1.0 Analysing Open-Ended Survey Question Data in CAQDAS Packages: Data Preparation for QDA Miner (v 3.2)

The following material provides step-by-step guidance for preparing the texts, collected from a large number of respondents answering several open-ended questions in a survey situation, for qualitative analysis using QDA Miner software.

There are several stages in this procedure, some of which may not be relevant in your circumstances. Some users may have alternative methods or short-cuts, in which case please use your own judgment as to which elements from below to apply. Our purpose here is to provide a comprehensive guide, which has been tested and proved to work, for the benefit of those who have not achieved this task successfully before.

Unlike some other CAQDAS packages, QDA Miner was designed with this sort of task in mind and so the steps outlined below do not represent a "workaround" but a mainstream activity.

Outline

1.1 Select the software to use in an alternative in an intermediate process.
1.2 Locate and organise all of the data to be used in the analysis in a logical structure.
1.3 Import all of the data into QDA Miner in a single operation.
1.4 Check the accuracy of the import procedure.

Detailed Steps:

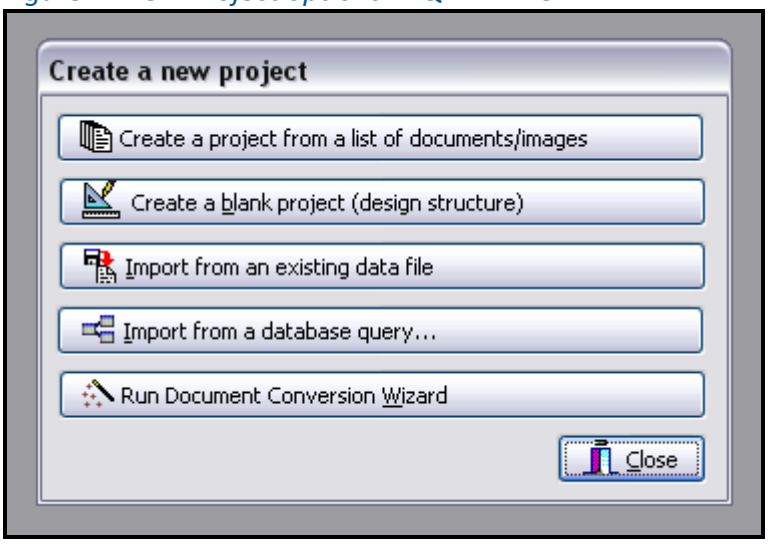
1.1 Select the software to use in an alternative in an intermediate process.

QDA Miner can import data in a wide range of formats so there is considerable flexibility over this step.

TIP: Before you do any data preparation it may be a good idea to explore the possibilities available to you.

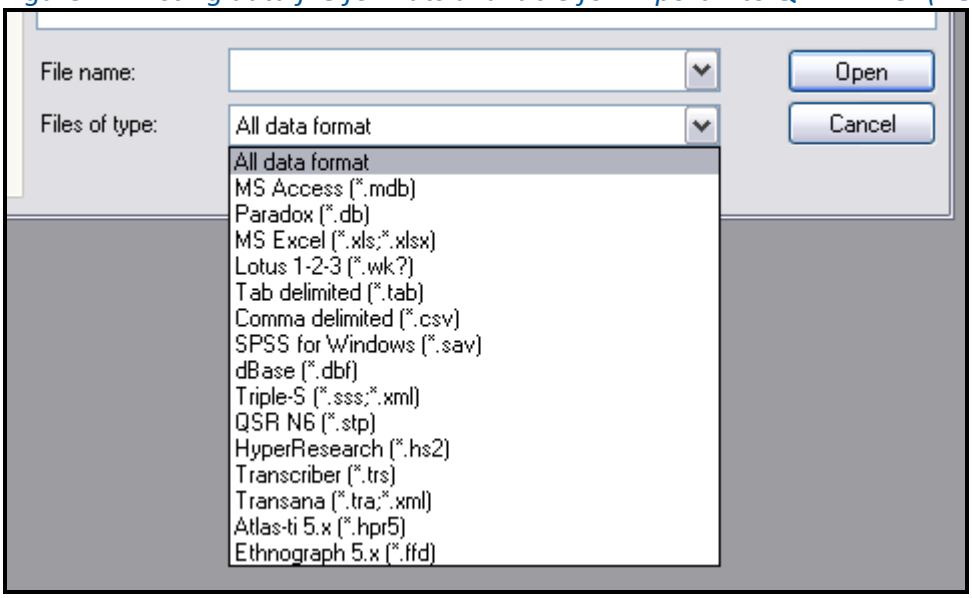
Start the QDA Miner program and, at the first dialogue screen, select the *Create a new project* option. This takes you to the dialogue screen shown in Figure 1 below.

Figure 1: New Project options in QDA Miner



It is most likely that the third option above will be useful to you, so explore that one by selecting it first – *Import from an existing data file*. Move straight to the drop-down menu beside the field *Files of type:* near the bottom of the dialogue box and click on the down arrow to display the full list of data formats which can be imported directly into the program. Figure 2 shows the list in version 3.2.2 of the software at the time of writing (July 2009), you may have different possibilities available in your version.

Figure 2: Existing data file formats available for import into QDA Miner (v 3.2.2)



If your data is already set up in one of these formats then that will be the option to explore first, otherwise you should consider which of these formats might be the most convenient to use as an intermediate step between your existing data format and QDA Miner. Obvious choices to consider are the spreadsheet options (Microsoft Excel or Lotus 1-2-3) or the statistical program (SPSS for Windows).

TIP: For this purpose it is most unlikely that you would want to use one of the other CAQDAS programs listed here as they generally have more difficulty handling open-ended survey question data, those options are made available for converting other types of project which have been started in another software package.

Two of these options will be discussed in the sections below, Microsoft Excel and SPSS, as it is considered that these are the ones most users of this website are likely to use. There is some guidance in the QDA Miner help file which may also be worth consulting.

Note also an alternative option from the dialogue in Figure 1, *Create a project from a list of documents/images*. This is the route that many conventional qualitative analysis projects would take, whereby groups of documents are imported into a QDA Miner project and variable data about their cases is added later.

TIP: It is not recommended that this option is used where you have a large number of cases each making brief comments because this situation is much more efficiently addressed by the methods explained below.

1.2 Locate and organise all of the data to be used in the analysis in a logical structure.

We will look at the spreadsheet method first because this is probably the least software specific method available. This will be illustrated with reference to Microsoft Excel, although any other spreadsheet program can be expected to work in a similar fashion and no specialised functions are required.

The recommended structure to aim for is one in which you arrange the data in a table, or grid, with a separate row for each respondent (or case). The columns should hold both the texts of the answers to the open ended questions and the variable data about each respondent which may be relevant to the analysis.

TIP: Before carrying out any processing of the data, try to examine it in its most original format in order to identify some of the longest individual responses and to look for unusual characters in the texts. The longer responses may get truncated in some conversion processes (for example on being brought into SPSS if the 256 character default was not changed) and it is useful to be able to check that you have the fullest versions of these before you carry out the analysis work. Unusual characters, other than basic alpha-numeric and punctuation ones, sometimes affect conversion processes so it is a good idea to check that these have been copied faithfully before doing analysis work.

It may be easier to understand if you keep the variable information in the left-hand columns and put the question response texts to the right. There should be no blank rows or columns in the table at the point of transfer to QDA Miner, so check for that problem from time to time. (If you have a blank row the import process will create an empty case for it, and a blank column will become an empty variable in QDA Miner). When you import the data into QDA Miner the program will add a sequential case identifier number of its own to each case but, if you anticipate wanting to relate some data from this analysis to other materials stored outside the program, you may wish to retain your existing case identifiers as a string variable – no specific format is required for these. The first row of the table should carry the variable names and there should be no duplications amongst these.

TIP: It will be possible to import further variables or sets of response texts at a later stage, so the decision as to what to include at this point is not an irrevocable one. However it is worth making sure that you use all of the data that you can reasonably expect to consider during the analysis, as it is easier to incorporate it at this stage.

See more discussion and advice about selecting variables for inclusion in the transfer to CAQDAS on this page.

An illustration of a spreadsheet prepared for data transfer to QDA Miner is shown in Figure 3. In this example column A has been used to store an ID for each respondent, columns B to H hold demographic and other variable data about the respondents, and columns I onwards hold the response texts. Note how the formatting of columns I to K includes *Word Wrap* and *Autofit rowheight* functions so that the longer texts can be seen in full. In this presentation it is difficult to see if there are any blank rows so it can be useful to turn this formatting on and off for different viewpoints.

Figure 3: Microsoft Excel Spreadsheet prepared for import into QDA Miner

	A	B	C	D	E	F	G	H	I	J	
1	ID	sex	age	area	work	flowarn	where	tenure	QMORE	QADVI2	QBETT2
2	Resp 04402	M	age18\24yr	Worces	Student	flNwN	Notfl	Prp05\10yr			
3	Resp 04403	F	age55\64yr	Worces	Retired	fYwY	Outer	Prp20\99yr		leaflet from the environment agency	
4	Resp 04407	F	age18\24yr	Worces	Wrkfull	fYwN	Outer	Prp03\05yr			
5	Resp 04408	F	age35\44yr	Worces	Wrkfull	fYwN	Outer	Prp05\10yr	step by steps approaches "in Resp of flooding" to do this that or the other i.e. A checksheet		
6	Resp 04409	M	age55\64yr	Worces	Wrkfull	flNwY	Notfl	Prp03\05yr			
7	Resp 04410	M	age55\64yr	Worces	Retired	fYwN	Outer	Prp20\99yr			
8	Resp 04417	M	age55\64yr	Worces	Wrkfull	flNwN	Notfl	Prp20\99yr			
9	Resp 04418	F	age25\34yr	Worces	Ukghome	flNwY	Notfl	Prp01\03yr			Things th
10	Resp 04419	M	age65\74yr	Worces	Retired	flNwY	Notfl	Prp01\03yr			
11	Resp 04420	F	age45\54yr	Worces	Wrkfull	flNwN	Notfl	Prp20\99yr			
12	Resp 04421	F	age65\74yr	Worces	Retired	flNwN	Notfl	Prp10\20yr			
13	Resp 04422	M	age55\64yr	Worces	Wrkpart	fYwN	Outer	Prp10\20yr			
14	Resp 04423	M	age45\54yr	Worces	Wrkfull	fYwN	House	Prp20\99yr			
15	Resp 04424	F	age65\74yr	Worces	Retired	fYwN	House	Prp20\99yr	needed someone to advice of help available p she heard nothing at all		
16	Resp 04428	F	age35\44yr	Worces	Wrkfull	flNwN	Notfl	Prp05\10yr			
17	Resp 04429	M	age35\44yr	Worces	Wrkfull	fYwN	Outer	Prp05\10yr			
18	Resp 04443	F	age65\74yr	Worces	Retired	flNwY	Notfl	Prp05\10yr	knowledge of electricity is dangerous for elderly people and they needed to know what to do		
19	Resp 04447	F	age75\99yr	Worces	Retired	fYwY	Outer	Prp20\99yr			
20	Resp 04448	M	age75\99yr	Worces	Retired	flNwY	Notfl	Prp20\99yr			
21	Resp 04449	F	age65\74yr	Worces	Wrkfull	fYwY	Outer	Prp10\20yr			
22	Resp 04454	F	age65\74yr	Worces	Retired	flNwN	Notfl	Prp20\99yr	something! Respondent had no advice at all		
23	Resp 04455	F	age55\64yr	Worces	Retired	flNwY	Notfl	Prp20\99yr		given leaflets explain what we should do	
24	Resp 04457	M	age55\64yr	Worces	Wrkfull	flNwN	Notfl	Prp20\99yr			
25	Resp 04806	M	age35\44yr	Woodfo	Selfemp	fYwY	Outer	Prp05\10yr		either to evacuate or to move upstairs, turn electricity off	Regular h issue sar
26	Resp 04807	M	age45\54yr	Woodfo	Retired	fYwY	Outer	Prp10\20yr		flood pack, storing water, food and turning off electricity	

TIP: QDA Miner will distinguish between socio-demographic variables and texts for analysis on the basis of the number of characters in the data cells. It is best to keep the variable labels short, say less than 30 characters.

Having prepared the spreadsheet carefully, save it to a location where it can be easily located for importing into QDA Miner.

As an alternative to working through a spreadsheet program like Microsoft Excel above, it may be possible to take data straight from SPSS for Windows to QDA Miner. At Figure 2 above you can see that

the format "SPSS for windows (*.sav)" is an option. However it is not necessarily a problem free operation and some preparation may be needed for this. In particular the frequency with which new versions of SPSS are issued increases the possibility of incompatibilities arising in such a transfer.

TIP: As a preliminary point, if you have a very large data set in SPSS it may be worth filtering off a modest sized sample of cases and variables into a temporary file and exploring how well it is imported into QDA Miner. If any problems are identified with this subset of the data you will save yourself time by not processing large amounts of data fruitlessly. When you have proved that a set of procedures works successfully for the sample, you can apply them to the full dataset with confidence.

It is also recommended that you look carefully at some of the longest responses in your dataset to see if they have been truncated in SPSS by a default character number limit. If possible check back to an earlier source of the data before it was copied into SPSS.

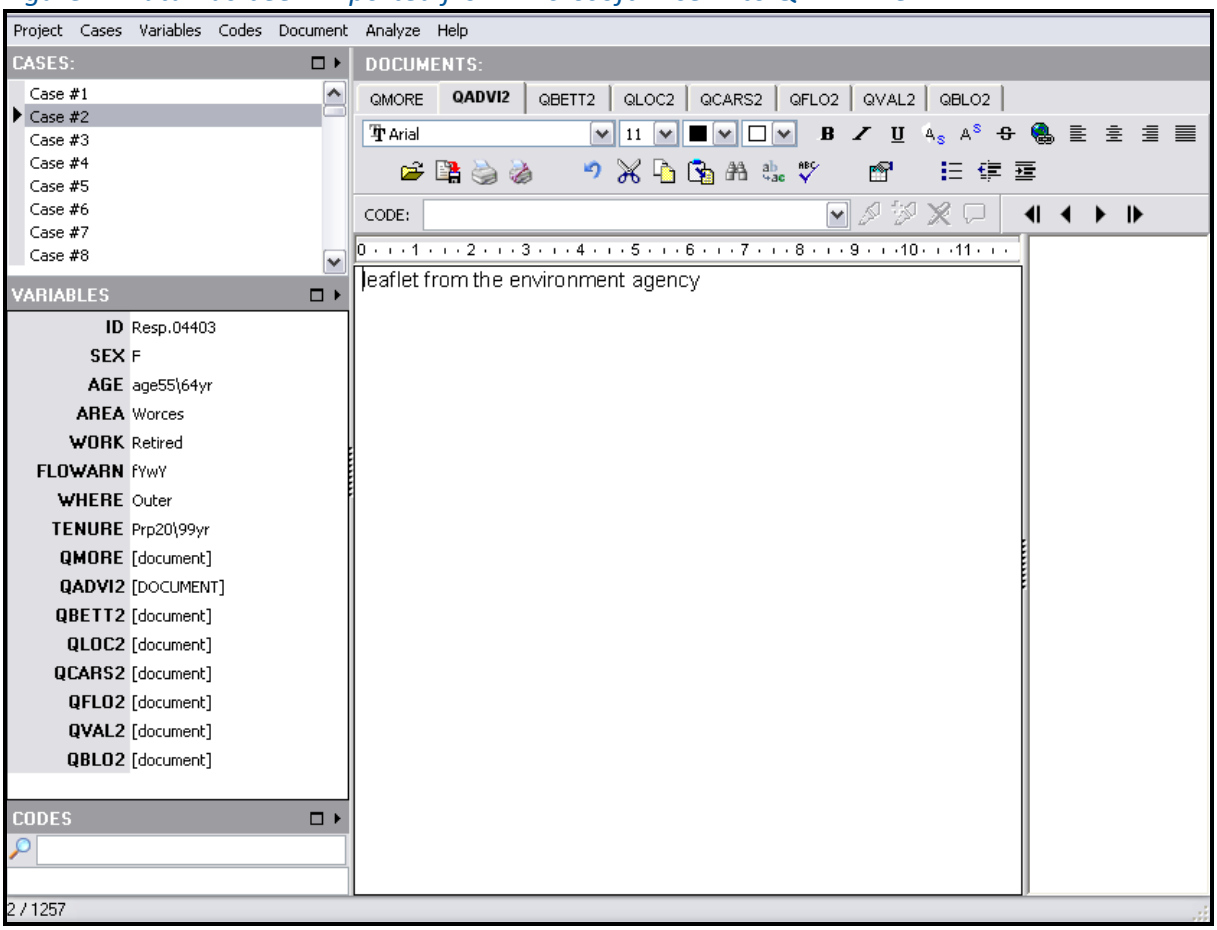
1.3 Import all of the data into QDA Miner in a single operation.

If necessary, open a new project and, as shown in Figure 1, select the option *Import from an existing data file*. You will have to change the *Files of type:* field to the right setting for the program you have used in the above preparation step.

If you are importing from Microsoft Excel, select that from the pull-down menu (see Figure 2) and then navigate to the location where you saved your input file in the previous step, select it and click on the "Open" command. The dialogue screen changes subtly as you are asked to provide a filename for the new project. Next you will be asked whether you want to import a whole worksheet or a specified range. It is likely that you do want to import the whole sheet so leave the default range on "All", otherwise specify the appropriate range for the data table, and click on "Import". If in doubt, click on cancel and check your source spreadsheet before starting the import process again.

TIP: You can observe the progress of the import as the screen shows a count of the number of cases imported. As a further control check, on completion of the routine the bottom left hand corner of the working window shows 1 / #### (where #### stands for the total number of cases imported – in Figure 4 this appears as "2/1257", the 2nd case out of 1,257 is highlighted), so you can confirm that the number of cases you expected has been imported successfully.

Figure 4: Data has been imported from Microsoft Excel into QDA Miner



One way of interpreting the desktop layout illustrated in Figure 4 is that some elements of the spreadsheet layout have been retained. The cases are held in rows, as shown with the column of case numbers in the Cases window. The documents are held in columns, as shown by the row of tab labels at the top of the Documents window. But only one cell's contents are visible at a time – so the text "leaflet from the environment agency" which can be seen in Figure 4 is the response by Case #2 (highlighted) to question QADVI2 (emboldened tab label). From this position, moving the highlighter line down the Cases window will bring other responses to question QADVI2 into view in the Document window, and clicking on other tab labels will bring Case #2's responses to other questions into view.

TIP: Note in Figure 4 how QDA Miner has added its own case identifiers ("Case #2" is highlighted) and shows the imported identifiers as the first variable ("ID Resp.04403" in this example).

1.4 Check the accuracy of the import procedure.

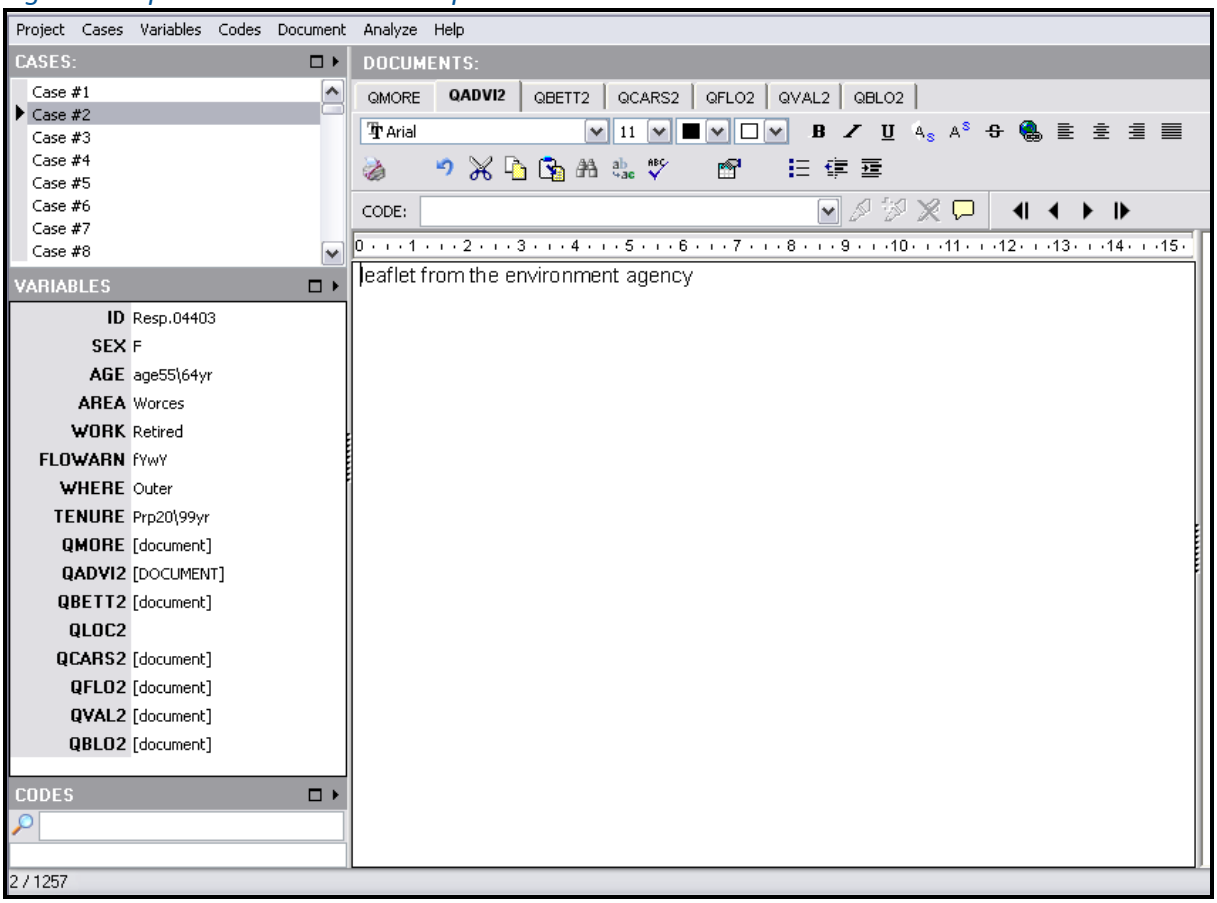
Having confirmed that the correct number of cases is showing at the bottom of the screen, you should next check the number of variables. The Variables panel to the left of the main window can be enlarged to help display a long list of variables. The list of variables should be in the same sequence as the columns in your source spreadsheet, and this should aid checking its accuracy.

