

C3B: Exploiting The Numerous Possibilities Web Technology Offers To Elevate Questions

Arnaud Wijnant/Maurice Martens - CentERdata, Tilburg University, the Netherlands

Abstract

BlaiseIS provides the possibility to administer questionnaires over the Internet. A next step would be to exploit the numerous possibilities web technology offers to elevate questions, share data sources or create extensive statistics like paradata. This can be achieved by creating special web applications that collect data and return a set of answers to the Blaise engine based on the collected data. These web applications are called custom questions.

A problem, however, is that the development of these custom questions requires different skills than programming a questionnaire in Blaise. For this reason the Custom Questionnaire Control for Blaise (C3B) was created. C3B is a mechanism which has two sides: one side which can be used by a questionnaire designer to define custom questions in a BlaiseIS questionnaire, the other side can be used by web developers to create a context-aware web application which can easily communicate with the Blaise IS engine. C3B takes care of the mappings between these two sides.

The web developer is free to choose between different development platforms (like, for example, JAVA, .NET, ASP or PHP), different techniques (like, for example, JavaScript, AJAX) and even different servers where the custom question can run on.

The questionnaire designer can easily parameterize custom questions and share them within different questionnaires, projects or even organizations.

At CentERdata, several C3B-compatible custom questions have been created. Examples of these custom questions include question-to-speech, clicking and dragging images in a certain order and auto-completion of open answers. The number of available custom questions can increase quickly since the development of a custom question is not complicated and the web offers many possibilities.

1. Introduction

CentERdata is a research institute, specializing in online survey research. CentERdata is affiliated with Tilburg University, the Netherlands. One of CentERdata's most important activities is to carry out panel surveys through online panels: the CentERpanel and the LISS panel. Both panels consist of several thousand households in the Netherlands. The members of those households complete questionnaires on a regular basis, weekly or monthly.

CentERdata started using Blaise in 2000. We intended to do so for our online surveys as well, but had to develop a software package to mold it to our needs: C2B (Weerman, 2001). With the recent improvements and needs in web technology we decided that C2B does not meet our requirements any more, and rather than redesigning C2B we decided to switch to BlaiseIS.

At CentERdata, we appreciate the advances BlaiseIS has made over the years. However, for our purposes one key feature is still missing. Sometimes, an online experiment has to be carried out that involves some advanced functionality, be it some javascript drag and drop experiment, a widget that tracks mouse

movements, a complex layout scheme or it should use extended media like movies or sound. We refer to these enriched advanced applications as web technologies.

Compared to Standard Blaise with C2B, BlaiseIS offers a more structured way of separating the questionnaire definition and presentation of the questionnaire. However, only some basic Blaise question types can be defined, which makes it impossible to have “advanced” questions without altering the program code again. The decision was made to create a more generic way to include advanced questions in BlaiseIS questionnaires. This was done by creating the Custom Questionnaire Control for Blaise (C3B). The requirements for C3B are the following:

- Questionnaire designers should be able to include custom questions in a questionnaire without the help of a programmer
- Programmers should be able to design web applications independent of BlaiseIS
- Custom questionnaires should be reusable
- No adjustment of the BlaiseIS code should be necessary to include a custom question

This paper describes how we managed to create C3B, the mechanism that managed to fulfill the requirements. Also, some examples are given that show how C3B can be used to enrich BlaiseIS questionnaires.

1.1 BlaiseIS Custom Questions

In Blaise it is possible to define question types like open questions, multiple choice questions and set-of questions. BlaiseIS transforms these different question types to a suitable HTML form which enables respondents to see a web page which looks similar to the DEP representation of a questionnaire.

On the Internet there are many more possibilities to create non-standard (custom) ways of asking questions or gather information. Some examples are:

Interactive questions

Questions that give feedback to the respondent while the respondent answers a question. A good example is auto completion. Auto completion tries to complete the text in an open answer box while the respondent is typing.

Alternative user-input questions

Respondents can use other types of input mechanisms. For example, clicking on areas in an image to answer a specific location on a map, dragging objects to locations on the screen or spoken question texts.

