Redesigning the LFS: omniode and b1-language

M. Cremers MSc (Statistics Netherlands)

1. Introduction

Statistics Netherlands (SN) has been using web interviewing as part of the data collection strategy of the Labour Force Survey (LFS) since 2012. Since then, no major changes have been made to either the questionnaire or the data collection strategy. However, times are changing. Online data collection is getting more and more important and respondents expect to be able to fill out a questionnaire on their smartphone or tablet. This requires a questionnaire that is brief, easy and simple to fill out. Furthermore, the labour market in the Netherlands is changing as well: i.e., steady jobs are less common and more people are starting their own (freelance) business. The questionnaire should be changed thoroughly to better accommodate for these new employment situations. All these new developments, plus the new IESS regulations and our new data collecting operating system (Phoenix / Blaise 5), provided the perfect opportunity for SN to redesign the LFS.

Our goal for the new LFS is:

- to design a brief, simple and clear survey
- with a design that is suitable for internet (CAWI), telephone (CATI) and face-to-face (CAPI) interviewing and which minimizes mode effects
- that the survey can be filled out on several devices such as the smartphone, tablet and PC (smartphone being the most limited device: this means a smartphone first design)
- to comply (as much as possible) to the new regulations

How are we planning to achieve these goals?
Firstly by applying SN's new survey designing method which we've called the "omnimode design". By using the omnimode design we are able to design our questionnaires faster and with less risks of mode effects. Secondly, by critically reviewing the wording we've

used in the questionnaire. The question wording should be easy to understand and written in the so-called "B1 level". And thirdly; by using a new version of software Blaise it allows to make questionnaires suitable for several devices.

2. Omnimode Design

At SN designing and programming questionnaires are two separate tasks performed by different persons. The survey designers designs the questionnaire (using Word for the question text and Visio for the routing). It concerns the wording and order of the questions and the routing. Once the wording and design of the questionnaire is finalized and approved by the client, it is handed over to the programmer who programmes the questionnaire in Blaise.

For a long time, CATI and CAPI have been the only Computer Assisted modes at SN. The survey design was therefore specifically tailored to telephone and face-to-face interviewing. In 2008 we introduced the first CAWI survey. Though we adopted an unimode principle (i.e., same questions in every mode), we quickly discovered that there were still a lot of differences between CATI/CAPI on the one hand and CAWI on the other hand. As a result, we had to make two questionnaire designs (in Word and Visio) and also two programmed questionnaires.

Working with two designs proved to be very inefficient.

- It more or less doubled our work.
  We not only had to make 2 on paper designs but also program 2 questionnaires and test 2 questionnaires.
- A lot of iterations were necessary.
  We usually started designing for the CAWI mode. Once done, we adjusted the on paper design to fit the CATI or CAPI mode. Often at this point we discovered that some questions were not suitable for CATI or CAPI and, following the unimode principle, we had to change the CAWI design as well.
- The method was error prone.
  E.g., when you make last-minute adjustments to one design and you forget to change it also in the other one.

The errors did not only affect our own work, but it also meant more work for the programmers and other departments, such as data processing, analysts etc.
In order to optimize our design process we started to think about a way of combining the three modes into ONE questionnaire suitable for all modes: omnimode. One design and one programmed questionnaire would not only save design and programming time but also reduce the risks on errors caused by mode differences. Moreover, being forced to think ahead of all three modes should lead to less last minute alterations.

Changing the on-paper design

When we started in 2015 with the omnimode design, we were not yet technically able to make a single programmed questionnaire which included all three modes. Nevertheless, with the prospect of our new data collecting system in mind, we decided that we could start by merging our on-paper designs.

Figure 1 shows an example of the on-paper design for CATI (left) and CAWI (right) that we used to work with at SN. The differences between the two designs are highlighted by different colours and shapes. Some differences were small. E.g., for the CAWI design we had to add screen breaks (purple circle). For CATI/CAPI we used the phrase "ENQ" to indicate a hint or instruction (red circle) which we had to omit voor CAWI. Some differences were merely of a technical nature: e.g. we had to use different question types for introductions (yellow square) and checks. Two differences were more challenging. First; the use of non-response categories: hidden but always present in CATI/CAPI and visible but not always present in CAWI (blue circle). Second; the interviewer rules we adopted for CATI/CAPI. Interviewers at SN are instructed to read a text aloud until they reach the question mark. This means that when answer categories have to be read aloud, the question mark should be placed after the last answer category. As a result, the question itself would end with a dot (green circle). Furthermore, we would rephrase the answer categories to form a sentence that could easily be read aloud (green square). Of course this "misuse" of punctuation marks and the rephrasing of answer categories wasn't suitable for CAWI. This was the main reason for the many differences between the modes.

Figure 1. Comparing our CATI (left) and CAWI (right) on-paper designs

How were we able to merge these two designs?

- By keeping questions and answer categories exactly the same in all modes.
  In CATI/CAPI the punctuation marks now have their "normal" function again. Answer categories that should be read aloud have a different colour: blue. We indicated this in our on paper design by the statements {Voorlees} (meaning 'Read aloud') and {/Voorlees}.
- By keeping non-response options exactly the same in all modes.
  In general we use only one non-response category "no answer". If a question is obligatory in CAWI, the hidden non-response categories in CATI/CAPI are blocked as well. Is "don't know" a legitimate answer? Than the option should be presented in all modes.
- By making smaller changes: the "ENQ" phrase has been omitted, pages breaks are added, "soft checks" in CATI/CAPI are replaced by questions etc.