A significant point is that the text variables, which contain your response texts, should have received different treatment from the demographic variables. In the Variables panel these items should show

the word "[document]" where the other variables show the appropriate value for the current case (which is highlighted in the Cases panel above), and these document variables should also be visible as tab labels at the top of the Documents panel. This is illustrated in Figure 4 above, which shows the situation just after a data import from an Excel spreadsheet.

However, on some occasions the import process does not quite work correctly for all document variables. In the example shown in Figure 5 below, one text variable has not been fully recognised as a document during the process. Compare Figure 5 with Figure 4 and note two significant differences.

Figure 5: A problem with a data import



The data import shown in Figure 5 has not been fully successful because the document called "QLOC2" has not been fully recognised as a text document. This is apparent in the Variables panel where the space to the right of the variable name is blank, and also in the Documents panel where there is no tab showing "QLOC2".

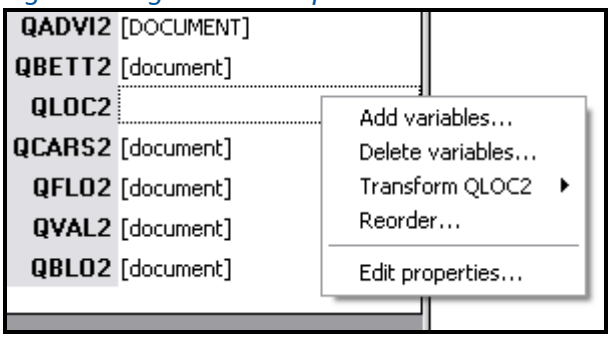
The best way to rectify this situation is to adjust the setting in the *Project / Program Setup* command for "Import as document text longer than:" to a low value (but greater than the length of your longest socio-demographic variable), say 50 characters, and then re-import your data into another new project.

Alternatively, in this situation the problem can be resolved quickly with the following procedure:

- In the Variables panel, left Click in the empty space to the right of the document name (a dotted field outline appears).

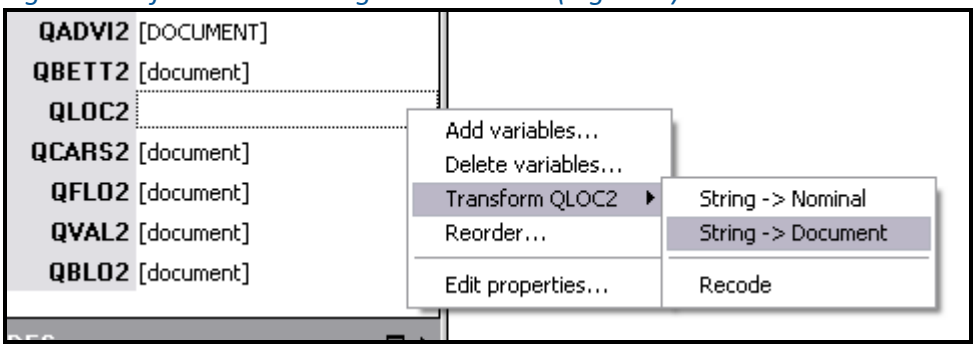
- Right Click in the same place to open a menu (Figure 6).

Figure 6: Right click to open a menu



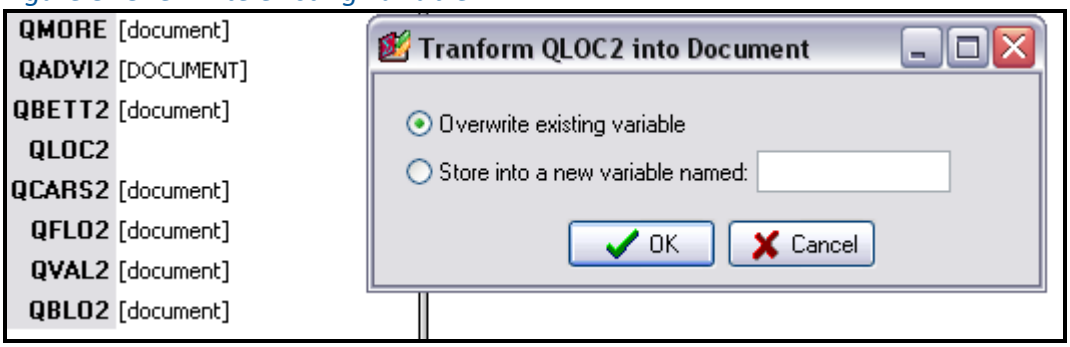
- Left Click on "Transform xxx" in the menu (Figure 6 – xxx will be replaced by the name of your variable, this is context sensitive so do not proceed if this is showing the wrong name).
- Left Click on "String -> Document" (Figure 7).

Figure 7: Left click on "String > Document" (Figure 7).



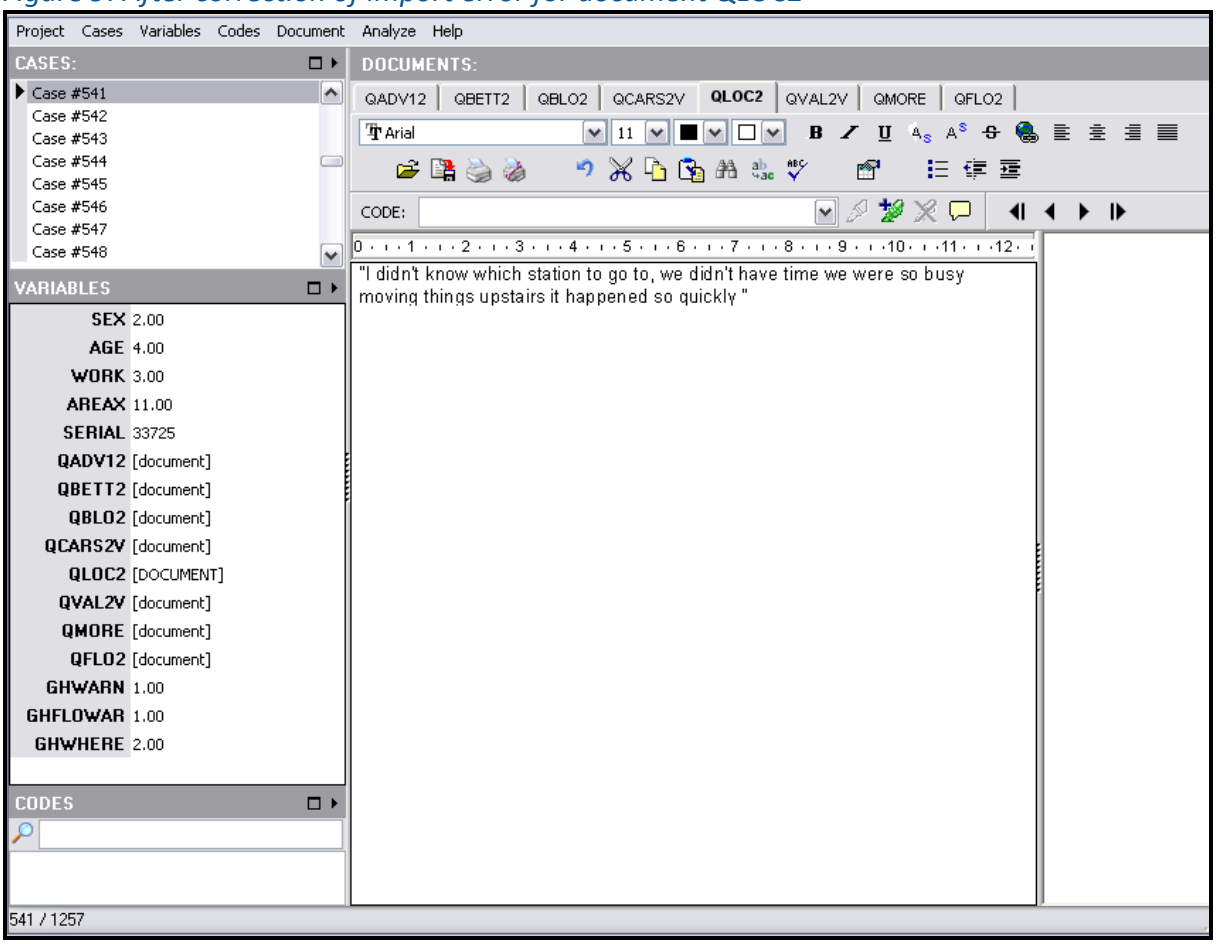
- At the dialogue shown in Figure 8 leave the default on "Overwrite existing variable" and click on OK.

Figure 8: Overwrite existing variable



After a brief time of processing, dependent on the size of the document, the word "[document]" should appear beside the affected variable name and a new tab should be inserted in the appropriate place at the top of the documents panel – see Figure 9.

Figure 9: After correction of import error for document QLOC2



TIP: Note also that the sequence in which the variables appear can be changed, by using the "Reorder" command which is visible in Figure 6 above. It makes sense to have the socio-demographic variables at the top of the list, where they will usually be visible, and the text variables at the bottom, where they can be safely obscured when a larger panel for codes is needed.

You may also notice that the word "document" sometimes appears in the Variables panel in capital letters, and sometimes in lower case letters. The capital letter version indicates that there is a response text for that document from this respondent, whereas the lower case version indicates that there was no response. If you look at the data in the Simstat module you will observe a similar pattern with "TEXT" and "Text".

Finally, to check the accuracy with which the texts have been imported it is recommended that you generate some Text Retrieval reports where you can verify that the longest responses have been imported successfully. To view all of the responses for one question use the command *Analyze/Text Retrieval*, select the document with the drop-down menu by *Search in:*, set the Search unit to *Paragraphs* and click the radio button by *Retrieve all units*, then hit *Search*. On the Search Hits results page, click the check box by *Multilines grid* in order to see the longer responses wrapped over multiple lines (you can adjust the size of the window and the width of the text column to see more material if

you like). You could check a specific text by clicking on the appropriate case number and document tab but the re-numbering of cases during the data import process may have made locating a particular respondent more difficult.

2.0 Analysing Open-Ended Survey Question Data in CAQDAS Packages: Initial Coding Approaches for QDA Miner (with particular reference to WordStat)

There are, of course, many different ways to analyse responses to open-ended questions. This page is not a step-by-step guide on how to do analysis, it is rather a series of observations about how the features of QDA Miner and its additional WordStat module might interact with a particular type of dataset. This page should be read in the context of the related materials concerning the use of QDA Miner for the analysis of open-ended survey question data, accessible from the main Analysing Survey Data page.

The tools discussed below are illustrated with examples from the same post flooding event survey that was used to illustrate the data preparation processes. This data is characterised by a fairly large number of short statements.

Summary:

Details:

2.1 Reading the texts – by respondent or by question?

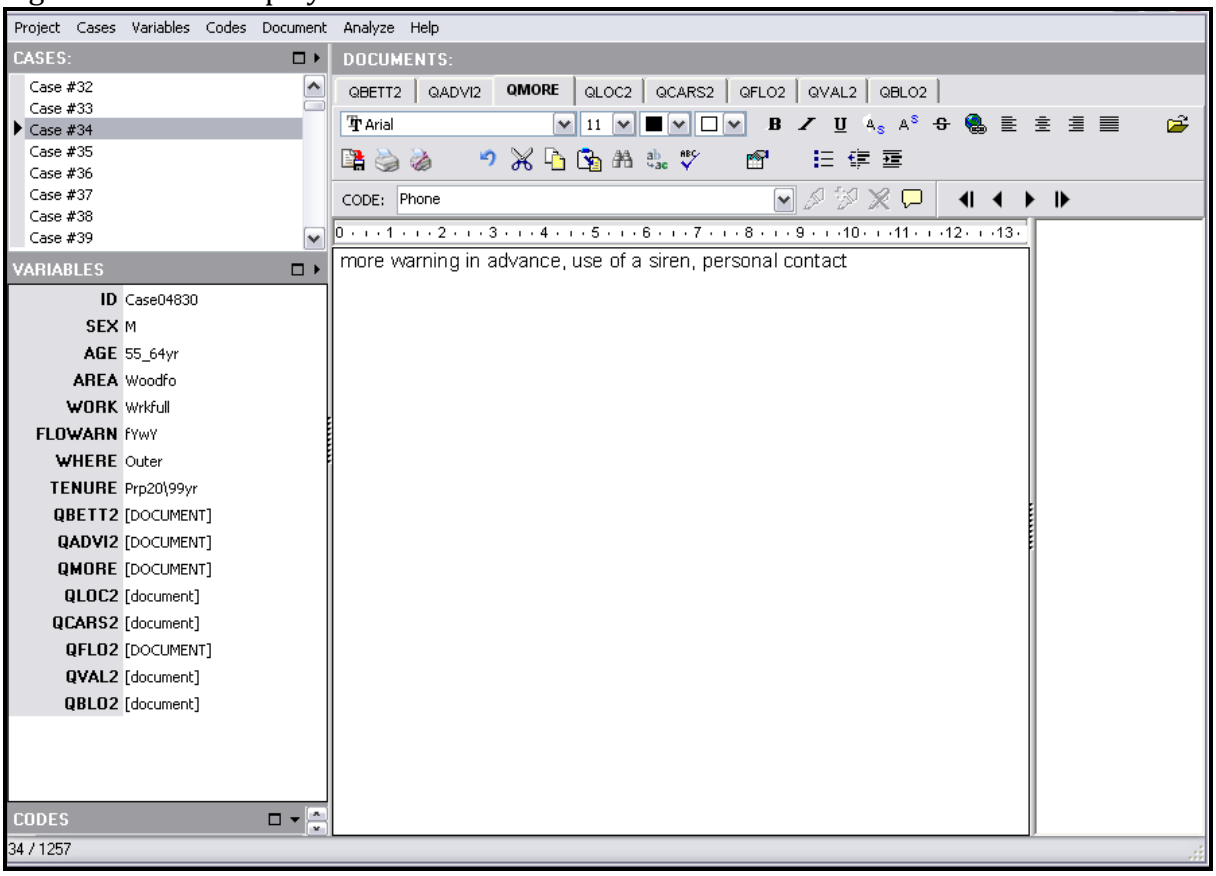
QDA Miner offers several different ways of approaching this sort of data and the final decision of how to do this will be determined by personal preference and analytic approach.

Figure 10 shows the main working screen (with the coding panel temporarily closed). In this display the user can select any respondent by clicking on their line in the Cases panel (here Case #34 has been selected). The Variables panel beneath that has been expanded to show all the variables that have been brought into this project, and those with response texts can be identified by the word "[document]" here, where this word is in capital letters there is a text response in the dataset but

where it is in lower case then there is no response. To read any of these texts it is necessary to click on the appropriate tab label at the top of the Documents panel (here "QMORE" has been clicked and shows up in bold font). So the text on display is the response to question "QMORE" made by Case #34.

Now by clicking on the other tab labels (particularly those with "[DOCUMENT]" in capitals in the Variables panel) one can read each of the responses provided by this case. Or, leaving "QMORE" in bold, either by clicking on the forward and back scrolling arrows in the lowest toolbar or on the cases labels instead one can read each of the responses to this question. The choice to work by case or by question is completely open.

Figure 10: Main display Screen



However, where there are many gaps in the responses, because most respondents only answered a few of the open questions, this approach will be frustrating with many empty screens being seen. In this situation a Text Retrieval report may be found to be a more satisfactory alternative.

The Text Retrieval report is found under the *Analyze* main menu. First let us consider using this to generate a list of all the responses to a single question. In Figure 11 just three settings are required in the dialog box for this operation: beside "Search in" use the drop-down menu to select the document label for the required question, set the "Search unit" to "Paragraphs", and click the radio button beside "Retrieve all units". Hit the "Search" button to see the report within the same window. Note how "Search Expression" and "Search Hits" are the two tab labels within this window, it is easy to switch back and forth between these to try different settings and see their effects.

Figure 11: Text Retrieval dialog settings to view all responses to QMORE

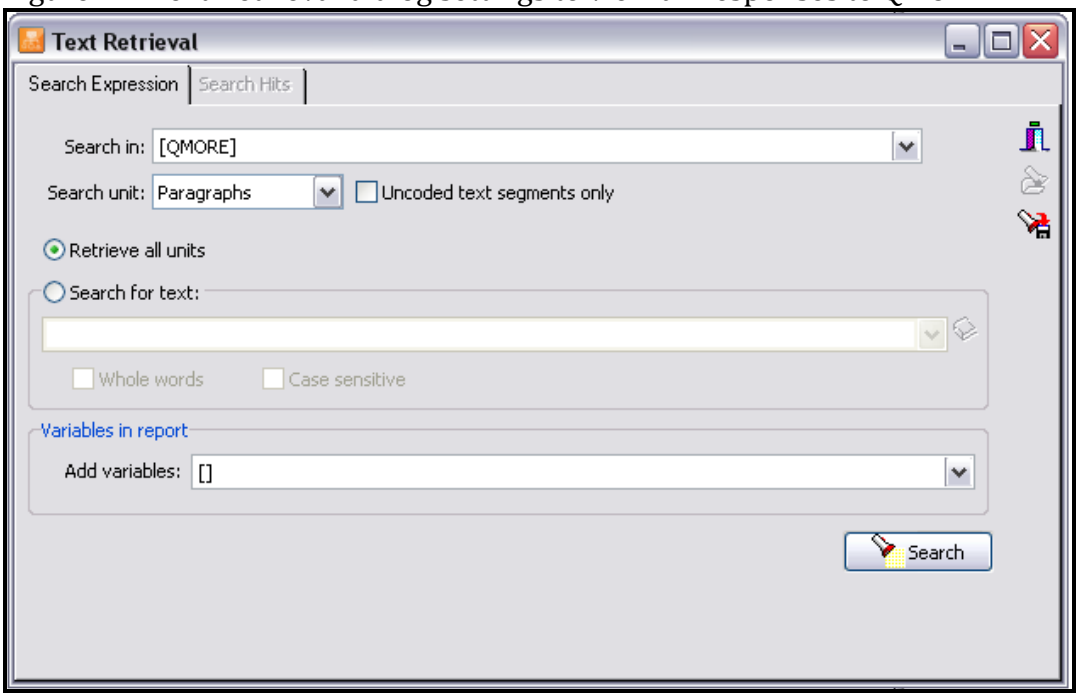
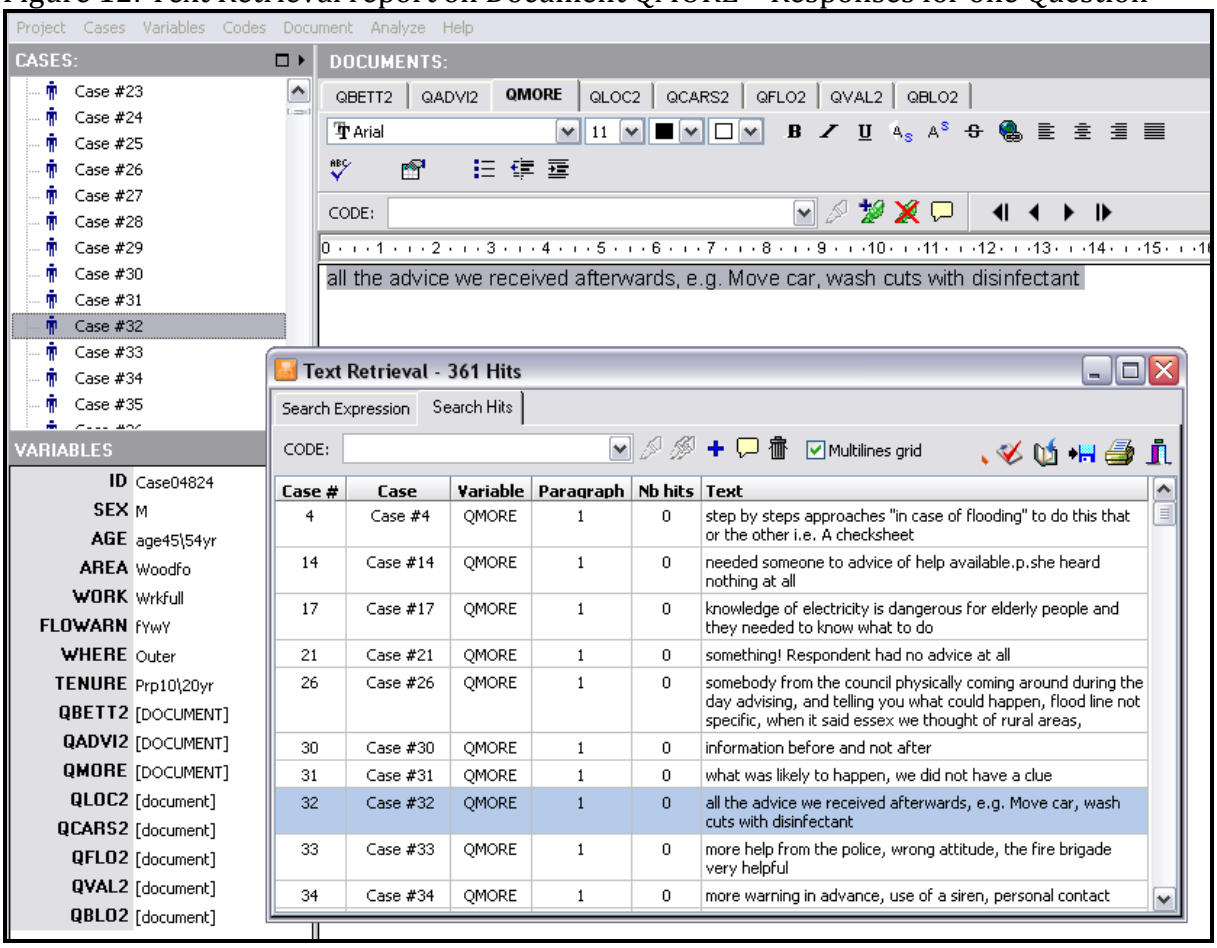


Figure 12 shows an example report for the document "QMORE" placed over the main working window. The query has found 361 hits, so these represent the full set of responses to this single question. A tick has been placed in the "Multilines grid" box just above the results list, and this forces the complete display of the longer texts (such as that by Case #26). It will be important to be able to read complete texts to be sure that nothing is missed in the analysis. Note also that the report window is synchronised with the main screen so, when an item in the report window is selected, the main screen behind it will change to display that response and the Variables panel will display the attributes for that respondent.