On other data source relying questions

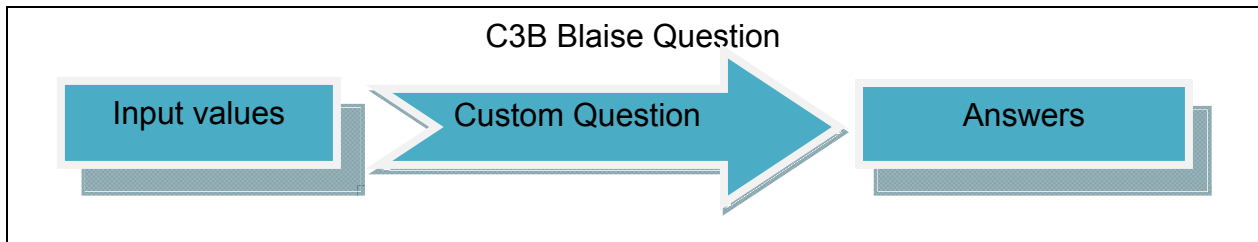
Another type of custom question is a question where the user does not need to answer a question directly, but the answer is derived from an external data source on the web. Examples are Social Media, Google maps.

In this paper, we refer to all these types of questions as custom questions.

1.2 Integrating custom questions in Blaise

Questionnaire designers should consider a C3B question as a box which can be plugged into a questionnaire. The questionnaire designer should not need to know how a custom question is programmed. He should only look at what input values are needed and what type of answer can be expected from the C3B-question. In general, a C3B question can be modeled as in Figure 1.

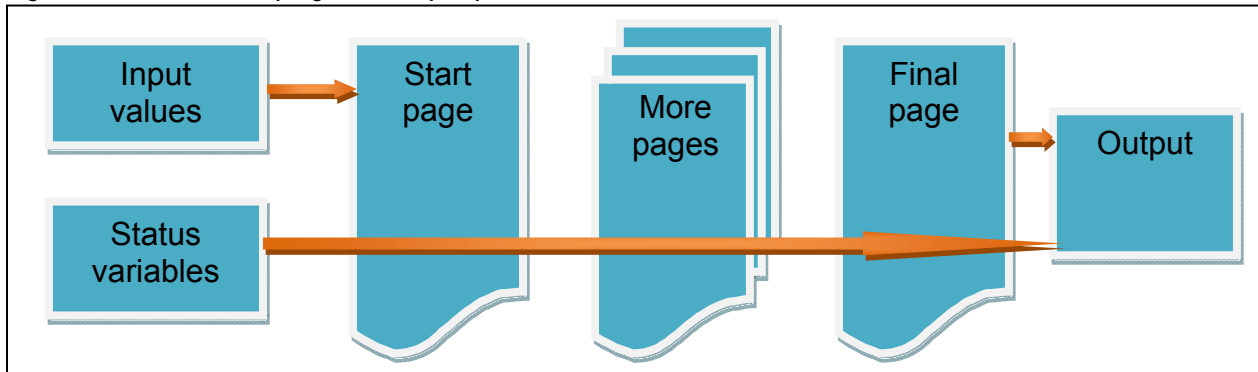
Figure 1. High level model of a C3B question



This model enables the questionnaire designer to adapt C3B custom questions easily by changing the input values. In addition, it enables the reuse of the same C3B custom question for several questionnaires or several questions within a questionnaire.

For the web programmer a C3B question looks like Figure 2.

Figure 2. C3B from a web programmers' perspective.



The input values in this model are the input values as they are defined in the questionnaire design. Together with the output, the input values take care of the communication between the questionnaire and the web application. The status variables are BlaiseIS specific status variables that will be created by the C3B software (in association with BlaiseIS). The C3B component should return the values of these variables at the end of the custom question. What happens between the start page and the final page is completely up to the programmer (even if there is a page in between, it could also be possible that the start page and final page are the same web page).