Figure 2 shows our solution: our new "omnimode" design.

Blok: Achtergrondkenmerken hulpbehoevende

{NewPage}

AantPPHlp
U heeft eerder aangegeven dat u ($1: op dit moment hulp geeft $2: in de afgelopen 12 maanden hulp heeft gegeven) aan familie, een vriend of buur vanwege beperkingen. Dit wordt ook wel mantelzorg genoemd. Over die hulp volgt nu een aantal vragen.

Aan hoeveel mensen ($1: geeft u op dit moment hulp $2: heeft u in de afgelopen 12 maanden hulp gegeven)?
We bedoelen geen hulp in het kader van uw werk of als vrijwilliger.
[1..9]
NORF

{NewPage}

IntroKeuzePers
Kies de persoon die u ($1: op dit moment de meeste hulp geeft $2: het meeste heeft geholpen in de afgelopen 12 maanden) en beantwoord de rest van de vragen alleen voor deze persoon.
($T:>>Toets <1> om verder te gaan.<<
$W: >>Kies 'verder' om verder te gaan.<<)
1. Verder   [Verder]
NORF

{NewPage}

RedGnHlp
U heeft eerder aangegeven dat u op dit moment niet meer helpt.
Wat is de reden dat u deze persoon niet meer helpt?
>>Meerdere antwoorden mogelijk.<<
SET[]
{Voorlees}
1. Deze persoon is overleden   [Overled]
2. Deze persoon had minder of geen hulp meer nodig   [MindHlp]
3. Beroepskrachten zijn meer gaan helpen   [Beroeps]
4. Andere mantelzorgers zijn meer gaan helpen   [AndMant]
5. Om financiële redenen   [Finan]
6. Deze persoon is opgenomen   [Opgenom]
7. Andere redenen   [Anders]
{/Voorlees}

Figure 2. Omnimode design

Our first experiences with the omnimode design were very positive; both the questionnaire designers as the questionnaire programmer found the design process took less time. And though we still have two datamodels, they are now much more comparable than in the past. We also evaluated the new interviewer rules with our CATI and CAPI interviewers. We had some concerns they might have problems with the answer categories that are now not rephrased anymore, sounding perhaps more staccato. 97,5% of the interviewers indicated they did not have any problems working with the new

interviewer rules. Some indicated that they even preferred the new method because it was more clear to respondents that the answers provided were options they could choose from.

Were we able to make the questionnaires exactly the same?

No, there will always be mode differences that can't be overcome. For example; in CAWI we sometimes use grids while we can only use single items in CATI/CAPI. Or because in CATI/CAPI you hear information while in CAWI a respondent has to read the information. Differences are either indicated by mode specific routing, by adding mode-specific text imputations ($W, $T, $P) or by adding specific lay-out instructions for the programmer.

Towards an omnimode programmed questionnaire (Phoenix / Blaise 5)

Currently we are working on a new CATI and CAPI channel and a new interface. This will allow us to programme questionnaires in an omnimode style, which is also the plan for the new LFS. Different style sheets will allow us to present the same questions differently in the different modes (e.g., coloured answer categories for CATI/CAPI) and for different devices. We are also hoping to further optimize our omnimode design: e.g. by making the hidden non-response categories in CATI/CAPI visible as well.

3. B1 Language level

Another important focus point of SN is to improve the readability and comprehensibility of our communication products, including our questionnaires. Governmental institutions usually have a reputation of using incomprehensible formal language. Often this is done to convey authority, expertise and professionalism. However, formal language often works counterproductive.
When people find it too burdensome to read an invitation letter they will not even participate. And when they do respond but do not understand the questions, it will not provide the data we are looking for. This will all have a negative impact on the quality of the data. Fortunately, the realisation has set in that simple and clear questions in a less formal style are much more effective. This not only holds for the less educated people. Clear and simple questionnaires are much more pleasant (and quicker) to fill out for higher educated respondents as well. As the emphasis on web interviewing increases, clear and simple questionnaires (which can be filled out without the help of an interviewer) are becoming an absolute necessity.
The Common European Framework (CEFR) distinguishes 6 levels to determine someone's linguistic ability: A1, A2, B1, B2, C1 and C2. A1 is the most elementary level, C2 the most advanced. It is decided that the language level of the questionnaires at SN should be 'B1' at the most. This is the level that can be understood by almost 80% of the Dutch citizens.

Our new LSF is designed with the following "B1-principles" in mind:

- Use only words that are common and have a literary meaning (avoid jargon!)
- Use short sentences (max. 10 to 15 words) and short paragraphs
- Use colloquial language instead of formal written language
- Use active verbs over passive verbs
- Present the most important message in a question first
- Phrase an instruction as a question. This will trigger a respondent and makes it more likely that he/she will read the instruction.

4. Redesigning the LFS

In my presentation at the workshop I will illustrate the omnimode principles further with LFS related examples. I will also share some examples of how we've learned from past mistakes and hope to make our new LFS "B1-proof".