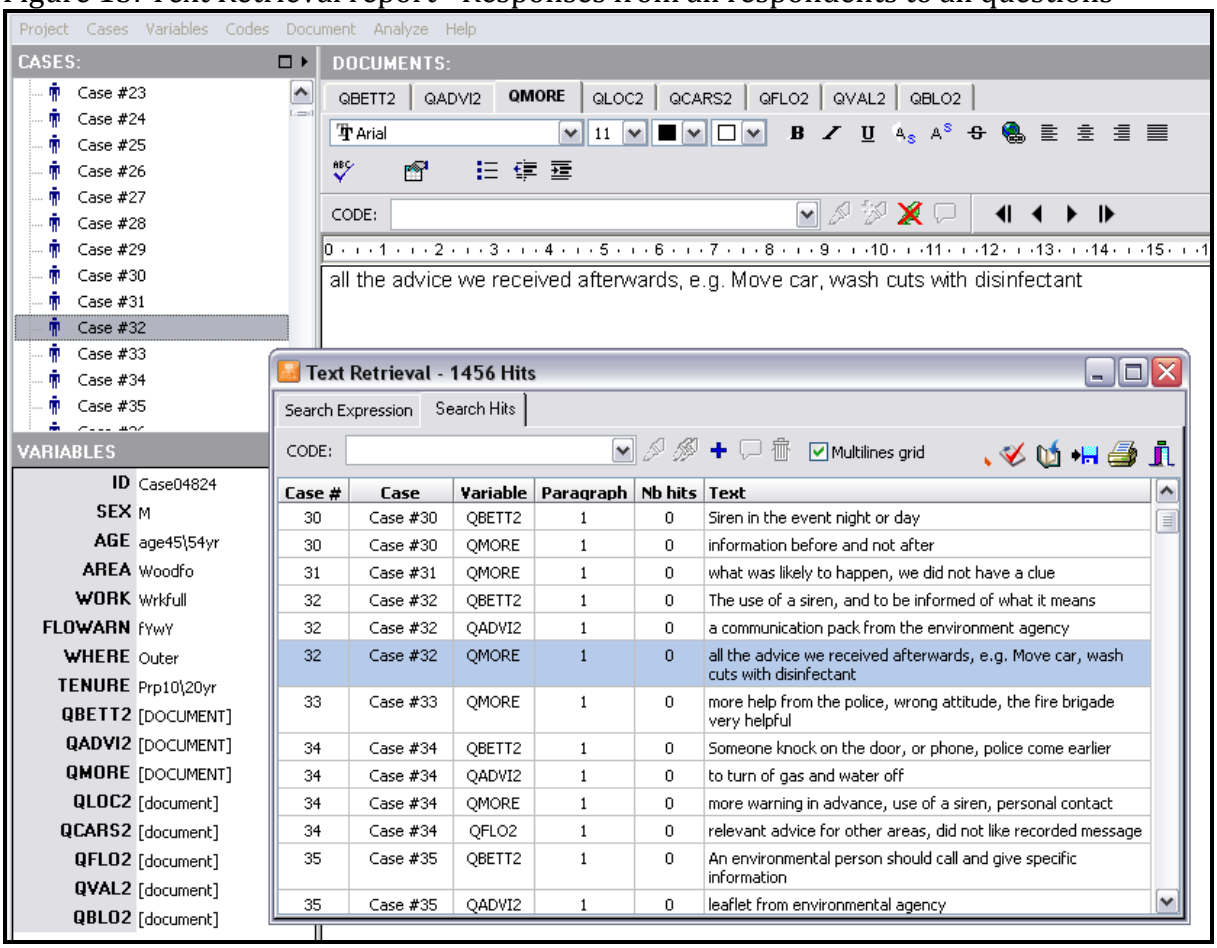
Figure 12: Text Retrieval report on Document QMORE – Responses for one Question



Secondly, if you prefer to read all of the responses made by each respondent, then select all of the documents at the "Search in" setting on the "Search Expression" screen (Figure 11). You will get a longer report, but it will be sorted by case number before variable. Figure 13 shows an illustration of this using the same dataset as before. Once again the search window is synchronised with the main window, so selecting one item in the search window brings up all of its related data in the main window, and codes can be applied to individual responses in the main window without closing the search window.

In both Figure 12 and Figure 13 the same particular response has been selected, that of Case #32 to question QMORE. In Figure 12 we can easily compare what this respondent said with the responses to this question made by other cases. In Figure 13 we can easily read all of the responses made by this case to all of the questions, so #32 only answered three questions while #34 answered four questions.

Figure 13: Text Retrieval report - Responses from all respondents to all questions



Further, it is possible in QDA Miner to control the sequence in which cases are listed according to their values in a variable. The default, case number, is simply the order in which they were arranged in the spreadsheet from which the data were imported. By using the menu option *Cases / Grouping / Descriptor...* the user can select one or two variables as grouping terms and also include those values in the display in the cases panel. These settings then apply to the Text Retrieval reports as well, so that they will be grouped according to the same variables. For example this facility might enable the analyst to read the responses to a question about flood warnings firstly by those who did receive a warning and then by those who were not warned.

2.2 Developing a coding scheme – manually or by using word frequencies?

Many analysis projects will be set up with a coding scheme derived from other work or sources; in these situations the following comments will not really be relevant. If, on the other hand, you are expecting to derive your coding categories from the ideas mentioned in the response texts themselves then you have a choice as to whether to do this by reading the texts and choosing categories that seem to be mentioned in those texts (I have termed this “manually” for want of a better term) or alternatively to let the software help by creating a list of the most frequently used words in the texts.

The manual method will be required at some stage if really accurate coding is needed, because only human readers can detect all of the subtleties of human expression involving multiple ways of

phrasing any particular idea. However to get started, particularly in a large dataset, it should be worth trying the word count method to get an early idea of the range and density of words used. The most frequently used words may be expected to provide indications of the most frequently expressed concepts.

QDA Miner has a substantial extra module designed for automating the process of searching for equivalent meanings in multiple texts called "WordStat". This module is a sophisticated suite of content analysis programs with considerably more functionality than will be described here. However, as there is no word frequency function in the main QDA Miner program it will be necessary to use WordStat for that purpose.

Before looking in some detail at the use of WordStat it may be worth summarising three different approaches to this analysis challenge. If your approach is to be mainly deductive, because you have a good idea of the concepts that you are looking for (and maybe also the language in which they may be expressed) then you probably do not need the WordStat module, you can create the basic coding scheme first and then use various Text Retrieval strategies to identify the responses that those codes should be applied to. Secondly, if your data is not particularly 'rich' in detail (especially if it was heavily mediated by interviewers paraphrasing the responses as they typed them) but you wish to work inductively and develop the coding structure from the response data, then the Text Retrieval and Query by Example tools in QDA Miner may yet be sufficient for your purposes. The WordStat module really becomes useful when the language recorded in the data is likely to be more expressive and differentiating, or where a similar analysis is likely to be repeated on fresh data around the same topics from time to time (so that the effort of developing categorisation "dictionaries" is repaid with labour-saving efficiencies).

TIP: The decision whether to use WordStat may have significant financial implications if you are considering purchasing the software for this analysis as, depending on the status of the purchasing organisation, WordStat may be almost as expensive or even more expensive than the QDA Miner program. This is why these instructions include some suggestions of analysis without the use of WordStat.

Although most qualitative researchers may prefer to do the coding work manually, that is to say by reading each response and making a personal judgement as to which codes to apply to it, there will be a lot of scope for allocating codes automatically at the large scale end of the survey spectrum where QDA Miner comes into its own. These functions are available in the core QDA Miner program and in the WordStat add-on module and are described below with the other routines in which they are embedded.

2.3 Developing and applying a coding scheme in QDA Miner only.

The initial, basic, approach to analysing responses to an open-ended question asked in a survey generally involves the analyst reading a sample of those responses and noting down the concepts which can be identified within that sub-set of the data. The list of concepts is then studied to see if it can be simplified by grouping some similar ideas together in fairly inclusive ways, and a systematic list of codes can then be developed from that list of grouped ideas.

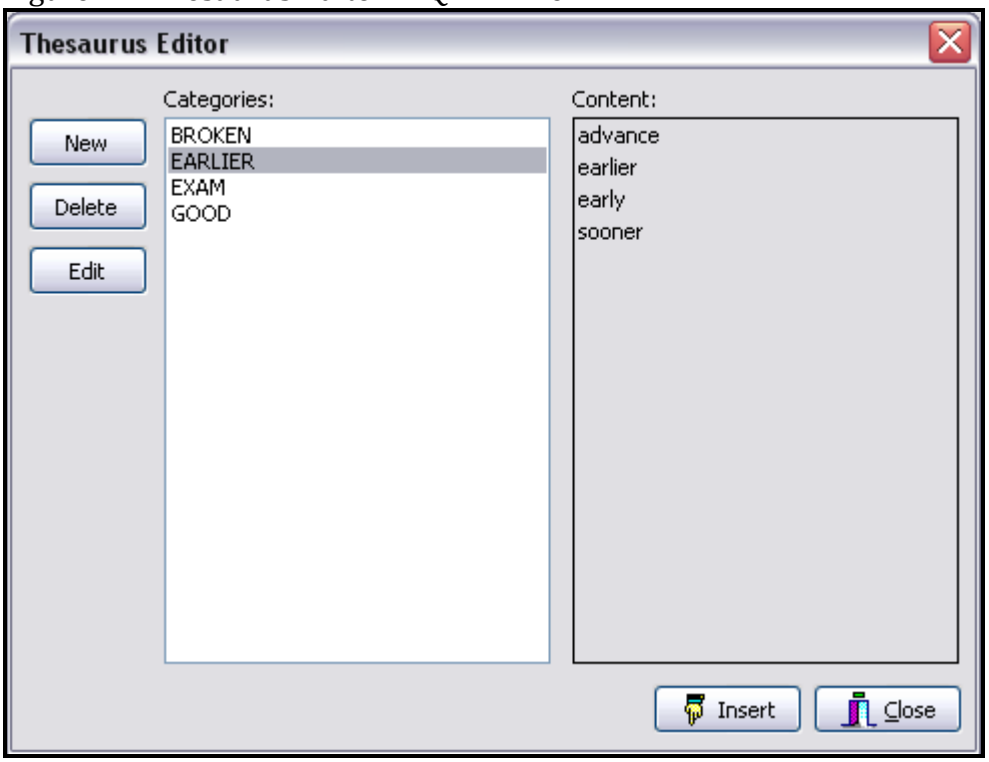
If you are working manually, with only the main QDA Miner program to assist you, it will be important to note down the particular words that you notice in the data as indicators of the presence of a potential theme. One way of doing this is to have a blank sheet of paper on which you write down the useful words and to try to group those words along separate lines from the outset. So in our data we analysed a question that had asked what more advice people should have been given prior to the flooding event. One set of words that we noticed included the following "wash cuts, disinfectant, portable loo, boiling water, fill bath, contamination, health, safety" and these all seemed to indicate a possible theme connected with health. However these words did not appear neatly in the order just listed but were interspersed with other words that gradually built up other themes. Soon the blank sheet of paper was covered in webs of connections as a variety of themes emerged from the data.

The next step is to identify common labels that effectively describe each sub-group or theme, as these will become the code labels in the next phase of the work. It is important to recognise that these groups and labels may change as your understanding of the data grows, but you have to start somewhere. What you are doing is effectively the first stage of much content analysis, you are building "dictionaries" for use with the program. This use of the term "dictionary" is different from the common meaning because you are not attempting to define the meaning of each code label with precision, you are instead attempting to identify sets of words with similar or related meanings, an activity which is more commonly attached to the term "thesaurus". QDA Miner and WordStat use both of these terms in various places.

Having identified some themes, and a set of words found in the data that can be associated with each, from a subset of the data for one question, you are ready to use the program to explore these with the full response set for that question. There are three possible ways to proceed in QDA Miner – by using a Text Retrieval and Thesaurus approach, by using a Code and Keyword Retrieval approach, or by using a Query by Example approach – each will be described briefly below. Each has its own advantages and disadvantages and there is no reason why a combination of all three should not be used sometimes.

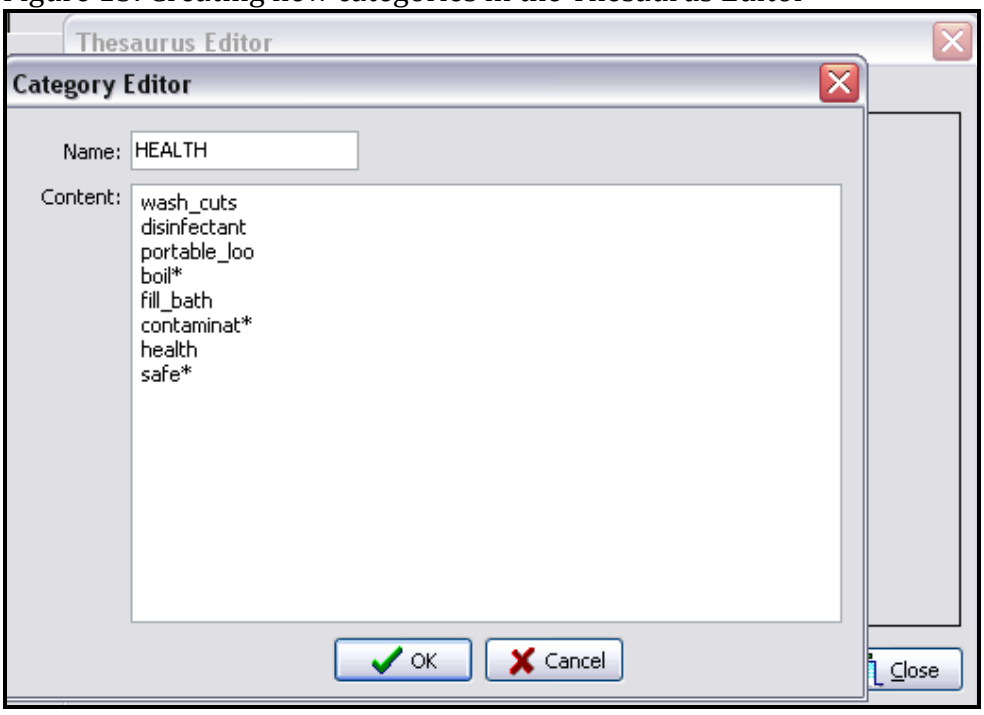
Text Retrieval and Thesaurus:
We have already described the initial Text Retrieval process to extract all of the responses to one question. This procedure will now be taken a stage further. Start a new *Text Retrieval* from the *Analyze* menu and select the question code required for the "Search in:" field. Set the "Search unit:" to "Paragraphs" as before and leave the "uncoded text segments only" box empty. This time click in the radio button beside "Search for text:" and then click on the red book symbol at the extreme right of that option field to open the thesaurus editor. Figure 14, below, shows an illustration of the thesaurus editor when it is first opened with the example data inserted by the program.

Figure 14: Thesaurus Editor in QDA Miner



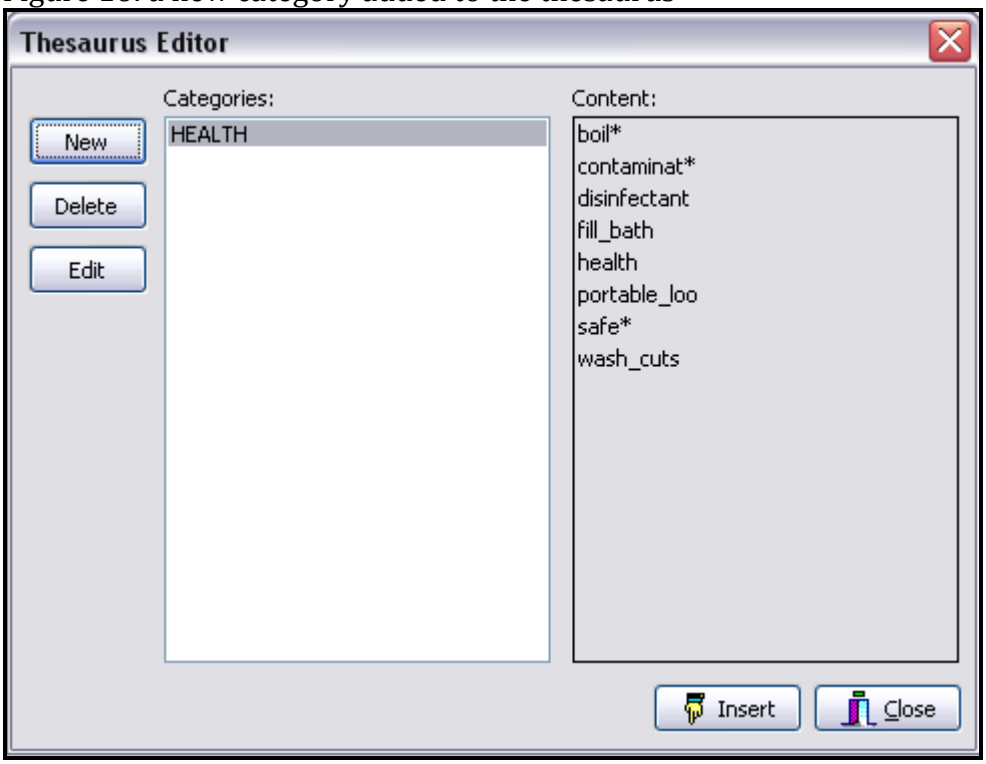
The program suggests four initial categories for illustrative purposes, "Broken, Earlier, Exam, and Good", here "Earlier" has been selected and the suggested words for that category can be seen in the "Content:" panel. Once you have grasped the idea you can delete these illustrative categories and start to create your own, by using the "Delete" and "New" buttons on the left. To continue our earlier example, we created a new category for "Health" and then added the words already identified with it into the "Content:" panel to create Figure 15.

Figure 15: Creating new categories in the Thesaurus Editor



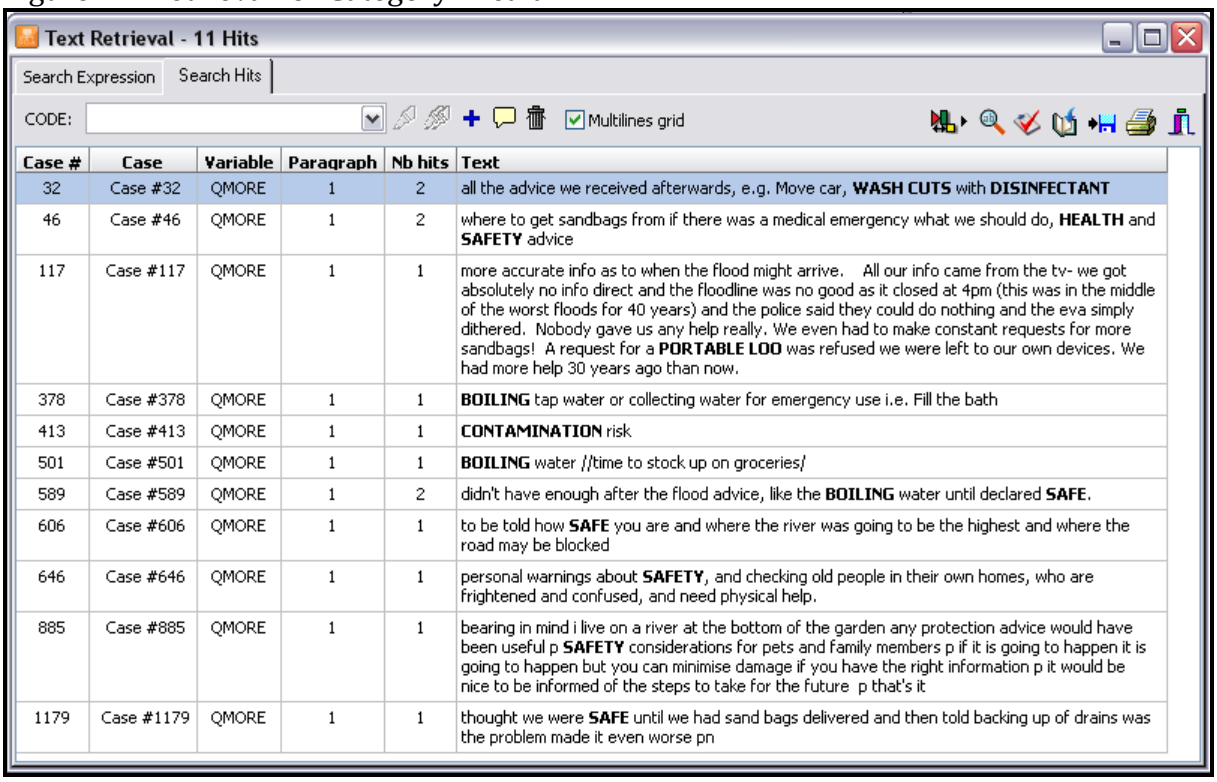
Note in Figure 15 how the underscore character has been used to create terms with two words and the asterisk has been used as a wild card character so that words like “boiled” and “boiling” will be picked up in addition to “boil”. When the “OK” button is clicked the new category is added to the thesaurus, as can be seen in Figure 16.

Figure 16: a new category added to the thesaurus



Further words can be added to the category content at any future stage by using the “Edit” button, which will reopen the Category Editor. At this stage more categories can be created as required by using the “New” button again. To use a category, highlight it in the Thesaurus Editor and click on the “Insert” button to add the highlighted category to the Text Retrieval expression with the prefix “@”. Running the search will then generate a list of hits, each containing at least one of the words or phrases listed in the category content field. Figure 17 shows the output achieved with this category in our example data.