2. The process

C3B is a component between BlaiseIS and web applications. The C3B component collects data from two different sources and sends it to the web application:

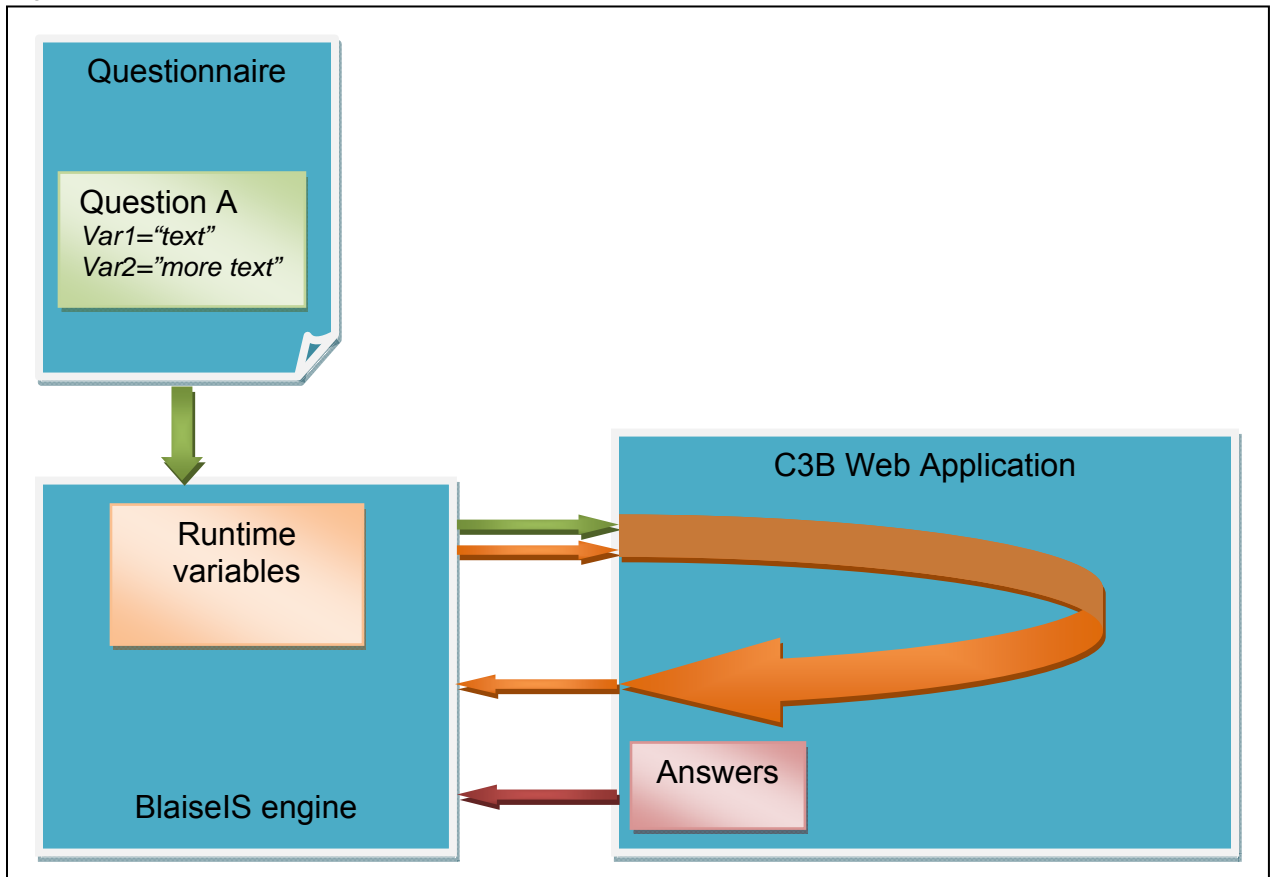
1. Variables and values as defined in the questionnaire
2. Variables and values generated by the BlaiseIS engine

An example of the first type of variable is a question text. The questionnaire designer can define a question text C3B variable and parameterize the custom question with a variable question text.

The second category contains interesting variables too. For example, the return page URL for the last page in the custom question sequence. Or the variable names of the Blaise questions and what name should be given to these variables upon the last postback action in the custom question.

An overview of how the different variables are combined with each other is shown in Figure 3.

