

# Conjoint Experiments with Multiple Questions in Qualtrics

*Kim Sass Mikkelsen*

*V-1-1 Wednesday, October 19, 2016*

## Conjoint experiments in Qualtrics

Conjoint experiments are becoming increasingly popular in political science, not least after the introduction of user friendly software to design (Strezhnev et al. 2014) and analyse (Hainmueller et al. 2014) them. This opens the opportunity for researchers to implement conjoint experiments so long as they are sufficiently familiar with R to use Hainmueller et al.'s *cjoint* package and sufficiently savvy to use Strezhnev et al.'s very simple stand-alone *conjointSDT* software.

In a conjoint respondents are presented with multiple *tasks* comparing typically two *profiles* each featuring *values* of a number of *attributes*. Figure 1 shows an implementation from an experiment I ran with Jan Meyer-Sahling and Christian Schuster in Chilean central government. The image shows a table and two subsequent questions (the study in fact has four such questions).

In the table, each row is an attribute, columns marked **Funcionario** are profiles, and each cell in each profile contains the values for an attribute for that profile. The two questions below are questions posed to respondents, asking them to choose one profile over the other (in this case, which profile will be easiest to motivate to work hard and which profile is more loyal to a government's policy program).

## The problem this note addresses

The conjoint experiment in figure 1 is implemented in Qualtrics and is one of four similar tasks. In each task, new profiles are presented to respondents and the same questions are asked. There is a lot of randomisation involved in this implementation to avoid design effects.

1. Attribute order is randomised for a respondent but consistent across tasks
2. Values are randomised for each profile in each task
3. The order of the two questions are randomised for a respondent but consistent across tasks

A continuación se presentan sucesivamente cuatro comparaciones de perfiles hipotéticos de funcionarios de su institución. Por favor observe los perfiles y luego responda a las preguntas, basado en su propia experiencia dentro de su institución.

Primera comparación (de cuatro):

	Funcionario 1	Funcionario 2
<b>Relación contractual</b>	Contrata	Contrata
<b>Remuneración relativa</b>	Superior a colegas en funciones similares	Comparable a colegas en funciones similares
<b>Año de ingreso al cargo</b>	2015(Actual Gobierno)	2010(Gobierno del Presidente Piñera)
<b>Modo de ingreso</b>	Contratación Directa	Concurso Público
<b>Educación</b>	Educación Media	Educación Media
<b>Estamento</b>	Administrativo	Administrativo

¿Qué funcionario sería más fácil de motivar para trabajar más allá del horario laboral, en pos de cumplir con una tarea urgente?

Funcionario 1



Funcionario 2



¿En qué funcionario confiaría más para implementar políticas públicas de acuerdo al programa de gobierno de turno?

Funcionario 1



Funcionario 2



Figure 1: Conjoint from a study of Chile's central government

Fortunately, Strezhnev et al.'s *conjointSDT* software takes care of 1. and 2. However, this software does not implement 3. And herein lies the problem. In most conjoint experiments, respondent are presented with multiple tasks. Qualtrics can randomise the order of questions below the conjoint table. But it will do this for every task, meaning respondents are presented with the same questions beneath similarly looking tables but in changing order. This is inelegant and may harm consistency, response rates, or respondent satisfaction.