Figure 17: Retrieval for Category "Health"



Case #	Case	Variable	Paragraph	Nb hits	Text
32	Case #32	QMORE	1	2	all the advice we received afterwards, e.g. Move car, **WASH CUTS** with **DISINFECTANT**
46	Case #46	QMORE	1	2	where to get sandbags from if there was a medical emergency what we should do, **HEALTH** and **SAFETY** advice
117	Case #117	QMORE	1	1	more accurate info as to when the flood might arrive. All our info came from the tv- we got absolutely no info direct and the floodline was no good as it closed at 4pm (this was in the middle of the worst floods for 40 years) and the police said they could do nothing and the eva simply dithered. Nobody gave us any help really. We even had to make constant requests for more sandbags! A request for a **PORTABLE LOO** was refused we were left to our own devices. We had more help 30 years ago than now.
378	Case #378	QMORE	1	1	**BOILING** tap water or collecting water for emergency use i.e. Fill the bath
413	Case #413	QMORE	1	1	**CONTAMINATION** risk
501	Case #501	QMORE	1	1	**BOILING** water //time to stock up on groceries/
589	Case #589	QMORE	1	2	didn't have enough after the flood advice, like the **BOILING** water until declared **SAFE**.
606	Case #606	QMORE	1	1	to be told how **SAFE** you are and where the river was going to be the highest and where the road may be blocked
646	Case #646	QMORE	1	1	personal warnings about **SAFETY**, and checking old people in their own homes, who are frightened and confused, and need physical help.
885	Case #885	QMORE	1	1	bearing in mind i live on a river at the bottom of the garden any protection advice would have been useful p **SAFETY** considerations for pets and family members p if it is going to happen it is going to happen but you can minimise damage if you have the right information p it would be nice to be informed of the steps to take for the future p that's it
1179	Case #1179	QMORE	1	1	thought we were **SAFE** until we had sand bags delivered and then told backing up of drains was the problem made it even worse pn

In Figure 17 it can be seen in the Text column that the words in the selected thesaurus category have been highlighted in bold font and capital letters, and the number of highlighted items is shown for each response in the fifth column ("Nb hits"). The wild card characters worked successfully, but the phrase "fill_bath" was unsuccessful as it was not matched with the phrase "fill the bath" used by case #378, so some refinement of the category may be necessary ("fill_*_bath" would match both "fill a bath" and "fill the bath").

On reading closely we decided that the final hit in Figure 17 used the word "safe" in a different sense and so we did not want to code that with all of the other hits. It could be removed from the list of hits by selecting the row (Case #1179) and clicking on the dustbin icon in the toolbar within the search hits window. Note that this does not delete any data from the project, it just excludes it from these search results. To code the remaining 10 hits with a single code, say called "Health & Safety", use the pull-down menu in this window to select an existing code, or click on the "+" icon to create a new code, and then click on the double highlighters icon (which becomes available when a code has been selected) to apply it.

TIP: Note that the code will be applied to whatever data you have asked to have listed in the search expression. In this example we asked for "Paragraphs" as the search unit and so the full response has been extracted for each hit and the code will be applied to each applicable response in full. We could have asked for "Sentence" and then the coding would have been applied just to each sentence that included one or more of the words in the thesaurus category. (Our data was made up of short statements so the paragraph unit is meaningful; in some other circumstances responses may be much longer, so there is an interaction between the data collection, preparation and analysis in this respect).

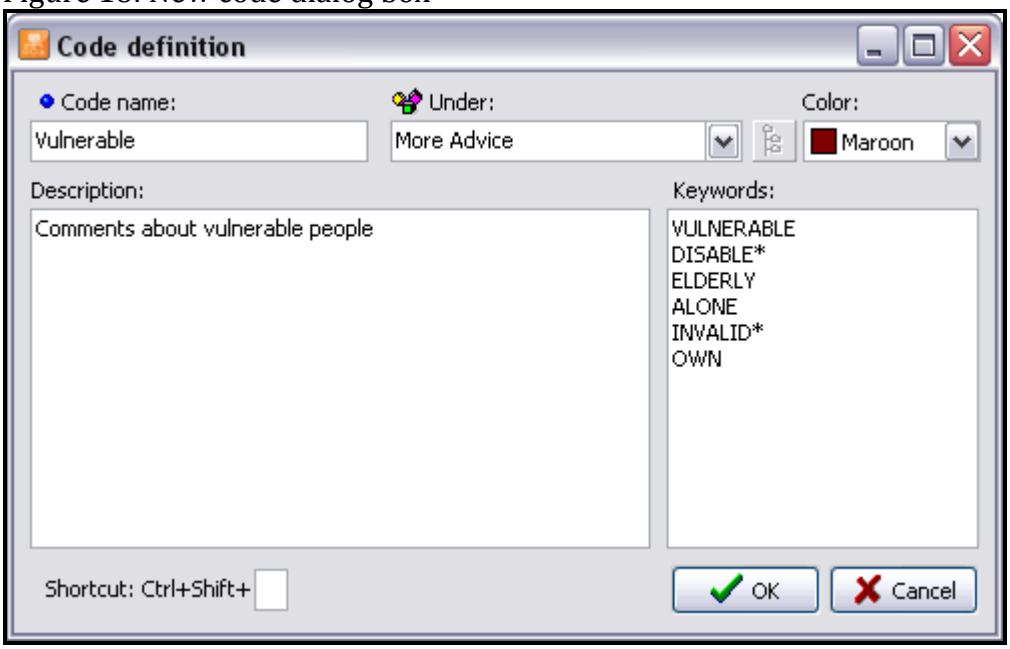
TIP: If more words are identified subsequently that relate to this code, they can be added to the category and the search can be repeated. Unfortunately, if you repeat the autocode QDA Miner will

duplicate the coding in all of the previously coded hits, so you may need to check each hit separately and apply any subsequent coding more carefully.

Code and Keyword Retrieval

This approach is essentially similar to the thesaurus approach described above but, instead of creating categories in the text search thesaurus, it uses the "Keyword" field in each code definition. Figure 18 illustrates the dialog box that appears each time you create a new code in QDA Miner.

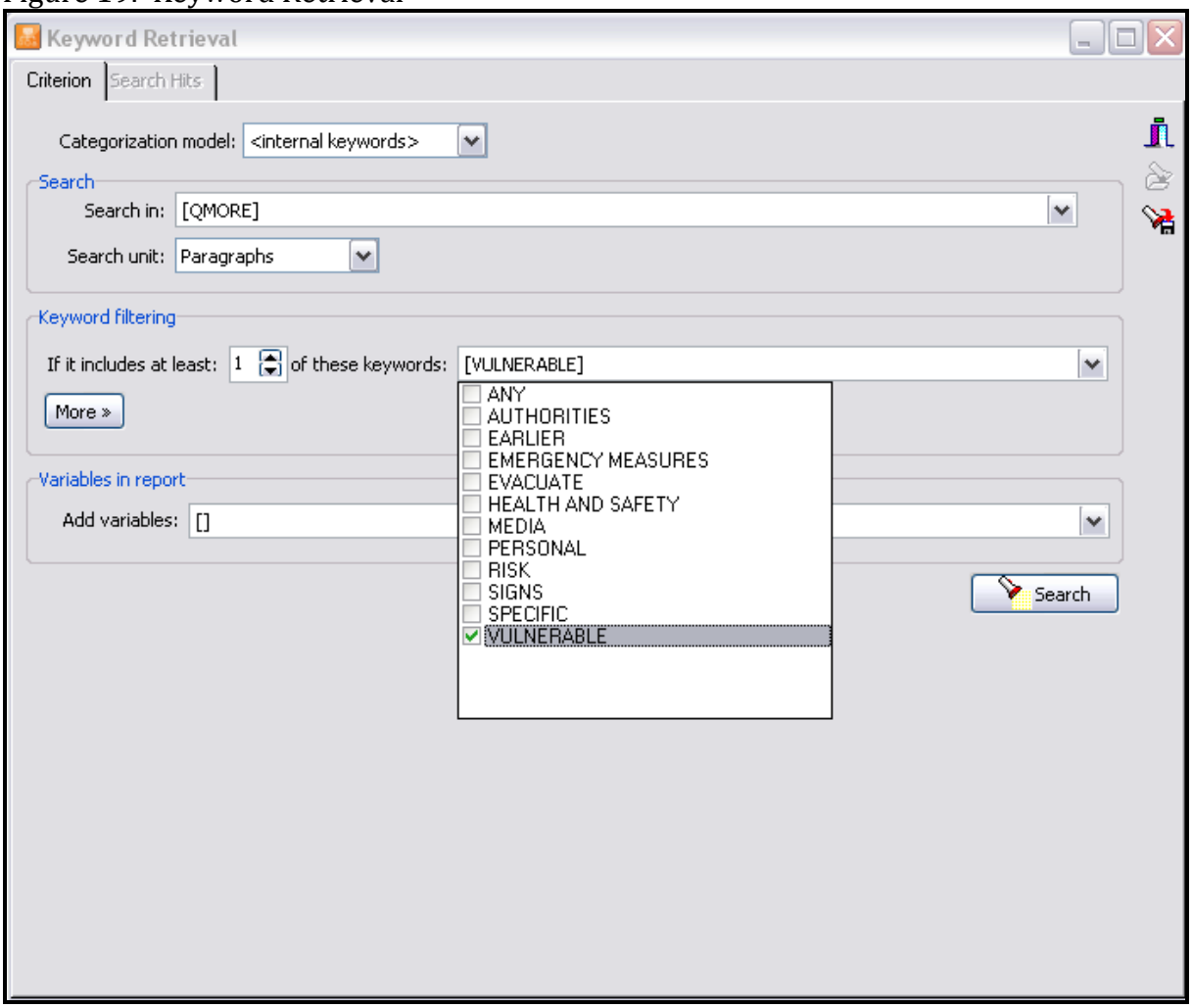
Figure 18: New code dialog box



Here a code has been created to capture comments about various types of vulnerable people within the coding group "More Advice". The set of words that have been typed in the "Keywords:" panel here can be used in a similar way to the set of words entered into a thesaurus category, and similar use can be made of wild card characters and underscores to create phrases.

To use the code keywords you need to apply the menu option *Analyze / Keyword Retrieval* which brings up the dialog box shown in Figure 19, below. When the option is first selected the dialog box looks rather different as only the first field can be seen, but when "<internal keywords>" is selected from the pull-down menu in that field then the rest of this dialog comes into view.

Figure 19: Keyword Retrieval



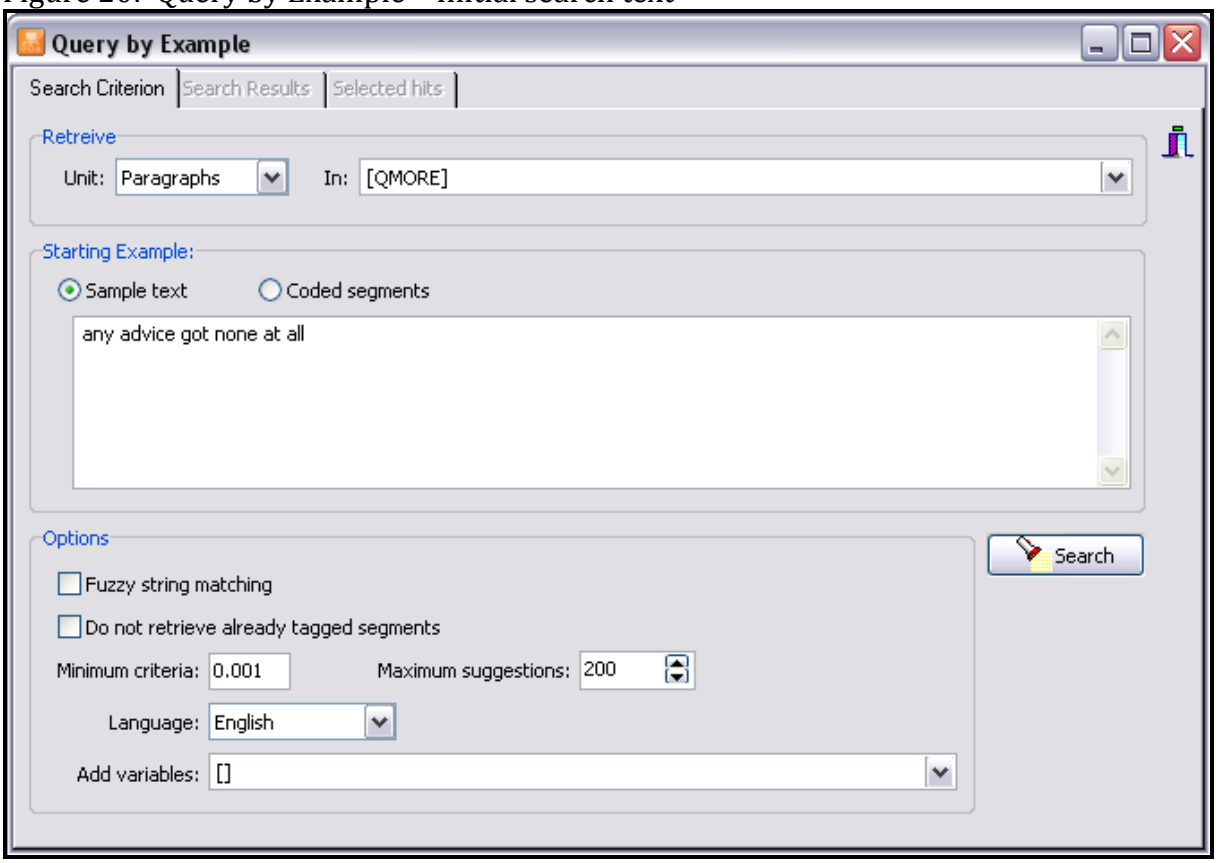
The main Search parameters work similarly to those in the Text Retrieval dialog box and have been set here to search in the single question document "QMORE" and to return whole paragraphs. The Keyword filtering section has then been used to select the "vulnerable" group from the list, and the default setting of "at least 1 of these keywords" accepted. When the Search button is clicked the Search Hits will be displayed in a similar way to those of the Text Retrieval searches illustrated above, where they can be reviewed, any that are not relevant to the specific theme can be excluded, and the remainder coded in a single process as before.

TIP: It may be asked what the difference is between the thesaurus approach and the keyword approach. The most significant difference concerns re-use of the categories. The keywords for a code are specific to that code in the current project alone and cannot be used anywhere else directly. The thesaurus categories are specific to the program set-up on the current computer and so could be used directly in another project as long as it is being run on the same computer. We understand that in the future there may be additional facilities to save and copy a thesaurus between computers, but that is not currently available in QDA Miner v3.2.4.

Query by Example:

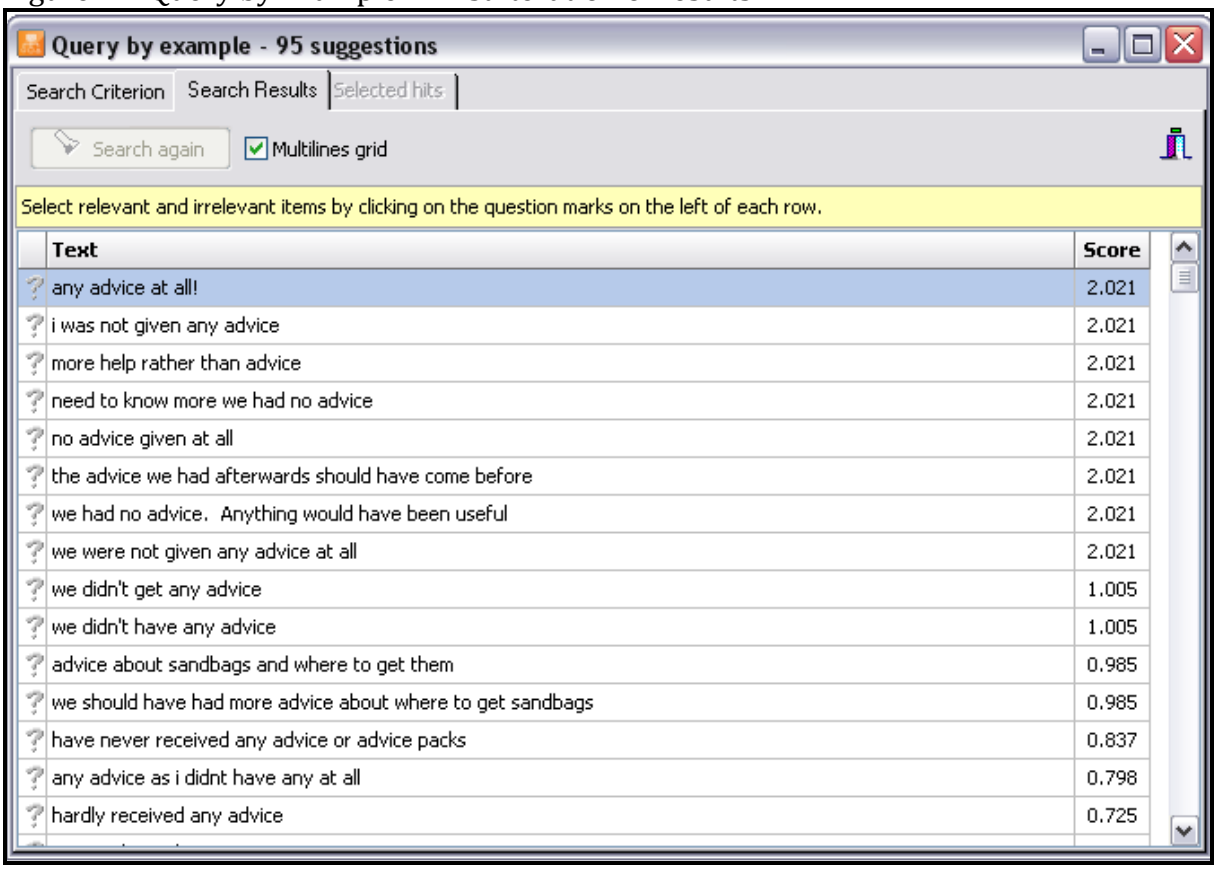
The third approach using functions available in QDA Miner alone uses the Query by Example routine which is found in the Analyze menu. Figure 20, below, shows an illustration of the dialog box with an initial search text of "any advice got none at all" entered as the starting example. This text was chosen as typical of several responses in the reviewed sub-sample without precisely copying any of them.

Figure 20: Query by Example – Initial search text



When the "Search" button was clicked the following screen appeared under the "Search Results" tab within the same dialog box.

Figure 21: Query by Example – First iteration of results



The next stage of the process involves working down the dialog shown in Figure 21 and changing the question marks in the left margin to train the program so that it can identify other equivalent responses accurately. A single click on a question mark changes it to a green tick to mark a 'relevant' response, a double-click changes it to a red cross to indicate an 'irrelevant' response, and a third click cancels the cross and returns it to the indeterminate question mark. When a sufficient number of these hits has been so marked, you should click on the "Search again" button (which becomes active once some hits have been marked) to re-run the search and obtain a more accurate set of results. This is an incremental process, in which you "teach" the program what you want and what you don't want, so that it can find further similar responses to those that you have told it you are interested in.

TIP: It is important to mark approximately as many crosses as ticks in this process, so that the program has some guidance as to what to exclude, otherwise it will add many more suggested hits because they include words similar to those in less relevant parts of the good hits.

