Other – Specify

<01> Comment trouver un logement
<02> Comment chercher un emploi
<03> Comment avoir accès aux services de soins de santé
<04> Comment obtenir de la formation linguistique
<05> Comment poursuivre mon éducation ou ma formation
<06> Comment faire évaluer des attestations d'études ou de l'expérience professionnelle obtenue à l'étranger
<07> Comment obtenir du soutien pour pourvoir aux besoins fondamentaux (p. ex., aliments, vêtements)
<08> Comment obtenir la citoyenneté ou une carte de résident permanent
<09> Comment parrainer d'autres immigrants
<10> Comment entrer en communication avec les organismes d'immigration pour d'autres sujets
<11> Comment avoir accès à d'autres services gouvernementaux (excluant les questions reliées à l'immigration)
<12> Comment obtenir des produits financiers et d'assurance
<13> Autre – Précisez

4.6 Sample – TYPE Output

```
TYPE
    tKindInfo = (
        V01 (1)     "@SHow to find housing"
                    "@SComment trouver un logement",
        V02 (2)     "@SHow to look for a job"
                    "@SComment chercher un emploi",
        V03 (3)     "@SHow to access medical care services"
                    "@SComment avoir accès aux services de soins de santé",
        V04 (4)     "@SHow to get language training"
                    "@SComment obtenir de la formation linguistique",
        V05 (5)     "@SHow to find education or training"
                    "@SComment poursuivre mon éducation ou ma formation",
        V06 (6)     "@SHow to get foreign education credentials or work experience assessed"
                    "@SComment faire évaluer des attestations d'études ou de l'expérience
                    professionnelle obtenue à l'étranger",
        V07 (7)     "@SHow to receive support for basic needs and services (e.g. food, clothes)"
                    "@SComment obtenir du soutien pour pourvoir aux besoins fondamentaux (p. ex.,
                    aliments, vêtements)",
        V08 (8)     "@SHow to obtain citizenship or a permanent resident card"
                    "@SComment obtenir la citoyenneté ou une carte de résident permanent",
        V09 (9)     "@SHow to sponsor other immigrants"
                    "@SComment parrainer d'autres immigrants",
        V10 (10)    "@SHow to contact immigration agencies for other business"
                    "@SComment entrer en communication avec les organismes d'immigration pour d'autres
                    sujets",
        V11 (11)    "@SHow to access other government services (excluding immigration issues)"
                    "@SComment avoir accès à d'autres services gouvernementaux (excluant les questions
                    reliées à l'immigration)",
        V12 (12)    "@SHow to obtain financial and insurance products"
                    "@SComment obtenir des produits financiers et d'assurance",
        V13 (13)    "@SOther - Specify"
                    "@SAutre - Précisez" )
```

4.7 Sample – RULES Output

Lastly, here is an example of the RULES output that *CodeBuilder* generates. All of the fields are listed in the specified order, and edits are inserted as they are encountered.

```
SI_Q42

                {
                CHECK
                ( edit logic goes here )
                "@H@R@BSI_E42
                @/@/Please return to SI_Q36 and correct.@H@R@B"
                "@H@R@BSI_E42
                @/@/Veuillez retourner à SI_Q36 et corriger.@H@R@B"
                CHECK
                }

SI_Q43
SI_Q44A
SI_Q44B

                {
                CHECK
                ( edit logic goes here )
                "@H@R@BSI_E44B
                @/@/This number cannot exceed the number of people from whom the respondent
                could ask for help. Please return and correct.@H@R@B"
                "@H@R@BSI_E44B
                @/@/Ce nombre ne peut être supérieur au nombre de personnes sur lequel le
                répondant peut compter pour de l'aide. Veuillez retourner et corriger.@H@R@B"
                CHECK
                }

SI_Q45
SI_Q46
SI_R47
```

5. Conclusion

The *CodeBuilder* Manipula scripts described in this paper have led to a significant reduction in the amount of time authors spend on static text. *CodeBuilder* has removed a time-consuming, repetitive and boring development practice, allowing the authors to spend a larger portion of time on their 'real' task: getting the flows and edits to work properly!

CodeBuilder source code and documentation are available to any agency.