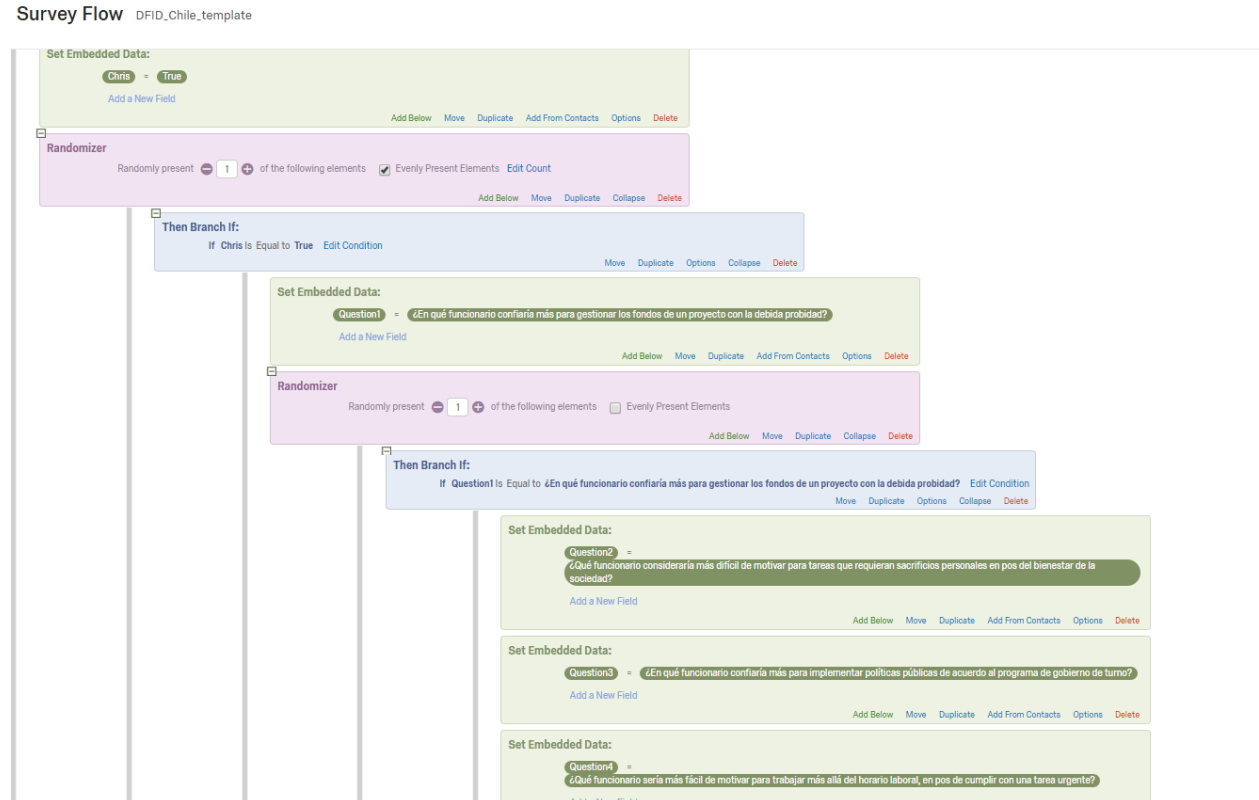


Figure 2: Previously suggested survey flow from the Chile study

Of course, Qualtrics has a hotline and they are very helpful in this respect. When presented with the problem by a colleague, they advised building the randomisation into the survey flow. Figure 2 shows a small fraction of the necessary survey flow. The details of what is going on are not important. What is important that the screen dump in figure 2 is but part of a much longer flow. This setup creates two problems:

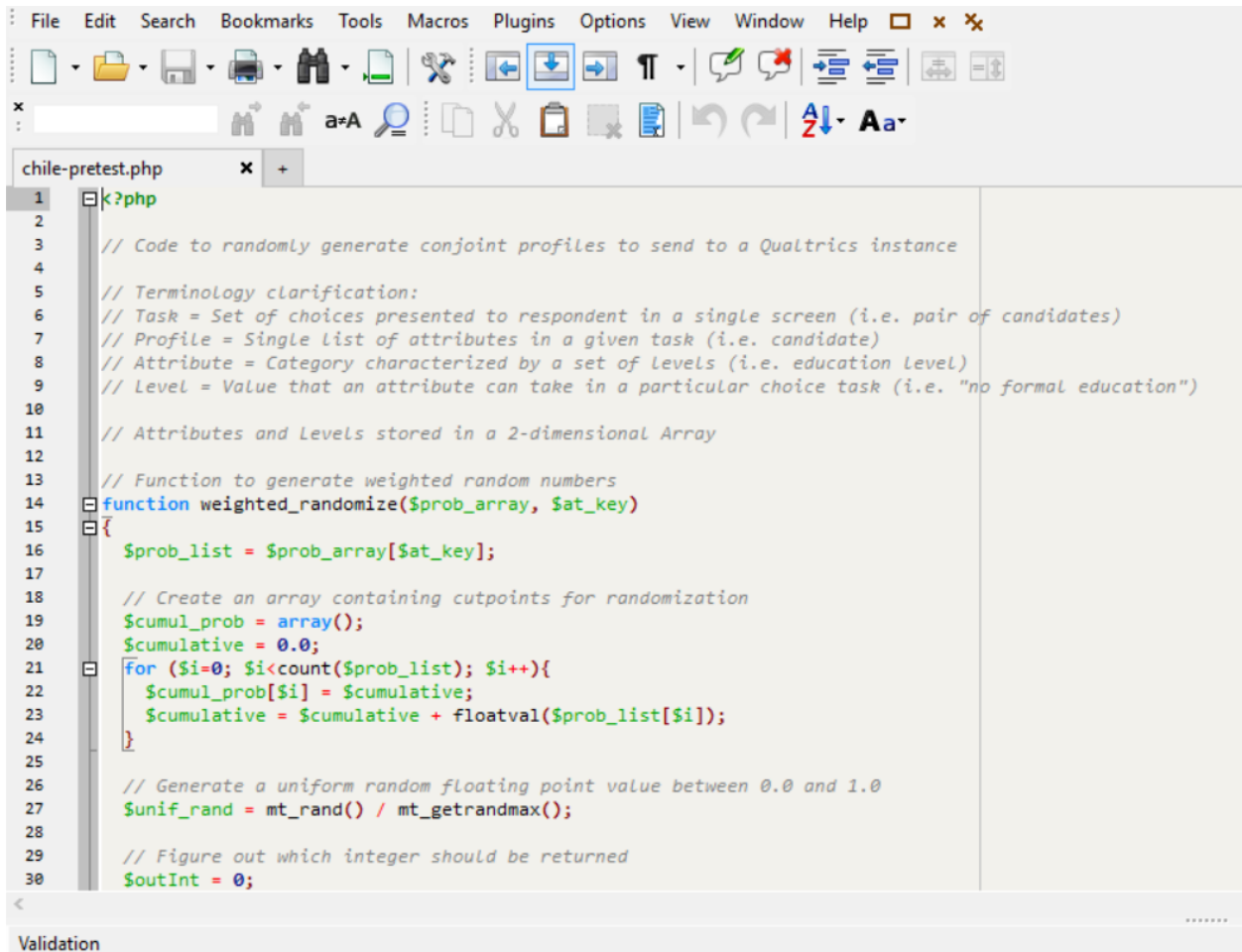
1. It is very strenuous to set up, requiring *a lot* of pointing and clicking
2. *Any* change to question formulation after the flow is set up requires numerous replacements of identical text snips in the flow, again by *a lot* of pointing and clicking

I wrote this note for people who face this problem but are not happy spending hours pointing and clicking,

for instance because a pre-test required a minimal change in a question formulation.

## A simple solution

The simple solution I came up with relies on two elements that implementation of conjoint experiments in Qualtrics using *conjointSDT* already requires. The first element is the PHP created by the *conjointSDT* software. This file needs to be changed slightly. To do this, one needs an editor (or notepad but it gets messy). Figure 3 shows the PHP from the Chile study opened in *Synwrite* (but any editor will do).



```
File Edit Search Bookmarks Tools Macros Plugins Options View Window Help
// Code to randomly generate conjoint profiles to send to a Qualtrics instance
// Terminology clarification:
// Task = Set of choices presented to respondent in a single screen (i.e. pair of candidates)
// Profile = Single list of attributes in a given task (i.e. candidate)
// Attribute = Category characterized by a set of Levels (i.e. education level)
// Level = Value that an attribute can take in a particular choice task (i.e. "no formal education")
// Attributes and Levels stored in a 2-dimensional Array
// Function to generate weighted random numbers
function weighted_randomize($prob_array, $at_key)
{
    $prob_list = $prob_array[$at_key];
    // Create an array containing cutpoints for randomization
    $cumul_prob = array();
    $cumulative = 0.0;
    for ($i=0; $i<count($prob_list); $i++){
        $cumul_prob[$i] = $cumulative;
        $cumulative = $cumulative + floatval($prob_list[$i]);
    }
    // Generate a uniform random floating point value between 0.0 and 1.0
    $unif_rand = mt_rand() / mt_getrandmax();
    // Figure out which integer should be returned
    $outInt = 0;
```

Figure 3: The beginning of the PHP from conjointSDT (in Synwrite)

The change that needs to be made to this file is slight. At the end of the file write (here demonstrated for three questions as if the questions were “first question”, “second question”, and “third question”).

```

$conjointquestions = array("first question","second question","third question");

shuffle($conjointquestions);

$returnarray["Question1"] = $conjointquestions[0];
$returnarray["Question2"] = $conjointquestions[1];
$returnarray["Question3"] = $conjointquestions[2];

```

The first line creates an array with the three questions (if you are unfamiliar with PHP do note the dollar sign and the semicolon). The second line randomises the order of the questions using a pseudo randomisation built into the PHP language. The remaining lines adds the randomised questions to the output array for *conjointSDT* (called *returnarray* by default) as “Question1”, “Question2”, and “Question3” (note that the first element of an array is indexed 0 rather than 1).