Figure 22: Query by Example – First suggestions marked-up

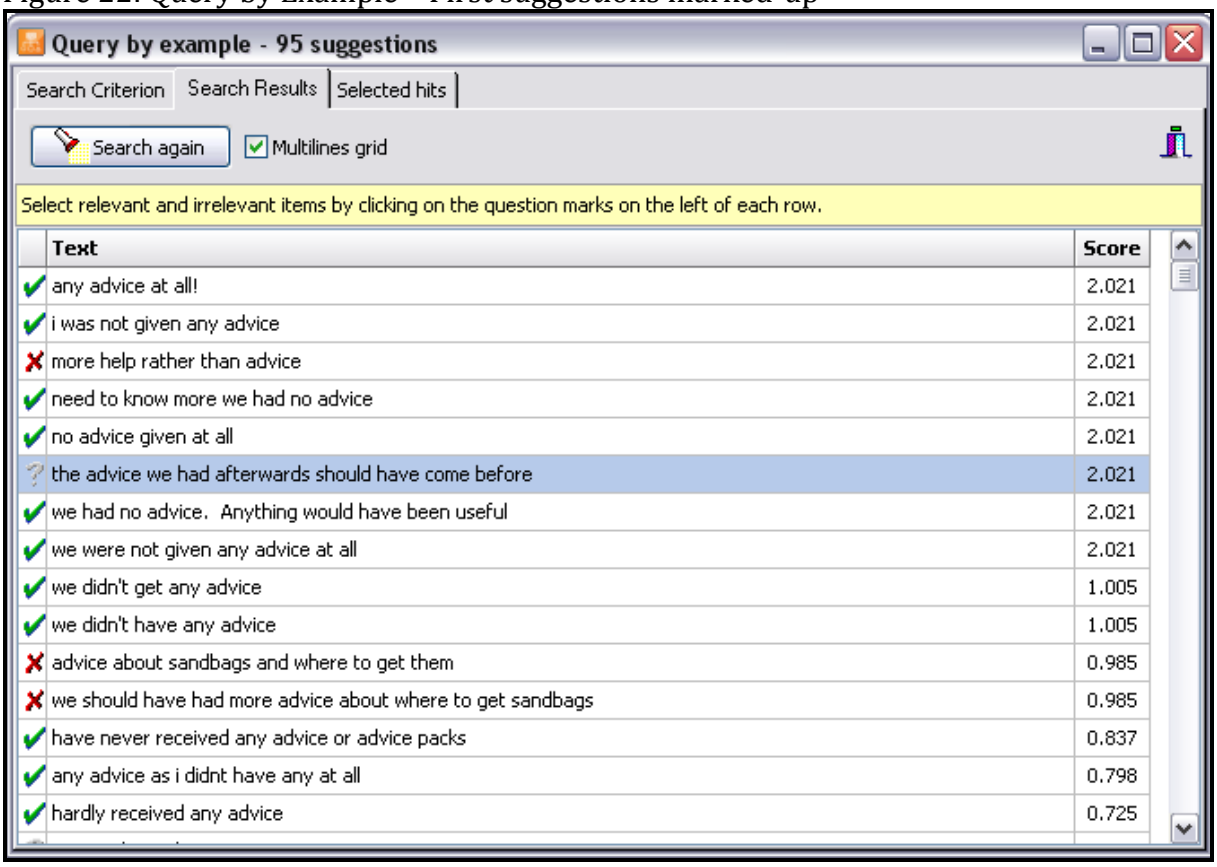


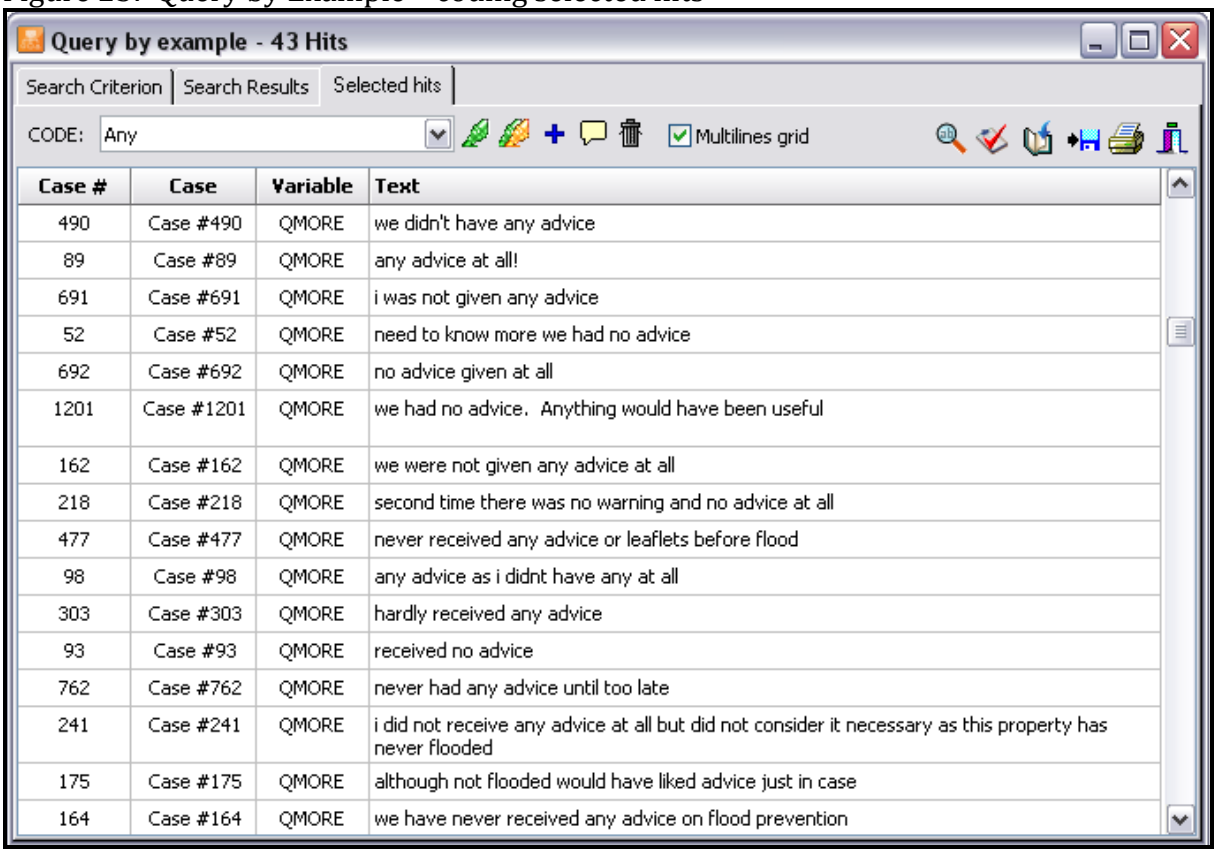
Figure 22, above, shows the same initial page of suggestions during the marking-up phase. Of the responses on view, some 12 items appear to have much in common with each other around the theme of not having been given any advice. But three items have been excluded as irrelevant because they introduce different concepts ("help" and "sandbags") and do not refer to the common theme directly. The item highlighted in blue is still under consideration, one interpretation of it could be similar to the current theme on the grounds that this respondent did not receive advice before the flood as they seem to have got it afterwards, but, if we 'tick' this one as also relevant, the program will probably include other responses that include words like "afterwards" and "before" in the next iteration and that may not be helpful. On balance we decided to exclude this response and to develop another theme around the timing of advice separately from this theme of no advice being received.

The single page of suggestions shown in Figure 22 above is probably not sufficient for this process, so it is necessary to use the scroll bar in order to view and mark-up more suggestions before clicking on the "Search again" button to get a second iteration of the query. Those responses that have already been marked retain their ticks or crosses after the second search but new suggestions will be inserted in the list and these should be read and marked as before. If this second iteration brings up a suggestion with a new word or phrase strongly associated with the current theme but not previously marked, then it will probably be worthwhile running a third iteration of the search, after ticking that response, to see if the program can find other responses with that word or phrase and a similar meaning.

It is necessary to continue marking-up the search hits until you reach the point where you are no longer finding any new relevant responses. It is not necessary to keep on marking crosses, unless you are planning to repeat the search in a further iteration of the process. But you do have to positively

mark all of the hits that you wish to be coded and the purpose of this routine is to provide you with a list of responses that become progressively less similar to the examples you have chosen, so that you can judge when to stop marking. At that point you should click on the third tab label in the dialog box, "Selected hits", to get a view similar to Figure 23, below.

Figure 23: Query by Example – coding selected hits



Case #	Case	Variable	Text
490	Case #490	QMORE	we didn't have any advice
89	Case #89	QMORE	any advice at all!
691	Case #691	QMORE	i was not given any advice
52	Case #52	QMORE	need to know more we had no advice
692	Case #692	QMORE	no advice given at all
1201	Case #1201	QMORE	we had no advice. Anything would have been useful
162	Case #162	QMORE	we were not given any advice at all
218	Case #218	QMORE	second time there was no warning and no advice at all
477	Case #477	QMORE	never received any advice or leaflets before flood
98	Case #98	QMORE	any advice as i didnt have any at all
303	Case #303	QMORE	hardly received any advice
93	Case #93	QMORE	received no advice
762	Case #762	QMORE	never had any advice until too late
241	Case #241	QMORE	i did not receive any advice at all but did not consider it necessary as this property has never flooded
175	Case #175	QMORE	although not flooded would have liked advice just in case
164	Case #164	QMORE	we have never received any advice on flood prevention

Figure 23 illustrates the final stage of the procedure. It shows part of the list of 43 hits which had been ticked during the three iterations of the searching procedure, all of these hits can be checked with the help of the scroll bar on the right. Provided you are confident that your list is satisfactory then the whole list can be coded in one process. Either select an existing code with the pull-down menu for the "CODE:" field (the code "Any" has been selected here) or create a new code by using the "+" icon in the toolbar, and then click on the double highlighter pen icon in the toolbar to apply that code to all of the selected hits (here to 43 responses).

TIP: In this illustration we did not use two of the initial settings on the Search Criterion tab at Figure 20, so some brief comments about those may be useful here. The option to use "Fuzzy string matching" may be useful in some situations. This allows the program to include more words that are similar to the initial text, for example misspellings and grammatical variations. This should broaden the coverage of the query and lead to more suggestions being offered, but it may also increase the burden on the analyst with more irrelevant hits to be marked off, so some experimentation with this option is recommended. Also there is a tick box option by "Do not retrieve already tagged segments" and this may be useful when you are repeating a query to extend a code's application. The problem may be that the exclusion may be applied to a response which does not have the code being worked on (although it should be so coded) but which already has another unrelated code, so this may not be as helpful as it looks.

It is, of course, also possible in QDA Miner to use the *Text Search* function to search directly for specific words or phrases entered in the "Search for text:" field and then to apply a code to the set of results generated in that way. The disadvantage of this procedure may be the risk of duplicating the code application when subsequent searches are run for similar terms when two or more of these may occur in the same response. It is possible to use multiple terms as search parameters from the outset but the advantage of the thesaurus and keyword approaches is that a record is created of the words and phrases that have been used, and there are some possibilities of re-using these sets of terms with other data.

2.4 Developing and applying a coding scheme with WordStat.

When the WordStat module has been installed its functions are loaded with the menu option *Analyze / Content Analysis* within QDA Miner. A preliminary dialog screen requires the selection of the document variables to be analysed, probably one question should be analysed at a time so pick just one for now, at the next choice click by "All text" for a comprehensive review, and initially select the "... descriptive analysis only" option below that.

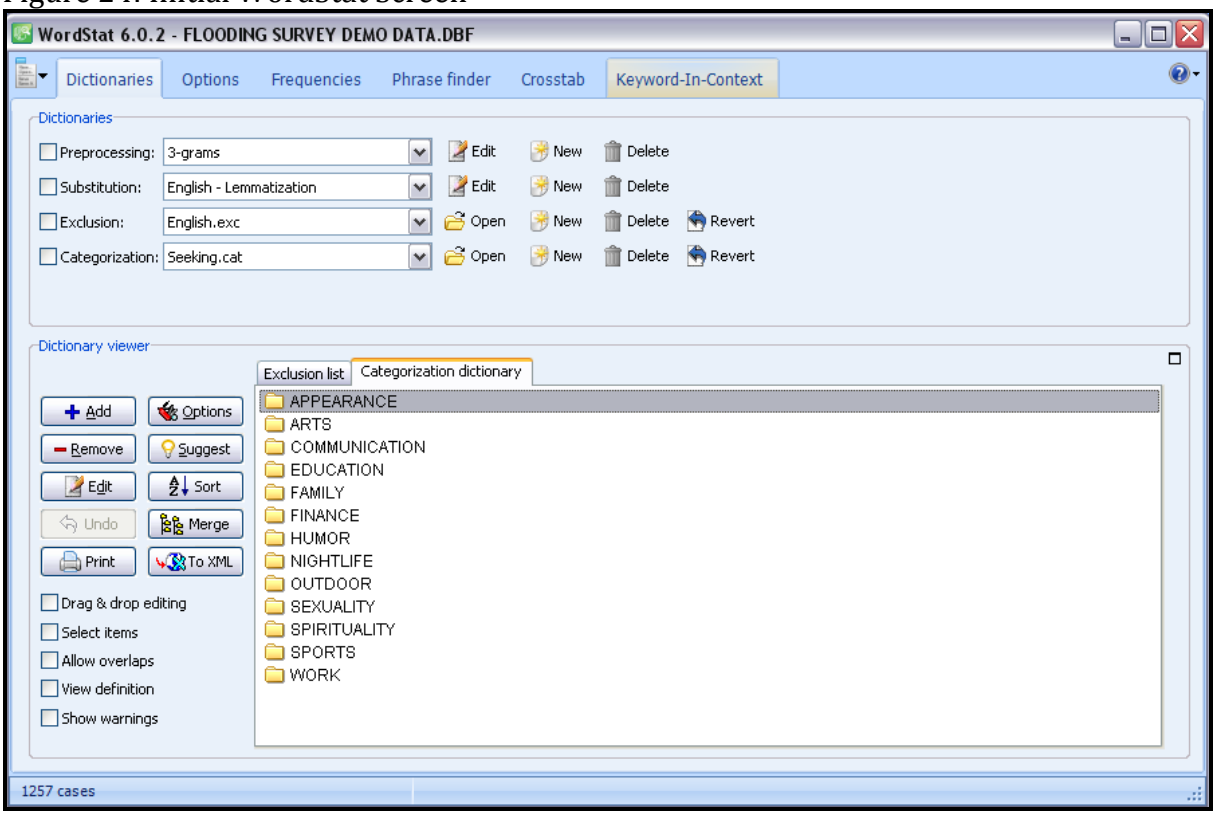
WordStat opens with a screen similar to that shown in Figure 24. This initial screen can be distinctly daunting, especially if you have a fairly simple dataset, because this is a sophisticated analysis program, but it can be used to run a simple word frequency routine.

Looking at Figure 24, the window can be divided into four horizontal sections; a set of six tabs at the top, a set of four tick boxes for the "Dictionaries" element, the "Dictionary viewer" part with two sub-tabs, and the bottom bar with some results statistics. As shown in Figure 24, remove any ticks from the boxes in the upper section, this makes the lower section with its "Exclusion list" and "Categorization dictionary" tabs inoperative. Then, click on the "Frequencies" tab in the top section and you will see a word frequency table for the document (or documents) that you selected at the start of this process.

TIP: Later, when you have become more familiar with this program, it may be useful to try ticking the "Substitution:" box, with the "Lemmatization" option selected, in order to bring together words with a common stem in order to see what effect this has on your subsequent work.

TIP: As a separate experiment, you could explore the effect of applying an Exclusion list. This is a list of words to be ignored by the program on the grounds that they are unlikely to be helpful in the analysis and may make it more difficult to see the important patterns. Words like "and", "the" and "at" may have little to contribute because they are so common that they are found in almost every response.

Figure 24: Initial WordStat Screen



For an example of the frequencies screen see Figure 26 below.

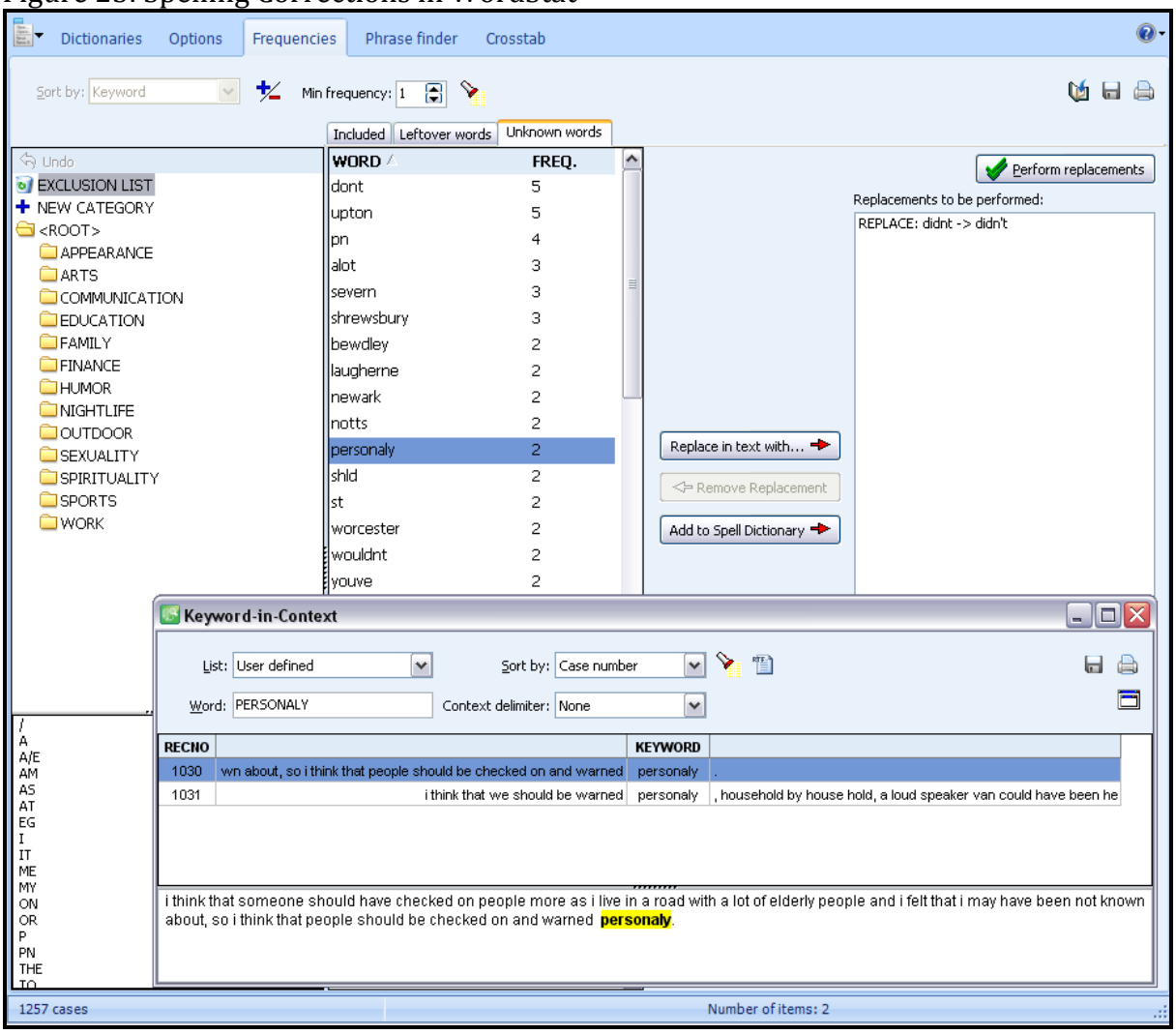
Because the option to use an exclusion dictionary was unchecked, the resulting word frequency table probably includes many words that do not help your analysis. However the default exclusion list is quite extensive and may remove words whose frequency you do wish to see. You can view the default exclusion list by clicking on the Dictionaries tab at the top of the screen, and then on the Exclusion list tab in the lower part of that screen ("Dictionary viewer"). If you are going to use WordStat for more analyses of this sort, you may find it useful to build your own exclusion list (the icons at the right hand end of the Exclusion line in the Dictionaries section of Figure 24 control this).

One problem that probably arises with much open-ended survey question data is the mis-spelling of words, because accurate spelling may not be regarded as a high priority in the survey situation. WordStat has some useful tools for correcting spellings and it may be helpful to apply these at an early stage. Firstly, you should check that the correct spelling dictionary is being used, and this can be done via the *Options* tab and its *Speller/Thesaurus* sub-tab. Then, if you click on the *Frequencies* tab and its sub-tab *Unknown words,* you can select an appropriate minimum frequency (we have used "1" to see all unknown words) and hit the *Search* button (flashlight icon) to display all sets of words found that do not match the dictionary.

Figure 25, below, shows a screenshot of the spelling correction stage. Select a misspelled word in the middle window, right-click for a context menu and select "Keyword in context" to view all of the occurrences of that word if you want to check how it has been used in the responses. In Figure 25 the word "personaly" is being checked and its two occurrences are shown in the lower overlapping window. When you have identified a word to be corrected, close the Keyword-in-Context window if it has been opened, then select "Replace in text with" if it has indeed been mis-spelled and either select

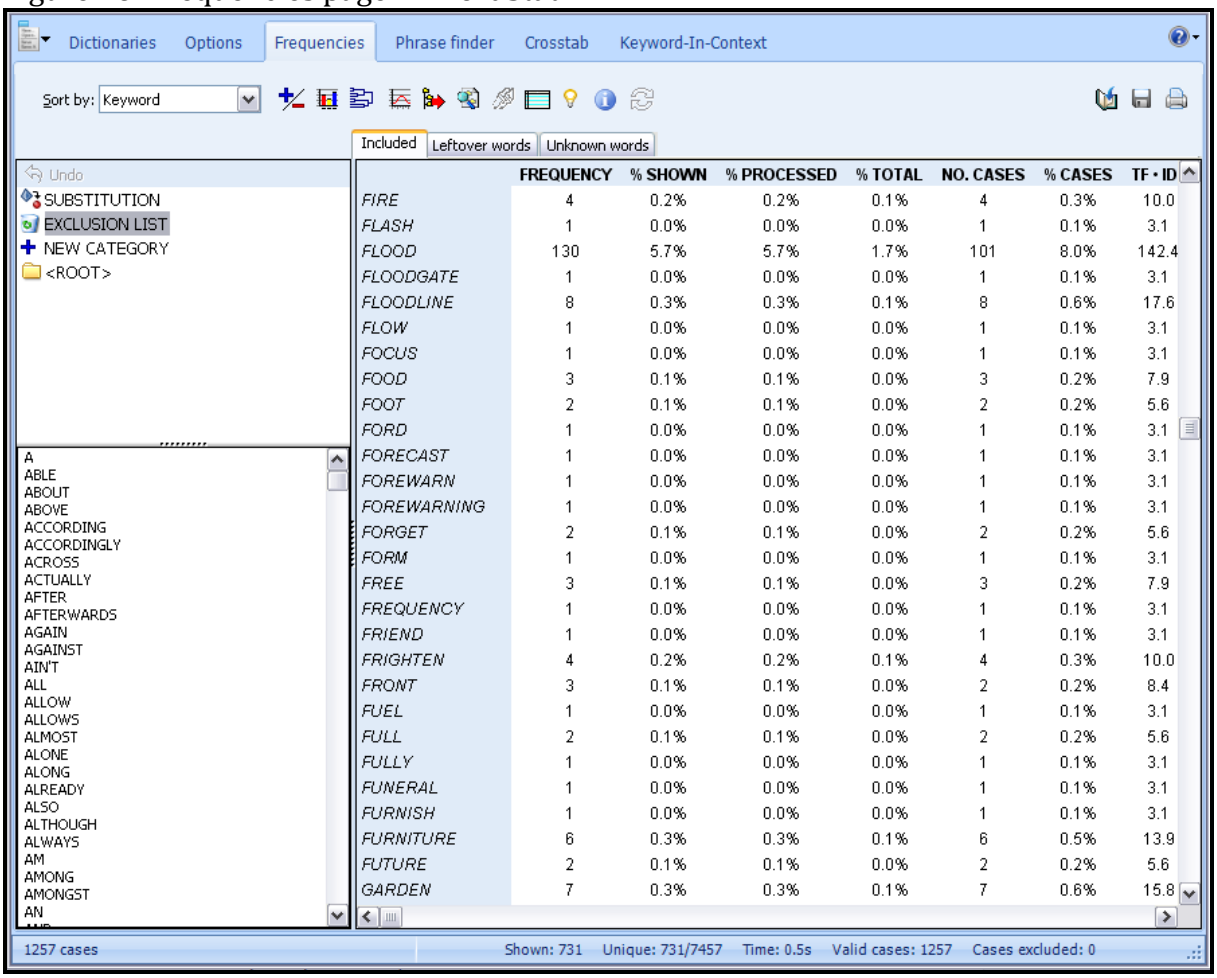
one of the offered corrections in the dialog box that then opens or type in your own correction and press OK, so that a record appears in the right hand panel under “Replacements to be performed:”. In this example we chose to correct “didnt” as “didn’t” and subsequently to correct “personaly” as “personally”. When you have corrected as many words as you wish, click on the “Perform replacements” button on the right and the source texts will be altered accordingly throughout the QDA Miner suite of programs. This simple routine should save you trouble later if you want to autocode for meaning using any of the correct versions of these words.