Figure 3. The communication between the different C3B components



The C3B web application is the part of the custom question that is programmed by a computer or web programmer. There are some guidelines for the web application to become C3B compatible. These guidelines mainly consist of variables that need to be passed on or changed during the custom question web application. In Table 1, a list of these variables and what action should be done with them is displayed.

Table 1. List of the C3B web application input variables

Input variable	Action
sh	Status variable needed for the output variable "sh"
lsi	Status variable needed for the output variable "lsi"
pli	Status variable needed for the output variable "pli"
spi	Status variable needed for the output variable "spi"
aqi	Status variable needed for the output variable "aqi"
cqi	Status variable needed for the output variable "cqi"
KeyValue	Status variable needed for the output variable "KeyValue"
InterviewID	Status variable needed for the output variable "InterviewID"
Lmr	Status variable needed for the output variable "Lmr"
statusvarnameX	For x=1 to N do create an output variable that is named " <i>statusvarnameX</i> " and has the value " <i>statusvarvalueX</i> "
statusvarvalueX	See statusvarnameX
varnameX	For x=1 to N do return value for Blaise variable " <i>varblaiseVariableX</i> " as output variable named " <i>varnameX</i> " the previous value for this variable is called " <i>varvalueX</i> "
varvalueX	See varnameX
varblaiseVariableX	See varnameX
nextvarname	This is the name of the "next button" to go to the next question in the questionnaire, for example: <input type="submit" name=" <i>nextvarname</i> ">
previousvarname	This is the name of the "previous button" to go to the previous question in the questionnaire, for example: <input type="submit" name=" <i>previousvarname</i> ">
returnpage	The URL of the page where the final page should submit the answers/variables to

These variables with their values will be passed to the C3B web application by a HTTP post-action. This post-action is initialized by the browser of the respondent. Because the HTTP post-action is a very general web mechanism, some flexibility is created for the programmer. Not only is the programmer free to create the web application in the preferred web programming language (like, for example PHP, ASP,

JAVA, .NET), but also the location of the web application is not restricted with C3B. Especially this last advantage is useful to link a questionnaire to other data sources on the web.

One should however note that some browsers have their security options configured to display warnings when a page posts to a different server.

A similar variable specification exists for the questionnaire designer. These variables are listed in Table 2.

Table 2. List of the C3B questionnaire output variables

Output variable	Description
url	The URL of the C3B web application start page
serveraddress	The address of the server where this questionnaire will be loaded (NOTE: only necessary for javascript disabled browsers)
[any other variable name]	The questionnaire designer is free to define new variables, these will be posted as input variables for the C3B web application

A special C3B construction can be used to specify these variables. This C3B construction looks like a Blaise question of the type “string , empty”. But there are two differences:

1. The question text is preceded by “(customapplication)”
2. The question text contains the variables and values in the following form:
 - *variablename1*;value1;*variablename2*;value2;*variablenameX*;valueX;

In Figure 4 an example is given of how to specify these variables in a Blaise questionnaire. In this example question v1 has the output variables and values specified as mentioned in Table 3.

Table 3. The C3B output variables and values used in the first question of the Figure 4 example

Output variable	Value
url	http://.../question.php
text	Deliver meals in a nursing home
audionumber	161

Figure 4. An example of a Blaise Questionnaire that uses C3B

```
DATAMODEL siwittest
LANGUAGES = NED "Nederlands"
PRIMARY
nohouse,
nomem
FIELDS
    nohouse: 0..100000
    nomem: 0..500000
v1 (customapplication)
"url;http://cdata4.uvt.nl/blaiseistest/question.php;text;Deliver meals
in a nursing home;audionumber;161;": string, empty
c1 (hidden) "Deliver meals in a nursing home" : string
v2 (customapplication)
"url;http://cdata4.uvt.nl/blaiseistest/question.php;text;Decorate
tables in a building;audionumber;178;": string, empty
c2 (hidden) "Decorate tables in a building" : string
v3 (customapplication)
"url;http://cdata4.uvt.nl/blaiseistest/question.php;text;Fix
computers;audionumber;189;": string, empty
c3 (hidden) "Fix computers" : string
LAYOUT
    AFTER c1 NEWPAGE
    AFTER c2 NEWPAGE
    AFTER c3 NEWPAGE
RULES
    nohouse.keep
    nomem.keep
    v1
    c1
    v2
    c2
    v3
    c3
ENDMODEL
```

Please note that several variables can be part of one C3B custom question. Blaise needs to know that these variables belong to each other. This can be done by ensuring that all variables belonging to one C3B custom question are on one page (by using the NEWPAGE command in the layout section of the questionnaire).

