

## Coding Qualitative Data: Working with a Team of Coders

Here we present a series of steps for working with a team to develop a codebook and complete a thematic coding analysis of qualitative data. We propose a step-by-step process that allows for modifications to accommodate different sized teams, complexities of datasets and timelines for coding. While teams can benefit from reviewing and revising the steps as their coding progresses, having an outlined schedule can be beneficial to ensure consistency and understanding at the onset. It should be noted that this process can be multi-directional, with previous steps revisited as many times as needed. Coding teams should meet as many times as the team feels is needed. Depending on the size and expertise of your coding team, you may need one or many meetings in steps five and six to discuss progress, interpretations and new codes.

<p><b>Step One:</b> Build a team</p>	<p>The number of coders on your coding analysis team can be determined by considering many different factors, including: relevant discipline expertise to analyse the data; availability of different members to meet proposed timeline; quantity of data (length of transcripts, amount of interviews); complexity of study topic; inclusion of interviewers as coders; and inter-rater reliability.</p>
<p><b>Step Two:</b> Establish an organized framework of codes</p>	<p>The study research question(s), objectives, as well as each team member's knowledge of the project, can serve as an overall framework for the codebook. It is important for the team to set out with an understanding of what is being researched, current literature on the topic of interest, the research question(s) to be answered and the diversity of the participants.</p>
<p><b>Step Three:</b> Code initial interviews independently to create preliminary codes</p>	<p>Each member of the coding team ('coder') should review a different set of no less than two transcribed interviews to generate initial codes. Each coder should:</p> <ol style="list-style-type: none"> <li>be assigned a unique subset of transcripts to review and code;</li> <li>read through each transcript (first read) to familiarize themselves with the responses (do not make any notes during this step);</li> <li>read through each transcript (second read) while thinking and making notes on the evaluative statements and critical commentary the participants are making, as well as objectives or other large points that are being addressed (these are <b>MACRO-CODES</b> or <b>PARENT NODES</b>);</li> <li>read through each transcript again (third and fourth reads) for more specific themes that breakdown the responses even further (these are <b>MICRO-CODES</b> or <b>CHILD NODES</b>).</li> </ol>
<p><b>Step Four:</b> Coding team meeting to review preliminary codes</p>	<p>After each coder independently comes up with a list of preliminary codes, a team meeting is held to review the range and similarities amongst the codes. Definitions for each code and parameters for its application are created with the input from all coders. By the end of the meeting (or series of meetings), a draft codebook is created that includes macro- and micro-codes, along with definitions for each code. This codebook should be viewed as a <i>draft</i> since codes can later be added, broadened (e.g. in definition) or combined when the interviews are coded.</p>
<p><b>Step Five:</b> First application of codebook to the data</p>	<p>Transcripts assigned to generate initial codes, should be re-assigned at this stage so they can be coded based on the developed codebook. If new codes are necessary, the coding team should meet to review, approve and add the new codes to the codebook and discuss any discrepancies with the use of codes. Depending on preferences and available resources, your team may choose to document coding using computer software (e.g. NVivo), a word processor (e.g. Microsoft Word) or hard copies of the transcript. When using NVivo, an electronic coding structure is created from the codebook and is used to organize the data. If you are using NVivo, all coding should be done in the software at this stage.</p>
<p><b>Step Six:</b> Subsequent application(s) of revised codebook to the data</p>	<p>After initial coding of the dataset, as well as addition and modification of codes to the codebook, transcripts should be reviewed and coded again so changes or additions may be made to earlier interviews as well. This step can be repeated as many times as needed. Coding is an iterative process and only the coders can determine when coding is 'complete' (or close to it).</p>

## Codebook layout

The layout of your codebook can take different forms depending on the length and level of detail of the codes, as well as the preference of team members. Alongside your definitions, it is helpful to provide examples of text that would go in the code. These examples can be generated while developing the codebook and coding the first transcripts. Depending on the complexity of your codes and number of codes, a quick reference definition along with a more detailed definition may be useful for coders.

Language Services Toronto Evaluation Project Codebook			
#	Code	Definition	Descriptions and examples of poi
1	Medical Comprehension	Conversations and conveyance of information relevant to medical encounter and medical terminology.	- Medical Terminology - Patient Understanding - Interpreter Understanding - Provider Communicating
2	Ease of Use	Specific to the structure and design of the Language Services Toronto program. This will include references to how easy/difficult using the LST program is and critiques of its design.	- Clear series of steps - Initial use - Service provider weaknesses - Use of interpretation become - Ease of access to the service
3	Ease of using an interpreter as a result of LST	Specific to using OPI. References to how easy/difficult it is to use specifically an OPI that is a part of the LST program, as a result of the design of the LST program.	- Program satisfaction - Ease in scheduling - Flexibility - Interpreters Readily available

## Inter-rater reliability

When engaging in team coding analysis for the purpose of data reduction, consistency amongst coders is important. Inter-rater reliability (IRR) is a way of assessing how well independent coders can agree with the rest of the team, in their coding of data. IRR can be done at any point in steps four to six, and should be done as often as the team sees fit. Generally, the more revisions a codebook has and/or the more discrepancies during team meetings around discussions of codes and their application, can indicate how often IRR should be assessed.

A practice for IRR is to select a subset of transcripts that cover a variety of: time periods in data collection, revisions in interview protocol, interviewers, participant demographics and transcript lengths. Each interview transcript in this subset should be assigned to at least two different coders; meaning more than one coder is coding each of the selected transcripts. When overlapping the transcripts amongst coders, make sure each coder has a different coding partner for each transcript they are assigned. For example, the table below shows the assignments of a subset of 6 transcripts to be coded by four coders (A, B, C, D):

Interview Transcript	Assigned Coders	
1	A	C
2	C	D
3	D	A
4	B	C
5	D	B
6	B	A

## Tips for team coding analysis

- **Set deadlines for each step** so individual team members stay on track and the team can move towards the projected timeline together.
- **Create a quick reference list of codes** that can go along with the codebook, to provide each coder with a glance of the inventory of codes.
- **Designate a team member** as a main point of contact for coding discrepancies, proposed new codes, NVivo updates, transcript assignments and other tasks, so communication is centralized and therefore more organized.
- **Document** process, coding decisions, transcript and team meetings including dates, to maintain transparency of coding methods and to have a record of analysis to be used in reports and articles.
- **A shared folder** on an institutional network or an online application that still complies with privacy requirements can make all coding documents easily accessible and organized for team members.
- **Meet as often as needed** to address discrepancies in coding, meanings of codes, additional codes and to revisit the analysis objectives and hypotheses.
- **Always keep in mind that coding is an iterative process** where only the team can determine when data analysis is truly complete or close to it. Codebooks and coded transcripts can be re-visited and re-coded as many times as needed.
- **Set clear terms of reference and boundaries** from the beginning to facilitate communication and productivity. The terms should include how discrepancies will be dealt with if they arise.

## Additional resource

Burnard, P. (1991). A method of analysing interview transcripts in qualitative research. *Nurse Education Today*, 11(6):461-466.

CRICH Survey Research Unit – Providing high quality and efficient research and evaluation services to the health and social sciences community.

**Contact information:**  
<http://sru.crich.ca>  
[SRU@smh.ca](mailto:SRU@smh.ca)



© Attributions are to be made to:  
 Devotta, K; Pedersen, C; CRICH Survey  
 Research Unit Methodology Bits

**CRICH**  
 Centre for Research on Inner City Health  
 Survey Research Unit