Figure 25: Spelling Corrections in WordStat



At any stage, after correcting spellings or altering an exclusion dictionary for example, you can return to the *Frequencies* page, select the “Included” sub-tab, and view the amended results. Figure 26 shows a display of this page. The default exclusion list has been applied in this example, removing several words which will not help the analysis much, and some potential themes are already apparent in this document (“floodline”, “furniture”, “garden” etc). This display can be sorted by frequency using the pull-down menu above the table, where other sort options are available, or by clicking on the appropriate column header. It is advisable to check the alphabetical sort sometimes (as shown here) because validly spelled variations of words (such as singular and plural versions) may still be shown separately (depending on the substitution/lemmatization setting on the Dictionaries page).

Figure 26: Frequencies page in WordStat



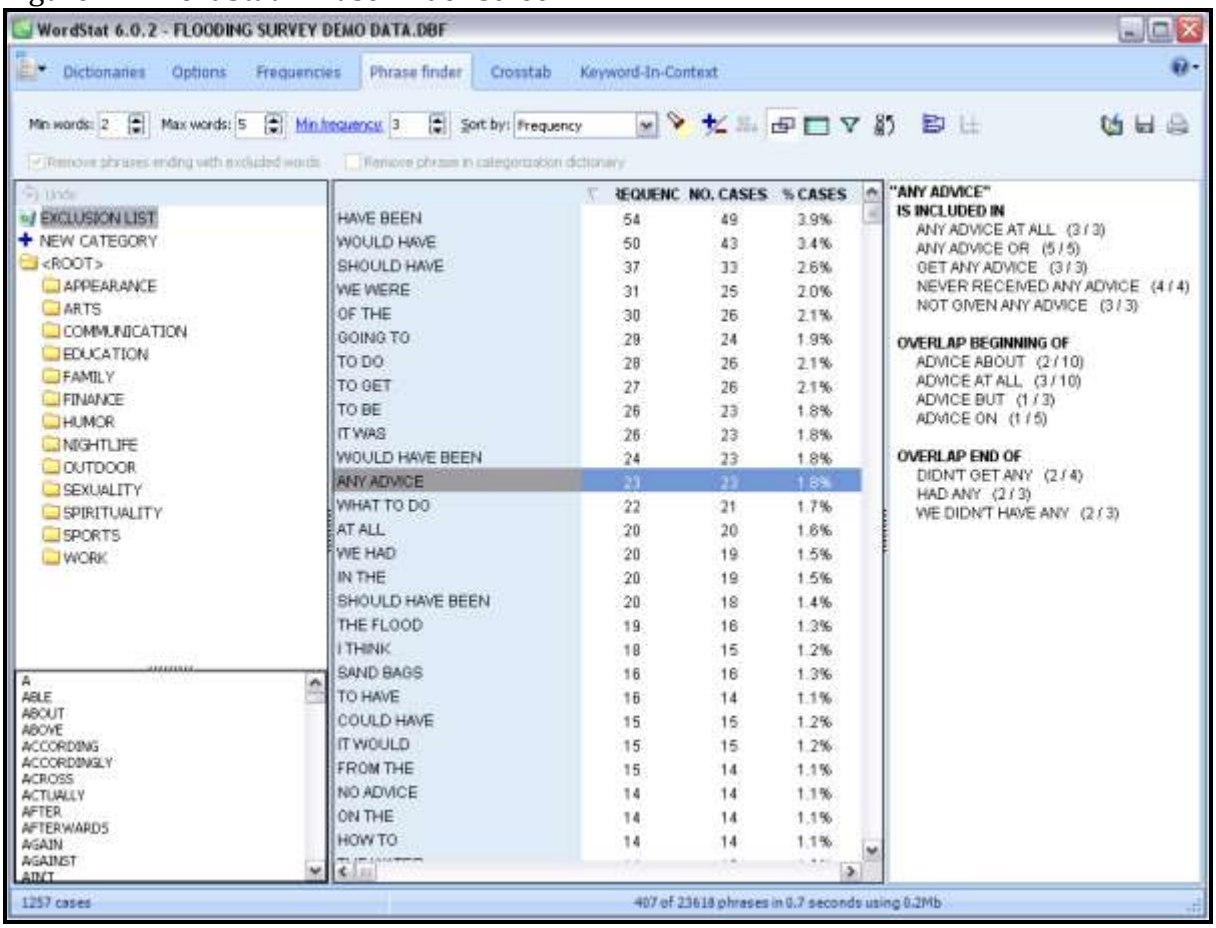
TIP: It is possible to display the exclusion list (as in Figure 26) by clicking on that phrase in the left hand panel, the full list then appears in the lower part of the left hand panel. To help show this, the category list has also been collapsed to its root. As can be seen here the default exclusion list is extensive, it was developed for particular analysis purposes and may not be appropriate for some types of dataset, for example it would exclude "after" and "afterwards" but these words may be significant in our example dataset when considering the timing of advice and warnings about floods. We would suggest not applying any exclusion list at first but to consider developing and applying your own list if you find your frequency table is getting clogged-up with too many trivial words.

TIP: A useful feature of this page is that a *Keyword Retrieval* function is available through the magnifying glass icon above the table. This opens in a separate window so that it can be used without obscuring the frequency table, and it has considerable functionality to include multiple words and apply filters. Alternatively the *Keyword-in-Context* tab can be selected to see an alternative way of examining how a single word has been used in the data. Both of these functions have selection boxes that work interactively with the frequency table list to aid the examination of important words. Moving between these various displays should help the formulation of principle coding themes. More will be written about the differences between *Keyword Retrieval* and *Keyword-in-Context* further down this page.

As a further aid to developing ideas about themes within the data it is possible to look at frequently used phrases in the text. These can be seen on the *Phrase finder* tab. Various parameters can be set by

the user but the defaults are a good place to start. Figure 27 shows an illustration of this function. Note the parameter settings near the top of the screen, "Min words: 2 - Max words: 5 - Min frequency: 3 - Sort by: Frequency" and then the search button in the form of a flashlight icon which has to be clicked to create the phrase list in the central panel. An icon of two overlapping rectangles switches on the overlapping phrases panel which is also shown in this illustration, this lists other phrases which overlap in whole or in part with the phrase selected in the central panel.

Figure 27: WordStat Phrase finder screen

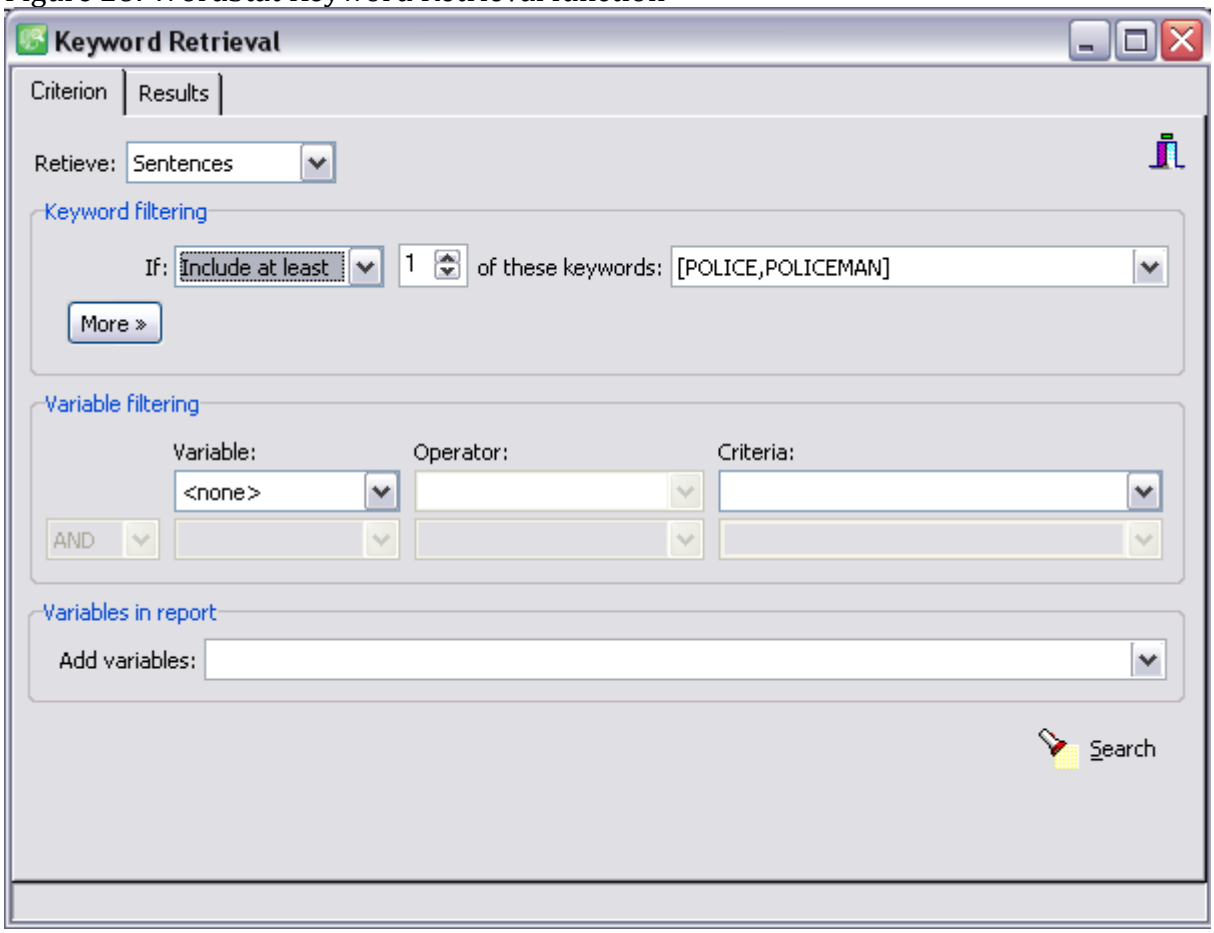


TIP: The extraction of phrases is particularly sensitive to the application of an exclusion dictionary and/or substitutions (such as lemmatization) as set on the Dictionaries tab. With this data the act of turning on the default exclusion dictionary (part of which can be seen in the bottom left corner of Figure 27) reduced the number of phrases from 407 matching the search parameters (indicated in the bottom bar of the screen) to just 16 phrases. And lemmatization made many phrases much harder to interpret as grammatical stems replaced many words. So care should be taken when those functions are combined with the phrase finder tool.

It is now possible to do the coding work within WordStat, building on the inductive derivation of coding themes described above. Although there are several different ways of analysing the texts in WordStat there is only one with a link to the coding function. This is to be found on the Frequencies page, by using the Keyword Retrieval button (the magnifying glass icon, sixth from the left), and generating a report similar to that of the Text Retrieval in the main program. An alternative way to open this is to right click on a word in the Frequencies list and select *Keyword retrieval* from the context menu, this jumps straight to the Results screen for the selected word but the criterion can still be adjusted as described below.

Figure 28, below, shows this Keyword Retrieval with a simple filtering criterion. Note that the keywords should be selected using the pull-down menu, which accesses the word list shown in the frequencies window (although only in alphabetical order). Complex and sophisticated combinations of keywords can be used to generate reports with this function. However the results screen, which is activated when the Search button is clicked, has facilities to add or select codes, remove single items from the report, and apply the selected code to individual items or the whole report list in ways similar to those described for the Text Retrieval report above. The only difference is that it is not possible to work interactively between this report and the QDA Miner main screen in order to apply codes manually to carefully selected individual passages of text, these codes have to be applied to the whole sentence or paragraph (depending on the Retrieve setting on the Criterion screen).

Figure 28: WordStat Keyword Retrieval function



It is unlikely that a single search will exhaust the potential autocoding for one code, and the process can be repeated with variations on the search theme. Where several different words can be identified as relating to a particular coding theme they can be used in separate retrievals or combined in complex ones.

The autocoding process will not complete the task if data reduction to accurate quantities of references is the goal of the analysis (see section 4 below). There will often be responses which relate to a particular theme without using any of the main keywords associated with that theme. So a combination of autocoding and human interpretation is needed to achieve a high level of accuracy. However, when dealing with extremely large numbers of responses in a large scale survey that degree

of accuracy may not be necessary and the automated procedures may deliver all that is required, provided they are used with imagination and thoroughness.

It is suggested that the word frequency and phrase finder tools may be used in an exploratory way to generate ideas for themes and codes from within the data. This is unlikely to be a linear process, instead requiring a lot of movement between different views of the data. It will often be useful to view a selected word or phrase in all of its contexts, which can be done from the menu available with a right click, and the context displays can be sorted by case number, keyword and before, or keyword and after (these latter two sorts referring to the preceding or succeeding words beside the keywords). This can be very helpful in identifying whether a word or phrase has been used with a consistent meaning or not. When such themes have been identified they may be noted, with their distinguishing words or phrases, for subsequent coding back in QDA Miner, or alternatively they may be autocoded directly from within WordStat as explained above.

2.5 Coding – data indexing versus data reduction.

The actual techniques of manually applying codes to segments of text are not discussed here. They are common to all applications of the program and are clearly explained in QDA Miner's help manual and in other sources. However, the possible uses to which the analysis of responses to open-ended survey questions may be put is a matter worth discussing further.

As a coding scheme is developed and applied to textual data, the analyst will inevitably encounter uncertainty and doubt. Does the text in front of me represent something different from others I have read before which mentioned a particular keyword? A common solution to this is to be generous and inclusive, applying specific codes to a range of comments that initially appear to be connected to those concepts, with the good intention of returning later and checking the work. This activity may be described as "data indexing" as it facilitates the retrieval of various passages that appear to relate to a particular topic.

When open-ended questions have been asked in survey situations it may be anticipated that the analyst will often be asked to generate numerical summaries of the data, probably in the form of statements of the type "X% of responses to this question mentioned Y". The obvious source of the numbers for this output is the coding of concept "Y". However the statement will only be valid if the use of that concept in every one of the responses allocated that code is consistent and equivalent, because the code that is used in this way has effectively replaced the words recorded for each respondent. The original textual data has been reduced to the code label.

When put this way it should be apparent that work needs to be done by the analyst to refine the inclusive indexing codes before they can be safely used as summarising reducing codes. In this example data one respondent answered the question about the advice they had received by saying "... move self and belongings upstairs. Contact floodline. I think there were a few tags to put on things in the kitchen. The pack itself was very well done and well thought out ..." while another dismissed this as "... a flood pack stating obvious, silly stickers ...". Initial index coding may have allocated a code "Given flood pack" to both of these passages but it could be potentially misleading to include both in a percentage of respondents who referred to the flood packs as though these comments are equivalent to each other.

It may be anticipated that more use will be made of automatic coding procedures in QDA Miner than other CAQDAS programs because it has such sophisticated tools for analysing text, and it will process large volumes of textual data quickly with those tools. In such circumstances the analyst needs to be aware of the risks of incorrectly applied codes but, where the numbers are large, a few errors will have little impact on percentage statistics so the concern needs to be kept in proportion.

2.6 Checking summarising codes – consistency and omissions.

There are a variety of tools in QDA Miner to assist with the refinement of codes when they have to be reduced to summarise what was originally said. Two particular aspects should be considered, firstly confirmation that all of the passages connected to any one code are all sufficiently similar to be treated as equivalent, and secondly confirmation that no other passages that are also equivalent have been omitted from that code.

The first step in confirming consistency or equivalence is to extract all of the passages that have been allocated to a code and check them carefully. This can be done visually in QDA Miner or with further software assistance in WordStat. Run the command *Analyze / Coding Retrieval* and select the relevant document and code to be checked from the drop-down menus, then click on the Search button. It is likely that you will start to read the coded texts looking for any that seem different from the rest but, if the resulting number of hits is very large, it may be more practical to use WordStat to analyse this set of responses. It is possible to jump into WordStat directly from many of the report screens in QDA Miner and the subsequent content analysis will be carried out on only the texts that make up that report, the icon to do this is a magnifying glass within the report window. It is not necessarily easy to identify the rare items in a set which do not belong there by using content analysis tools which were designed to identify the most common themes. However, using the *Keyword Retrieval* (via a right click and context menu) from the Frequencies page on the most commonly found words should pull up subsets of the data with high degrees of consistency. Similar searches can be run on some of the least frequently used words to check that these are not out of place.

After checking all of the passages linked to a single code it should be possible to write a concise definition of that code, possibly incorporating the most frequently used keywords from the above WordStat frequencies page. This definition may be typed into the description field for that code by selecting *Edit Code* from the context menu when you right click on the code label in the Codes panel of the main QDA Miner screen. If you find it difficult to write a concise definition of a code then it may be inferred that you should not refer to the number of references to that code in any data reducing statements.

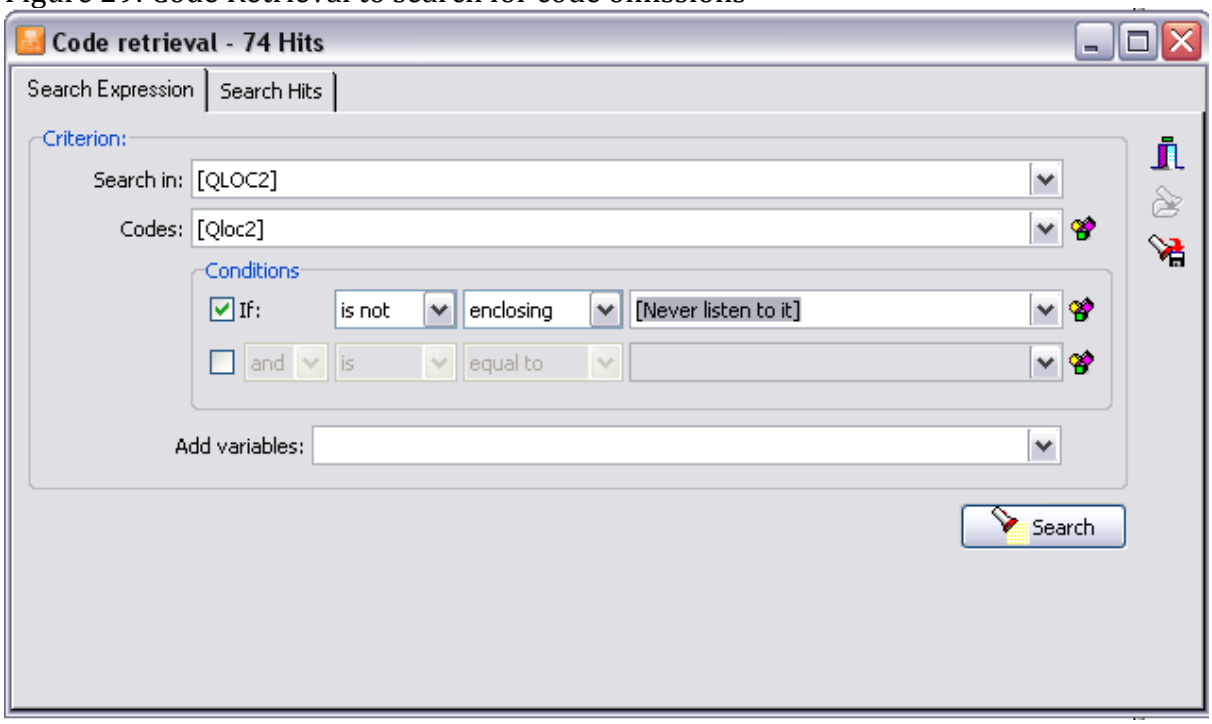
It is more difficult to search for code omissions, these are passages which are closely equivalent to those already allocated to a particular code but which have not yet been allocated themselves. One possible method for doing this is to filter-out all of the passages which have been allocated to the code and then carry out a content analysis on the remainder using key words connected with that code.

There is no simple query that allows one to search for all responses to one question that have not been allocated to one code, but it is possible to create a neutral code and apply it to all of the paragraphs in a document variable and then one can search for all the neutrally coded paragraphs that do not enclose the code of interest. So, in this example, a text retrieval was run to select all paragraphs in document QLOC2 (without any other selection criteria). This found 91 hits (or response paragraphs). A simple code “Qloc2” was created and applied to all of these hits using the “Code all hits” icon in the

report window. Then a code retrieval report was run as shown in Figure 29, using the code "never listen to it" as the code of interest. There were 17 positive allocations of this code, so the number of responses without it was expected to be 91 – 17 = 74 and, as Figure 29 shows, that many were identified by this report.

The results of this retrieval can be scanned visually for passages which may match the definition for the code of interest. These would represent errors or omissions from that code.

Figure 29: Code Retrieval to search for code omissions



Once again, by clicking on the magnifying glass icon within the Code retrieval search hits window, it is possible to move this subset of the data into WordStat and use that module to identify possible errors or omissions for this code. Now, if the coding has been done with reasonable care so far, the items of interest may be expected to be found in the lowest frequencies so an alphabetical listing of keywords may be useful to prove the satisfactory absence of the main terms.

2.7 Looking for similarities or differences?

When analysing the responses to open ended survey questions it may well be easy to slip into the expectation that the most frequently used codes, or rather the concepts to which they refer, are the most important. After all, these are the items that seem to have the most statistical 'weight'. However, it should always be worth looking out for contributions which are different from the common ideas. One-off comments will never feature in the quantitative tables because, by definition, they lack numerical support. But a small number of individuals may well take the opportunity of an open-ended question to add an unexpected thought and these contributions represent a challenge and an opportunity for the analyst.