3. Results

At CentERdata some C3B custom questions are already in use. It turned out that C3B is not only helpful to create custom questions. We also found out that complex layouts of tables in questions are sometimes easier to be generated by a C3B component than with the standard BlaiseIS functionality.

The number of variables within one C3B question varied from 1 to 16 (for example, the complex table layout contained up to 16 questions to be filled in by the respondent).

Respondents did not encounter any problems when using questionnaires with C3B.

3.1 Example

A good example of what is possible in C3B is the job interests test. This is a questionnaire that is used to advise students with some kind of a disability about what type of job would suit them best.

There are special requirements for the questionnaire since the group of respondents is not able to complete a normal questionnaire. These requirements are:

- Since not all students can read, the question should be read out by the system and the answers “like” and “don’t like” should be simple and clear images
- Since not all students understand what is meant by a profession, a question must be accompanied by a picture that shows what is done in this profession

Together with a design team the interface as shown in Figure 5 was created and programmed in PHP and Flash. Next to the visual interface the question text and answer categories are read out as soon as the question is shown on the screen.

Figure 5. A screenshot of a C3B custom question



To create a questionnaire that uses several questions of this type, a questionnaire similar to the example mentioned in Chapter 2 of this paper is used. The web programmers created a web application which used two additional parameters:

- text: the text that must be shown for this question
- audio number: an identifier for both an image and audio file per question

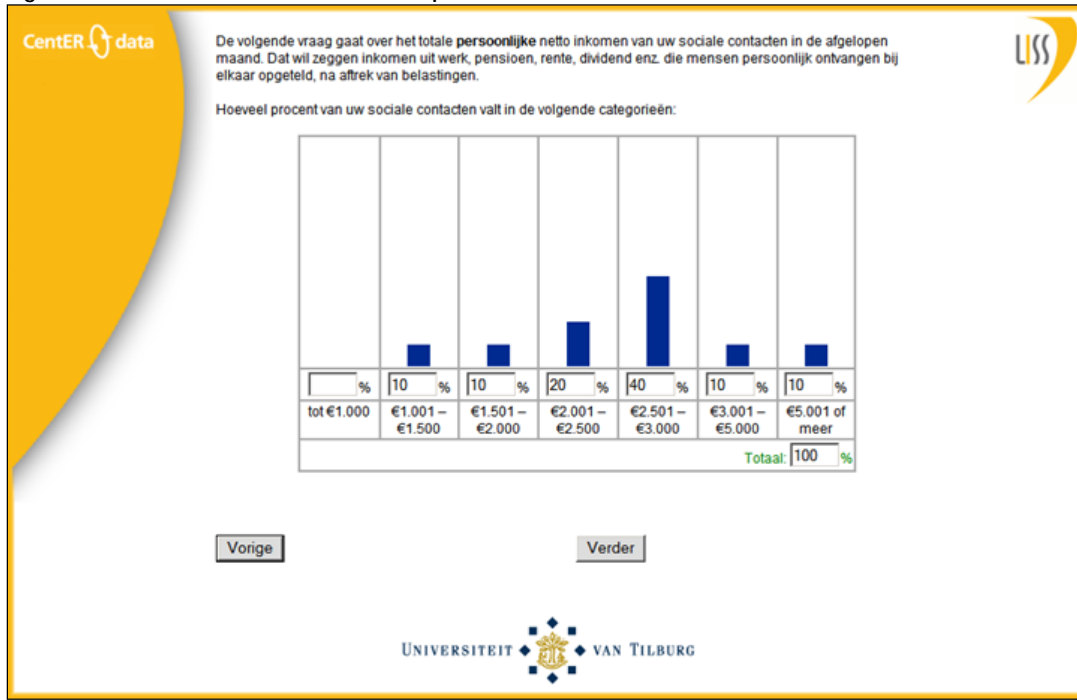
If both parameters are defined for a question, C3B and the web application take care of everything. For the questionnaire designer the questionnaire is not different from a normal BlaiseIS questionnaire with the exception that there are a few additional variables that are used to define which C3B application should be used.