## Survey Flow DFID\_Chile\_template

The screenshot shows a configuration window for a 'Web Service' element. At the top, there is a 'URL' field containing 'http://christianschuster.net/chileres.php' and a 'Test URL' button. Below the URL field is a 'Parameter to Web Service...' field followed by an equals sign and a 'Value...' field, with a red minus sign and a green plus sign to its right. A checkbox labeled 'Fire and Forget' is located below the parameter field. Underneath is a section titled 'Set Embedded Data' which contains a list of 12 rows. Each row consists of an 'ED' field, an equals sign, a value field, and a red minus sign with a green plus sign to its right. The values in the value fields are: F-1-1, F-1-1-1, F-1-2, F-1-1-2, F-1-3, F-1-1-3, F-1-4, F-1-1-4, F-1-5, F-1-1-5, F-1-6, and F-1-1-6.

Figure 4: Web service element for conjoint experiments

Adding more questions is easy. Add a fourth question after the third before the parenthesis separated by a comma. And add a new line stating `$returnarray["Question4"] = $conjointquestions[3]`. More questions can be added similarly.

The second element required by the solution is the **Web service** element that is embedded in the survey flow. Figure 4 shows the beginning of this element for the Chile study.

A continuación se presentan sucesivamente cuatro comparaciones de perfiles hipotéticos de funcionarios de su institución. Por favor observe los perfiles y luego responda a las preguntas, basado en su propia experiencia dentro de su institución.

Primera comparación (de cuatro):

	Funcionario 1	Funcionario 2
<code>#{e://Field/F-1-1}</code>	<code>#{e://Field/F-1-1-1}</code>	<code>#{e://Field/F-1-2-1}</code>
<code>#{e://Field/F-1-2}</code>	<code>#{e://Field/F-1-1-2}</code>	<code>#{e://Field/F-1-2-2}</code>
<code>#{e://Field/F-1-3}</code>	<code>#{e://Field/F-1-1-3}</code>	<code>#{e://Field/F-1-2-3}</code>
<code>#{e://Field/F-1-4}</code>	<code>#{e://Field/F-1-1-4}</code>	<code>#{e://Field/F-1-2-4}</code>
<code>#{e://Field/F-1-5}</code>	<code>#{e://Field/F-1-1-5}</code>	<code>#{e://Field/F-1-2-5}</code>
<code>#{e://Field/F-1-6}</code>	<code>#{e://Field/F-1-1-6}</code>	<code>#{e://Field/F-1-2-6}</code>

`#{e://Field/Question1}`

Funcionario 1 Funcionario 2

---

`#{e://Field/Question2}`

Funcionario 1 Funcionario 2

Figure 5: The conjoint table and subsequent questions in Qualtrics

In the web service element, a website is provided where Qualtrics can access a PHP array to get information. By default, the conjointSDT software labels attributes *F-task-attribute* and values *F-task-profile-attribute*, meaning F-1-1 is the first attribute in the first task and F-1-1-1 is the value of the first attribute for the first profile in the first task. These values can be set as embedded data automatically using the *Test URL* button in Qualtrics.

After changing the PHP, uploading it, and clicking Test URL, the *conjoint-question* values should appear

and can be included in the survey flow as embedded data. Once this is done, the conjoint questions can be randomised by setting up each task in Qualtrics as shown in figure 5.

Importantly, the *Question1* and *Question2* embedded datafields do not change from task to task, whereas the content of the table for the second task should use F-2-1-1 rather than F-1-1-1, and so on. With that, randomisation is set up but remains consistent across tasks.

## **At the end**

I hope this note will be helpful for those who wish to avoid both question order effects and unhappy respondents with conjoint experiments with multiple questions in Qualtrics. If you found it helpful, please do cite the note as:

Mikkelsen, Kim Sass. 2016. *Conjoint Experiments with Multiple Questions in Qualtrics*. Research Note. Odense: University of Southern Denmark