Perhaps it is worth asking yourself what was the purpose behind the inclusion of an open ended question in the survey. In many situations previous research will have revealed the most likely

answers and these will have been included as response categories in closed questions asked elsewhere in the survey, but then an open question has been included to pick up other ideas. In these situations it is the unusual answers which may be of most interest.

For example in the data used as an example for these instructions the question QADVI2 was used to ask respondents what advice they had been given in order to prepare themselves for an impending flood about which they had been warned. It may be interesting to note that out of 324 responses to this question, just three people mentioned warm clothing or blankets. It may be the case that for most people the need for warm clothing as you sit out a flood in an upstairs room is too obvious to be worth mentioning, but this may also be a clue that there was a potentially significant gap in the advice actually given to the flood victims in the incidents under consideration. It seems that the value of a detailed qualitative analysis of the responses to such a question is an opportunity to pick up the unexpected ideas which would be so easily overlooked in a statistical analysis.

Those using QDA Miner to analyse this sort of data may find locating these unusual responses particularly challenging as the design of this program is so geared to identifying commonalities in the texts. It is probably a good idea to set up a special code to be used to index any such rarer ideas that are noticed during the course of the analysis. In all probability this code would be allocated manually whenever a response is found which seems to be outside the common themes. Then, at a late stage in the analysis, a report could be generated to output all the texts allocated as "unusual" for manual review and consideration. This should provide a counterweight to the degree of automation suggested in the guidance above.

3.0 Quantitative Analysis Strategies for analysing Open-ended Survey Questions in QDA Miner

In common with other pages in this section of the website, this page is a series of observations about how the features of QDA Miner might interact with a particular sort of dataset. This page should be read in the context of the related materials concerning the use of QDA Miner with Open-ended Survey Questions, in particular the Data Preparation Instructions and the Qualitative Analysis Strategies, since the quantitative strategies outlined below can only be effected after the data have been imported and coded systematically in a QDA Miner project.

The tools discussed below are illustrated with examples from the same post flooding event survey that was used to illustrate the data preparation processes. For a summary of the project from which this data derives see here. This data is characterised by a fairly large number of short statements.

Outline:

Detailed Guidance:

3.1 Summary of codes applied to a single question – Coding Frequency

In many situations the most important output required from the analysis of some open-ended survey question responses will be a simple summary of the number of times each thematic code has been applied to the set of responses to a single question. In QDA Miner this can be achieved easily with the Coding Frequency routine.

The option can be found at *Analyze / Coding Frequency…* which opens a new window listing all of the codes in the project. Initially the program default is to list all of the documents (ie survey questions) in the "Search in:" field near the top of the window but, by pulling-down the menu list and clicking to remove the ticks from all of the unwanted questions, this can quickly be changed to a single document. No frequency counts are carried out until the search button (a torch or 'flashlight', as throughout the QDA Miner suite of programs) is clicked. Figure 30, below, illustrates some of the output window at an intermediate stage in the analysis of our example data.

Figure 30: Coding Frequency output

Coding Frequency

Search in: [QMORE] Multiple lines

Category	Code	Description	Count	% Codes	Cases	% Cases
Check Codes	Qloc2	All responses to QLOC2				
Check Codes	Qmore	All responses to QMORE	361	58.9%	361	28.7%
Check Codes	Qval2	All responses to QVAL2				
More Advice	Any	Comments about not having received any advice e	51	8.3%	51	4.1%
More Advice	Authorities	Comments on sources of advice and help	34	5.5%	34	2.7%
More Advice	Earlier	Comments about the timing of advice, the sooner tl	33	5.4%	33	2.6%
More Advice	Emergency Measures	Comments on emergency actions to be taken by h	67	10.9%	67	5.3%
More Advice	Evacuate	Comments about evacuating flooded properties	7	1.1%	7	0.6%
More Advice	Health and Safety	Comments on practical health & safety suggestion	8	1.3%	8	0.6%
More Advice	Media	Comments about advice in public media	24	3.9%	24	1.9%
More Advice	Personal	Comments about personally delivered warnings ar	28	4.6%	28	2.2%
More Advice	Risk	Comments on advice to help residents assess thei				
More Advice	Signs	Comments about visual warnings and advice				
More Advice	Specific	Comments about specific warnings and advice, pe				
More Advice	Vulnerable	Comments about vulnerable people				
No Source	At work					

The first three columns in Figure 30 are visible before the search has been carried out, being a simple list of the codes with their categories and descriptions. The search generates the quantitative data columns that then appear to the right including, not shown here, further columns for number of words, and percentage of words. The table can be re-generated for different question documents or combinations of documents by altering the values in the "Search in:" field and pressing the "Search" button again.

Some explanation of the interpretation of the quantitative columns may be helpful. In this data we applied the thematic codes in category group "More Advice" to the whole paragraph of each applicable response (because these represented meaningful units of data) and so there is no difference between the count of code frequencies and the count of case frequencies. However it is possible to imagine situations where mention of particular words may be counted in the responses by applying the thematic codes to single words or sentences and then the "Count" and "Cases" columns would show different values where some respondents used those words multiple times. The "% Codes" column shows the percentage of the total code frequencies in the current report for each code (there are 613 code applications in QMORE at this point and the 51 uses of code "Any" represent 8.3% of that number). The "% Cases" column shows the case frequency for each code as a percentage of the total number of cases in the whole project (in this example 1,257 cases, so the 51 cases with the code "Any" represent about 4.1% of 1,257). Thus it can soon be observed that when you alter the document parameters in the "Search in:" field the percentages of codes alter while the percentages of cases do not (if you have applied some codes to more than one document then all counts and percentages may change with the parameters).

The Coding Frequency table can be printed or exported to a spreadsheet by using the appropriate buttons on the toolbar within the window shown in Figure 30.

3.2 Filter Cases

QDA Miner has powerful functions to use one or more variables to extract data for subsets of the survey sample and, when these filters are applied they will alter the counts and percentages in the outputs. This is illustrated in Figure 31, below, where the Coding Frequency window was closed, the menu option *Cases / Filter* was used to restrict the display to those cases in age-groups 55 to 64, 65 to 74 or over 75 and then the Coding Frequency was re-run on the QMORE document alone. As can be seen, when the numbers are compared between Figures 30 and 31, a different set of statistics has been generated with these parameters.

Figure 31: Repeat coding frequency table after filtering to older age-groups

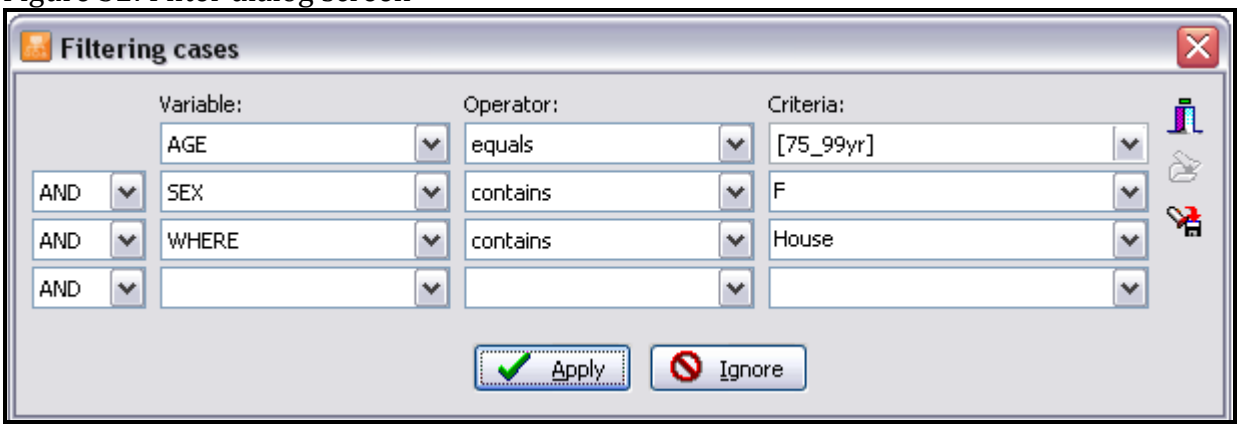
Coding Frequency

Search in: [QMORE] Multiple lines

Category	Code	Description	Count	% Codes	Cases	% Cases
Check Codes	Qloc2	All responses to QLOC2				
Check Codes	Qmore	All responses to QMORE	152	57.6%	152	24.5%
Check Codes	Qval2	All responses to QVAL2				
More Advice	Any	Comments about not having received any advice a	20	7.6%	20	3.2%
More Advice	Authorities	Comments on sources of advice and help	14	5.3%	14	2.3%
More Advice	Earlier	Comments about the timing of advice, the sooner tl	17	6.4%	17	2.7%
More Advice	Emergency Measures	Comments on emergency actions to be taken by h	32	12.1%	32	5.2%
More Advice	Evacuate	Comments about evacuating flooded properties	3	1.1%	3	0.5%
More Advice	Health and Safety	Comments on practical health & safety suggestion	3	1.1%	3	0.5%
More Advice	Media	Comments about advice in public media	10	3.8%	10	1.6%
More Advice	Personal	Comments about personally delivered warnings ar	13	4.9%	13	2.1%
More Advice	Risk	Comments on advice to help residents assess thei				
More Advice	Signs	Comments about visual warnings and advice				
More Advice	Specific	Comments about specific warnings and advice, pa				
More Advice	Vulnerable	Comments about vulnerable people				
No Source	At work					

Of course, it is possible to crosstabulate a set of codes against all of the values for a specific variable (such as age) but the filter function may be useful when the impact of two or more variables is being explored. For example Figure 32, below, shows a more complicated filter combining specific values for three separate variables, a combination that identified 11 of the 1,257 cases in this dataset.

Figure 32: Filter dialog screen



The Boolean operators "AND" and "OR" are the available options in the left-hand fields. If the variable has been set up as a nominal/ordinal type (as "AGE" has in this example) then the operator can be selected from "equals", "does not equal", "is empty" or "is not empty". If the variable has been set up as a string type (as "SEX" has in this example) then the operator can be "contains", "does not contain", "is empty" or "is not empty" and there will be no suggested values in the pull-down menu under "Criteria:" so you have to type in the necessary details.

TIP: It is possible to change a variable's type from 'string' to 'nominal' in order to ensure that the set of values it can take is readily available in the filter dialog. Right click on the value of the variable that you want to change in the variables section of the QDA Miner main screen, and from the context menu select "Transform XXXX" (where "XXXX" will be replaced by the name of the variable you have selected to alter) and then "String -> Nominal" from the sub-menu. At the next dialog box it is possible to click on the radio button for "Overwrite existing variable" and all of the strings will be converted to their equivalent values in a nominal list. If you are uncertain about the consequences of this it would be worth creating a security back-up file before the transformation because it is a change which cannot be undone through the program, alternatively use this routine to create a new variable of nominal type without overwriting the existing data.

When you apply a case filter in this way you should notice that the background colour of the cases panel changes to a light blue, and the heading of that panel changes to "CASES: (Filtered: xxx/yyyy) with figures instead of xxx and yyyy).

TIP: Don't forget to remove the filter before moving on to other analyses by reopening the *Cases /Filter* option and clicking on the "Ignore" button.

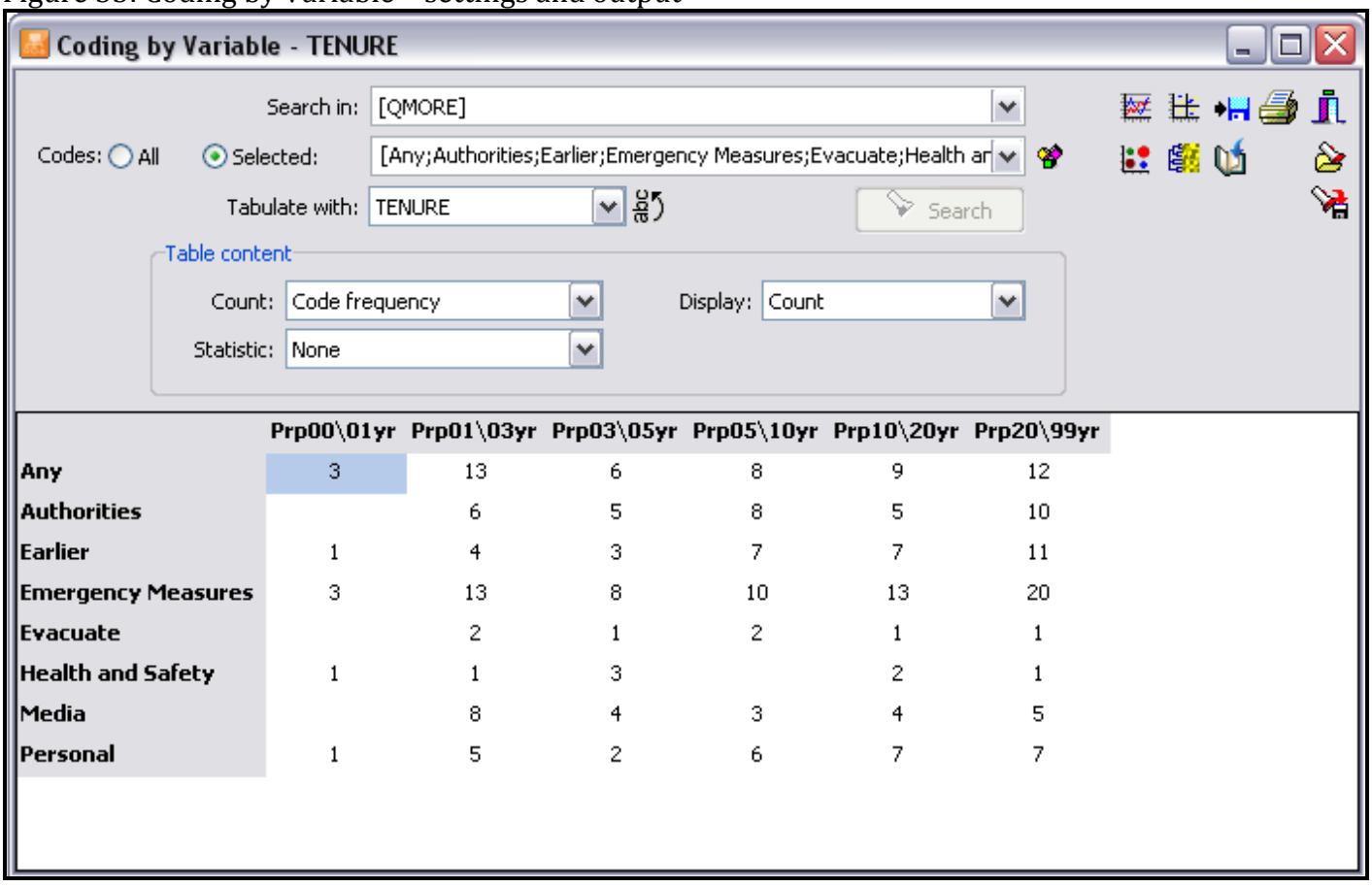
3.3 Coding by Variable - crosstabulation

The Coding by Variable routine is useful for exploring whether the codes applied to a question or a group of questions appear to vary systematically with the values of an attribute variable. For example

in our data we were interested to know whether the people who had lived in the home at risk from flooding for a long time responded differently in this survey to those who had only recently moved-in (this used the variable called "tenure").

Start the routine from the menu option *Analyze / Coding by Variables...* . Figure 33, below, shows both the settings and the results of the routine. The parameters for the query are entered using pull-down menus to select one or more specific question documents containing the data to be analysed, one or more groups of thematic codes to be shown in the rows of the table, and one attribute variable whose set of values form the columns in the table, followed by a click on the "Search" button. Here the question "QMORE" is being analysed, the selected thematic codes are "Any; Authorities;Earlier; etc" and the variable is "TENURE". The button just to the right of the "Tabulate with:" field is a toggle to turn the column labels sideways in order to reduce column widths and display more columns in the available space.

Figure 33: Coding by Variable – settings and output



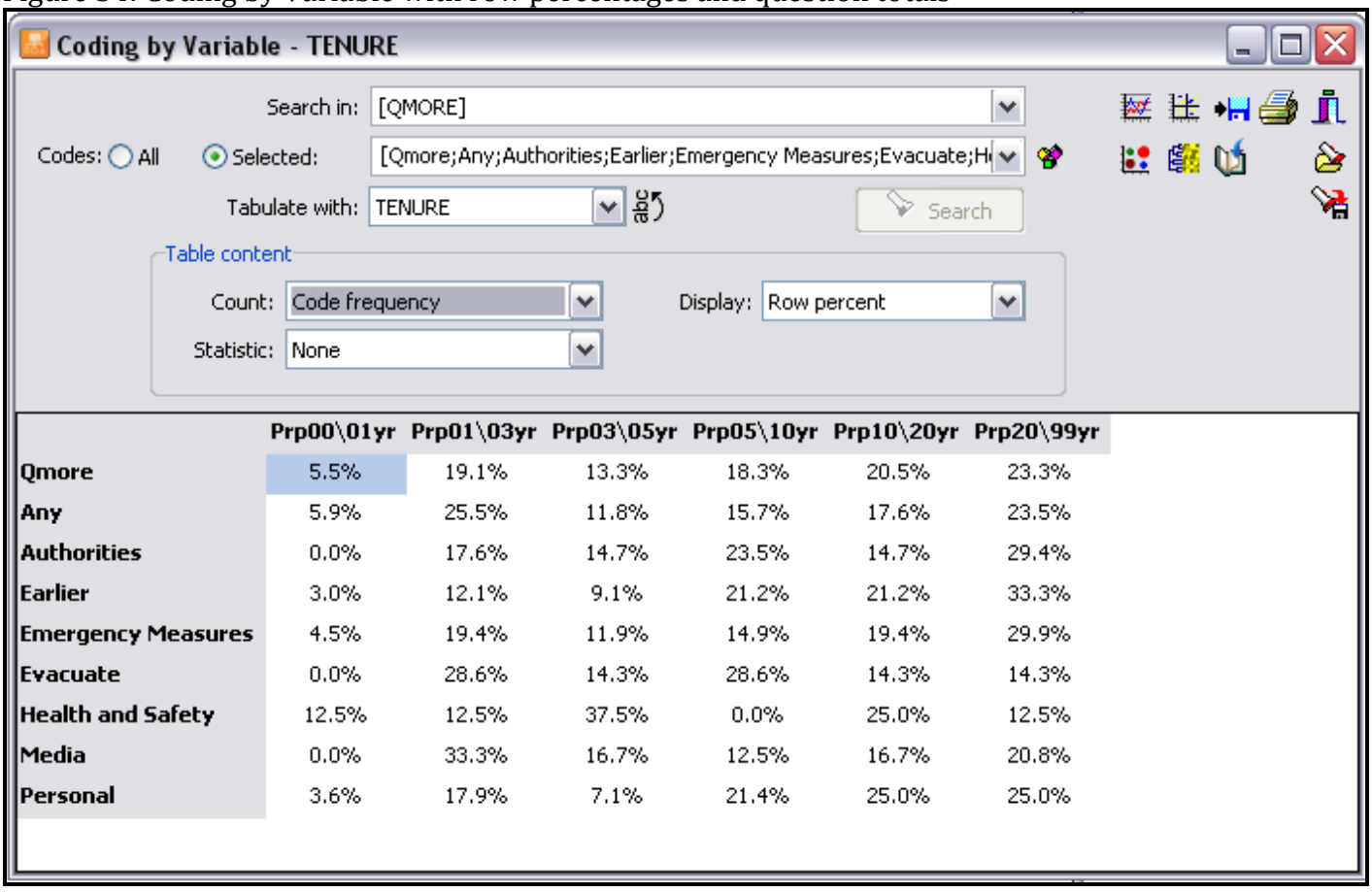
	Prp00\01yr	Prp01\03yr	Prp03\05yr	Prp05\10yr	Prp10\20yr	Prp20\99yr
Any	3	13	6	8	9	12
Authorities		6	5	8	5	10
Earlier	1	4	3	7	7	11
Emergency Measures	3	13	8	10	13	20
Evacuate		2	1	2	1	1
Health and Safety	1	1	3		2	1
Media		8	4	3	4	5
Personal	1	5	2	6	7	7

As can be seen in Figure 33, the selection parameters remain visible in the top part of the window. They can be changed directly, for example to explore a different attribute variable, but the "Search" button has to be clicked to refresh the window whenever those settings are altered. Conversely, the "Table content" section of parameters alters the display whenever any of these are changed, so it is straightforward to move between counts and row or column percentages using the "Display:" pull-down menu.

In the example shown in Figure 33 it is difficult to know if the low counts in the first column (those who had lived least time in the flood-risk home) signifies satisfaction with the advice received or

merely a small sample size for that category. For this reason it may be more useful to include the code which has been applied to every response to this question (a code called "Qmore" which was used to help list uncoded responses at an earlier stage) and then by comparing the row percentages for each theme with the row percentages for the "Qmore" code any important differences become more apparent. This is shown in Figure 34 below.

Figure 34: Coding by Variable with row percentages and question totals



	Prp00\01yr	Prp01\03yr	Prp03\05yr	Prp05\10yr	Prp10\20yr	Prp20\99yr
Qmore	5.5%	19.1%	13.3%	18.3%	20.5%	23.3%
Any	5.9%	25.5%	11.8%	15.7%	17.6%	23.5%
Authorities	0.0%	17.6%	14.7%	23.5%	14.7%	29.4%
Earlier	3.0%	12.1%	9.1%	21.2%	21.2%	33.3%
Emergency Measures	4.5%	19.4%	11.9%	14.9%	19.4%	29.9%
Evacuate	0.0%	28.6%	14.3%	28.6%	14.3%	14.3%
Health and Safety	12.5%	12.5%	37.5%	0.0%	25.0%	12.5%
Media	0.0%	33.3%	16.7%	12.5%	16.7%	20.8%
Personal	3.6%	17.9%	7.1%	21.4%	25.0%	25.0%

In Figure 34 the highlighted number shows that only 5.5% of the responses to this question were made by people who had lived for less than one year in that home, so the low counts in that column are largely explained by the small sample size. The code labelled "Any" does not show any clear pattern of difference from the "Qmore" code, so those who felt they received no advice were probably evenly spread across this tenure variable. But there may be some pattern for the "Earlier" code where it appears that the newest residents mentioned this rather less often than the longest residents (in the first three columns the percentage in this row is lower than in the "Qmore" row, whereas in the last three columns it is higher).