3.2 Javascript examples

The C3B method has already been used in various advanced online questionnaire applications where it helped visualising percentages. When asked about mathematical or economic concepts like chances or percentages, respondents might not completely comprehend. To help them, several instruments were developed to provide a better intuitive presentation of these concepts.

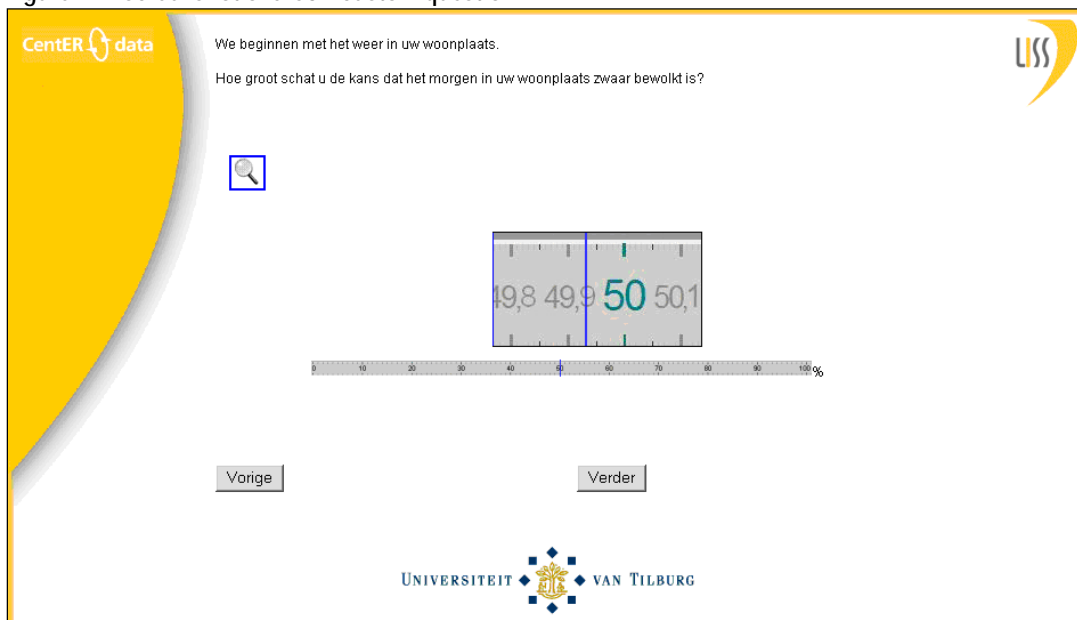
For example Figure 6 shows a screen that asks respondents to divide the persons in their social network into income groups. A javascript application was developed that visualized percentages. The bars change height when a respondent inserts integer values for each group.