Several options to alter the visualisation of this type of result are accessible through icons in the top right corner of the window shown in Figure 34, notably bar charts and bubble plots. These may be helpful sometimes to see patterns visually when they are hard to see in the mass of numbers. It is easy to explore these with your own data so we will not describe them here, but they work interactively with whatever data is displayed in the Coding by Variable window.

TIP: There are also possibilities of applying much more sophisticated statistical techniques to the data within this routine. The second icon on the lower row in Figure 34 is labelled "Heatmap" when

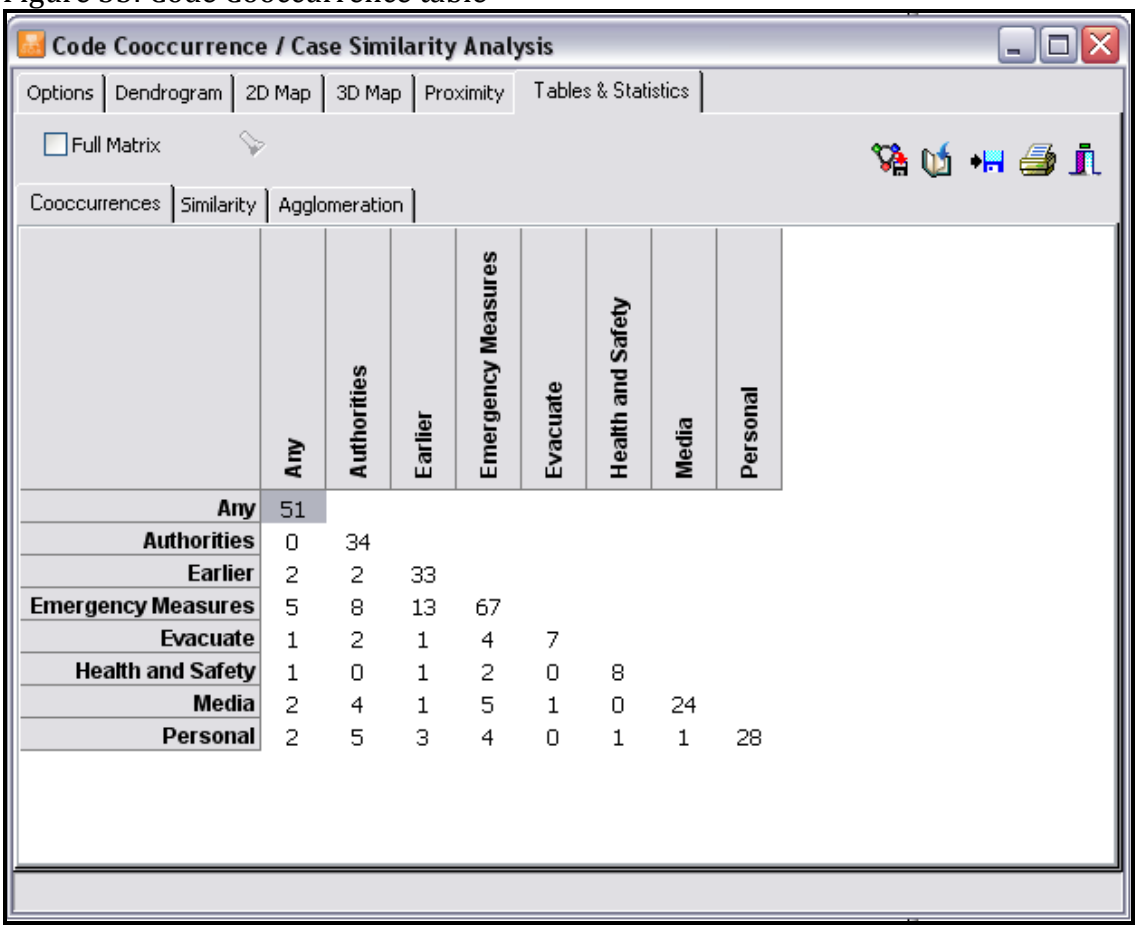
the mouse pointer is hovered over it. This opens a number of possibilities including dendograms to represent a cluster analysis of the codes or variables or both. The icon above it in the top row is labelled "Correspondence plot" and this can generate a three dimensional space to represent the data. Whether these have meaning with your data will be determined largely by how accurately the data collection process captured the respondents' language and how 'richly' that language describes the issues being studied. These are important tools in content analysis but they should be used with care and there is insufficient resource to describe their use here.

3.4 Code Cooccurrence

The final quantitative technique to be described in this section is the tabulation of co-occurrences between different thematic codes. QDA Miner has many options and settings to explore this both statistically and graphically, we shall only consider these at a basic level here. Essentially this is a matter of looking for combinations of two thematic codes which have both been applied to responses disproportionately often or rarely. In addition, by applying a case filter for some attribute variable, it is possible to explore such thematic combinations in the context of that variable.

Figure 35, below, shows a simple output table for eight thematic codes applied to the same set of responses to a question about better advice prior to a flooding episode. In this program the same codes are always shown on both axes of the table so the area beneath the diagonal is sufficient to display all of the possible combinations without duplication. The figures on the diagonal represent the total frequencies for the codes, so in Figure 35 the code "Any" has been applied 51 times (this matches the count shown in Figure 30, above).

Figure 35: Code Cooccurrence table

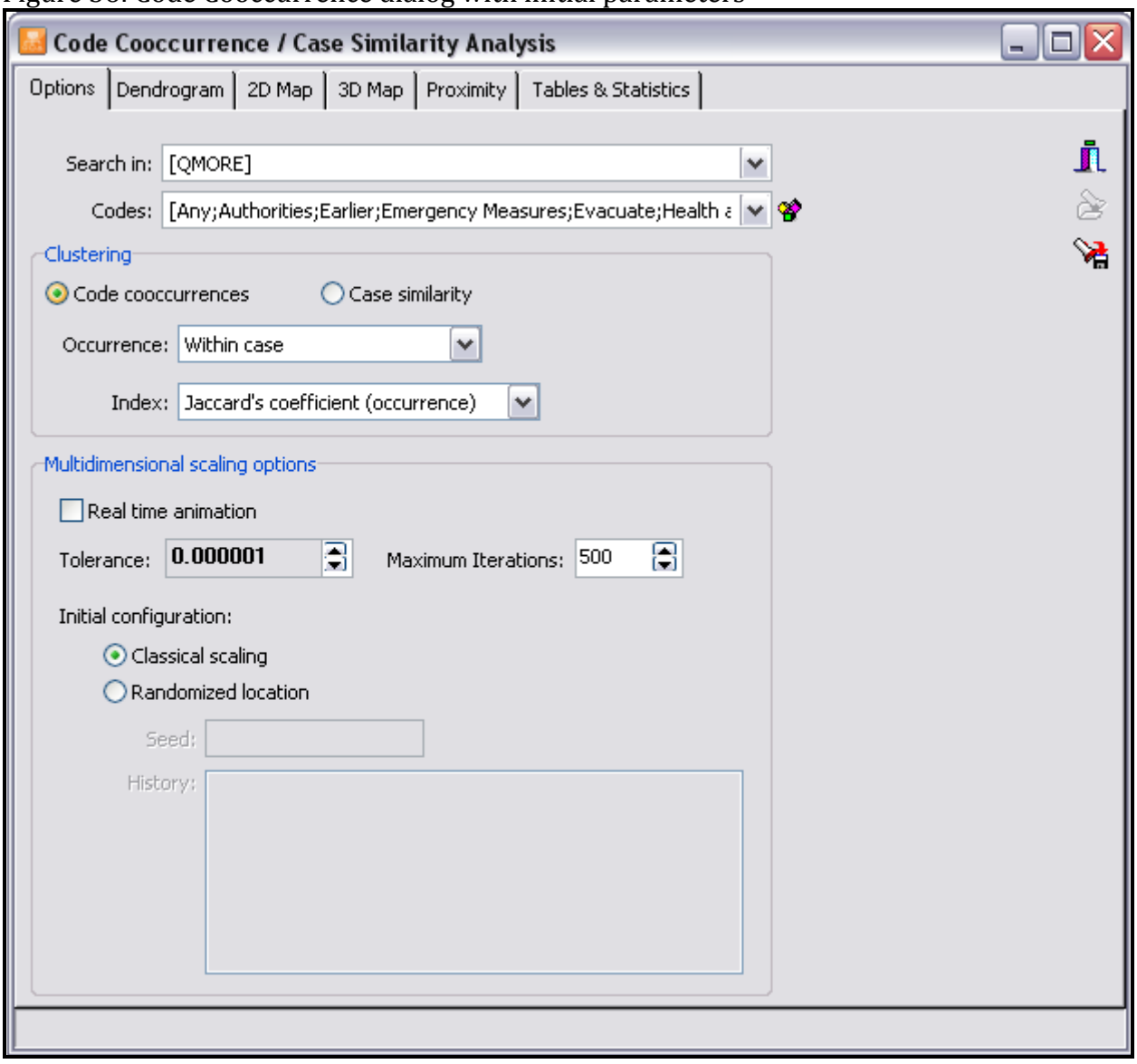


	Any	Authorities	Earlier	Emergency Measures	Evacuate	Health and Safety	Media	Personal
Any	51							
Authorities	0	34						
Earlier	2	2	33					
Emergency Measures	5	8	13	67				
Evacuate	1	2	1	4	7			
Health and Safety	1	0	1	2	0	8		
Media	2	4	1	5	1	0	24	
Personal	2	5	3	4	0	1	1	28

The largest co-occurrence showing in Figure 35 is that of 13 instances where “Emergency Measures” and “Earlier” have both been applied to the same response. It may also be noted that the code “Health and Safety” never occurs in the same response as “Authorities”, “Evacuate”, or “Media”. This is not the place to discuss what those results might mean in substantive terms, they are merely used as illustrations of how one might read the table.

The table in Figure 35 was generated with the parameters illustrated in Figure 36, which shows the dialog box that opens when you select the option *Analyze / Coding Co-occurrences...* on the main screen.

Figure 36: Code Cooccurrence dialog with initial parameters



The parameters are entered on the page with the “Options” tab label at the top of this window, the results table appears under the tab “Tables & Statistics”. Select one or more appropriate documents at the “Search in:” field using the pull-down menu and check boxes, and then select the set of codes of interest at the “Codes:” field using either the pull-down menu or the tree menu (just to the right). Note that there is only one place to select the codes as they will be shown on both axes of the table, and that

any hierarchical structure will not be shown in the table as the selected codes will be shown in alphabetical order. If you have always applied the codes to the entire response from any participant in the survey then the default clustering settings of code co-occurrences within cases will probably be appropriate.

TIP: Note that you can define what you mean by 'co-occurrence' through the pull-down menu in the "Occurrence:" field, the default "within case" shown here is the broadest definition. If you have applied codes to carefully selected segments within the responses then the other options here may be useful.

TIP: As with some of the other routines discussed above, if your data is particularly 'rich' in language or content and has been captured accurately then some of the more sophisticated content analysis tools appearing on the remaining tabs in the Code Co-occurrence dialog box ("Dendogram, 2D Map, 3D Map, Proximity") may be useful.

3.5 Summary conclusion

Many qualitative analysts, who would be familiar with these CAQDAS programs, may be reluctant to use quantifying statements about the data that they analyse. This would be for the very good reason that most often the size of the sample that they are working with is too small to justify a numerical conclusion. However, when one is working with survey data then there is more likelihood that the sample size is sufficiently large and sufficiently random to support some quantified statements.

At the same time many quantitative analysts may be wary of applying statistical techniques to data which has been coded separately from the data collection process. However, provided the data collection process was sufficiently robust to capture the data accurately, then the fact that the codes have been applied by research analysts rather than the respondents themselves should not be a reason to water-down the analysis processes. Because the coding process is transparent and replicable, the coded data may be used with more confidence than that derived from the 'amateur' coders of the respondents to closed questions.

Thus, if quantitative conclusions are being drawn from the rest of the survey data, then there should be no reason why quantitative conclusions cannot be drawn from the open-ended question responses as well.

4.0 Export Data to Statistical Packages after analysing Open-ended Survey Questions in QDA Miner

In common with other pages in this section of the website, this page is a series of observations about how the features of QDA Miner might interact with a particular sort of dataset. This page should be read in the context of the related materials concerning the use of QDA Miner with Open-ended Survey Questions, in particular the Data Preparation Instructions and the Qualitative Analysis Strategies, since the export strategies outlined below can only be effected after the data have been imported and coded systematically in a QDA Miner project.

The tools discussed below are illustrated with examples from the same post flooding event survey that was used to illustrate the data preparation processes. For a summary of the project from which this data derives see here. This data is characterised by a fairly large number of short statements.

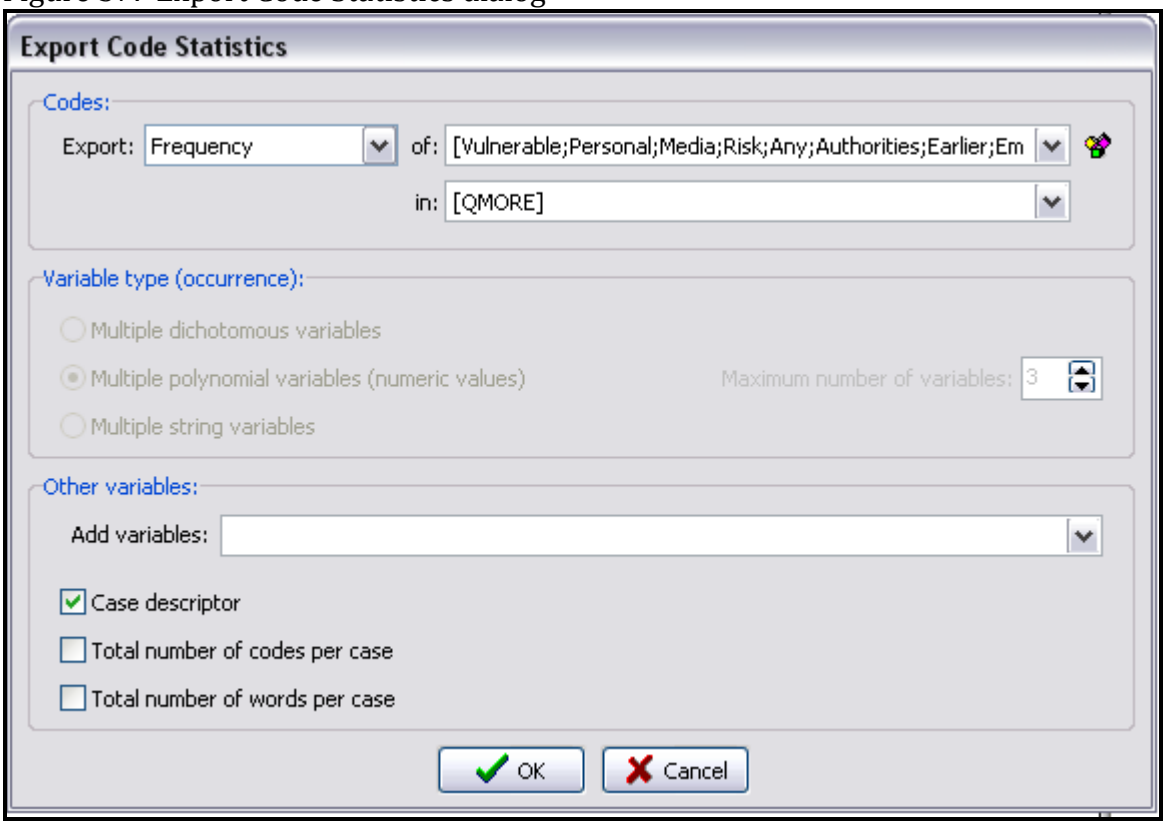
Outline:

Detailed Guidance:

4.1 Project Export Code Statistics

Creating a Microsoft Excel file with the count of code frequencies for each respondent is quite straightforward in QDA Miner as there is a specific routine designed for this task. It is found in the menu option Project / Export / Code Statistics and this brings up the dialog box shown in Figure 37.

Figure 37: Export Code Statistics dialog

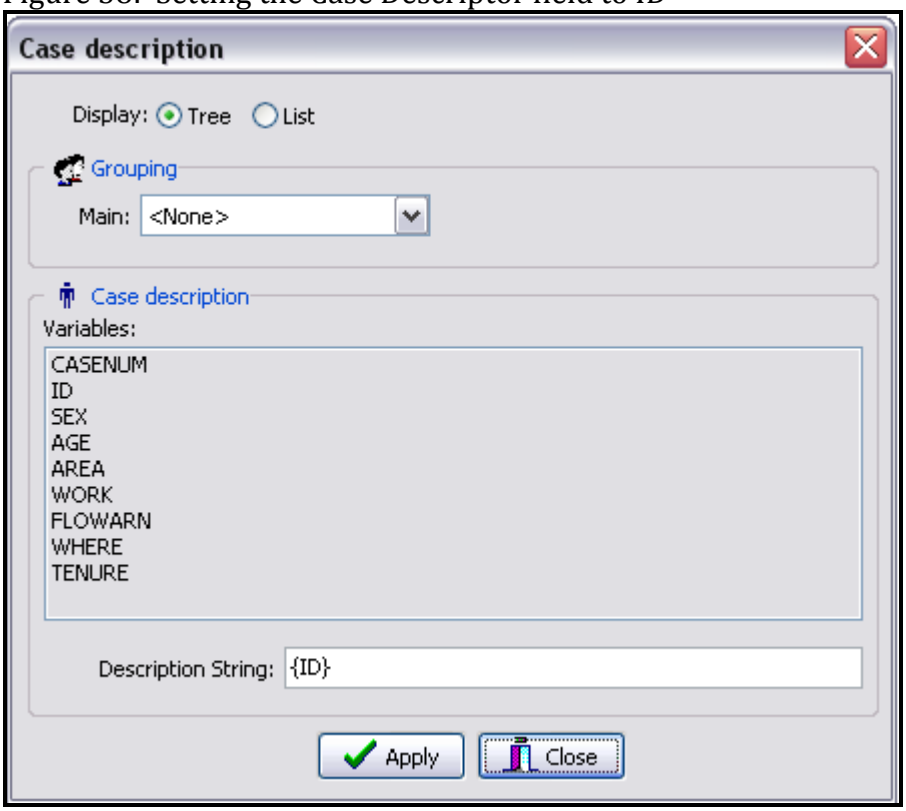


In Figure 37, above, the necessary fields have been completed for the task of exporting all the thematic codes applied to the question document "QMORE". The alternative options for the "Export:" field are "Occurrence, Word count, and Rate per 1000 words" – with the way we have applied relevant codes only once to each response it is likely that occurrence and frequency would generate identical results of 0 or 1 for each cell in the table. In the second field a pull-down menu offers a full list of codes but the

alternative way, using the colourful icon to the right, offers a hierarchical table making it easier to select a group of codes by ticking the box of its header item. One or more question documents can be selected with the pull-down menu for the field beside "in:", on this occasion we have restricted the export to a single document variable.

An important field is to tick the "Case descriptor" check box as this is the setting that will generate the case identifier label for each row in the output table. However, before proceeding to run the procedure it is worth checking that the case descriptor in use is as helpful as possible, because this item can be set to many different values by the user. What will appear in the spreadsheet table as the label for each row is the description for each case currently appearing in the CASES panel of your QDA Miner working screen. To adjust the descriptor use the menu option *Cases / grouping/descriptor...* which brings up the dialog illustrated in Figure 38 below.

Figure 38: Setting the Case Descriptor field to ID



For this purpose the case description that is needed is one that includes the number by which respondents are identified in the statistics package, so that the new frequency variables can be matched with the other quantitative data associated with each case. In our example the "ID" variable is the one needed, so that has been entered between braces in the "Description String:" field. Note also that the setting of "<None>" is the one required in the "Grouping" field for this output.

Returning to the Export Code Statistics dialog, when you are satisfied that the correct settings are in the necessary fields, click on the "OK" button to run the routine. You may notice a blue progress bar in the bottom left corner of the QDA Miner main screen but this procedure runs very quickly. When the calculations are complete a standard MS Windows dialog appears for you to enter a file name and location to store the data in MS Excel (*.xls) format.

4.2 Editing in MS Excel

When you open the file that you have just created in Excel, the only editing that you may need to do is to clean-up the case identifiers in order to make them match those already in the statistics database. In our example we had the word "Case" as a prefix to each unique ID number and needed to remove this. A quick way of doing this for a large number of cases in Excel is described below.

The first step in cleaning up the case identifiers is to insert two new columns between the current column A and the first column of code data, ie between columns A and B in the workbook. The corrected IDs will be created in these new columns. In our example the required data are the last five digits of the text in column A, and these can be extracted by typing the logic =RIGHT(A2,5) into the cell at B2 (in the new empty column just previously inserted) and then copying that logic all the way down column B. It will then be necessary to convert these logical statements that display the correct digits into absolute values in the new column C. This final step involves copying all of column B and then pasting it into column C with a "paste special" command to store "values" only. When this has been completed and checked for accuracy, columns A and B can be deleted and the workbook can be saved.

The column headers should not need any editing as QDA Miner exports the plain code name without any additional text or prefix. Provided these names are unique in your database they should be an adequate basis for the variable names in the statistics program.

4.3 Checking data transfer

Whenever large amounts of data are moved between programs there will be potential sources of error so, before continuing with the analysis in your statistical package, it would be advisable to check a small sample of cases in that package to confirm that the code frequencies have been matched with the correct cases. This is not simply a matter of confirming that the total frequency for each imported code variable is the same in QDA Miner and your statistical program but also that individual cases have correct codes.