Figure 6. A screenshot of a C3B custom question



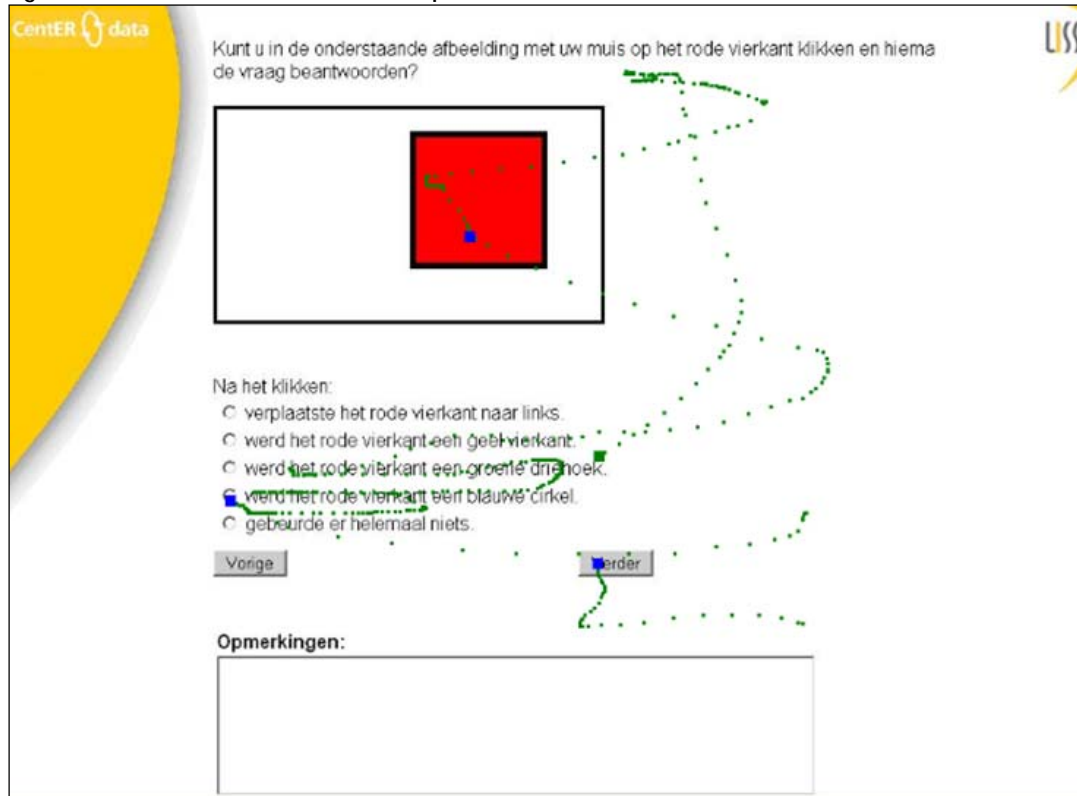
Another example is a project where respondents were asked about chances of events happening in their life, for example the chance of falling ill in the near future. To allow respondents to insert answers more significantly a scale was developed. Respondents could drag on the complete scale and specify their answers on the zoomed in window.

Figure 7. A screenshot of a C3B custom question



Javascript can also be used to check respondents' behavior. Respondents were asked to click on the red square, once clicked, this square automatically changed into a blue circle. The respondent was asked what happened. In the meantime, all mouse movements and clicks were recorded. This could be used to monitor respondent behavior, but also to test interface design. In Figure 8 a respondent's tracked mouse locations (green) and clicks (blue) are displayed over the original screen.

Figure 8. A screenshot of a C3B custom question



Other javascript examples that were developed include changing interface styles for certain groups of respondents, use of calendars, visual manipulation of graphs and puzzles.

3.3 Future development

There are several other new possibilities to create C3B custom questions. For example, social networks contain lots of data about respondents. C3B could help web programmers to link Blaise questionnaires to these data sources. Interface specifications are easier to be created, which makes it possible to create questions that measure actions of a respondent. We could for example track how a puzzle or other complex procedure was solved by a respondent. We can actually measure the respondent's actions.

Sharing components is another advantage of C3B. Reuse of C3B questions is easy and does not need involvement of a web programmer. If more projects use C3B in combination with BlaiseIS, a library of several innovative ways of asking questions in a web questionnaire could be created.

We believe improvements are still possible on the C3B interface side. At this moment we still learn from practice. For example the input and output variables can change or be expanded if questions need more

context information. Currently we think about error states and error messages and what would be the best way to provide them to the web application.

C3B is a new way of coping with web questionnaires which has lots of potential on both the improvement of the mechanism and on the usage. Its mechanism of stepping out of Blaise and returning back to it gives us the freedom to develop online questionnaires with the latest web based technologies without being hampered by third party development.

4 References

Weerman, Bas, Internet interviewing using Blaise API, IBUC 2001