

# **Tutorial**

## **“Immunization”**

**N6**

# Introduction

This tutorial provides a detailed walk-through of the N6 software. For a tutorial handling more unstructured data and a quicker overview of the software, please see the accompanying "Bush Schooling" tutorial. If you are familiar with N4, you may find the "What's new" tutorial more useful as it is particularly designed to demonstrate the differences and new features of N6.

You are about to start work in a project in which you will explore immunisation beliefs, attitudes and behaviours of parents with young children. Data for this research was gathered in two stages, the first involving exploratory interviews, and the second using surveys to obtain open responses to structured questions as well as some scaled and other quantitative information. Because the two data sources cover the same general area of investigation, they are built into a single project.

[Stage 1 – Getting Started With Interviews](#) introduces the program, its displays and its tools, and covers the basics of importing and working with unstructured interview documents, including memoing and beginning coding.

[Stage 2 – Building Up](#) further documents are imported in a variety of ways, and a more sophisticated coding system is developed.

In [Stage 3 – Working With Survey Data](#) the structured survey data is imported as a batch. The open responses are autocoded, along with demographic and quantitative data, to allow for initial analyses. More detailed coding-on of responses follows.

[Stage 4 – Accessing Data To Answer Questions](#) introduces further tools for reflection and analysis, and shows how to make links back to quantitative data.

If you are familiar with importing and coding documents, and want to see only how N6's tools can be used to "process" survey data, then move directly to Stage 3.

# Stage 1 - Getting started with interviews

## *In this stage:*

[Exploring A New Project](#)

[Exploring Documents](#)

[Working With Text](#)

[Review And Report On Your Work](#)

[Moving On](#)

Several interviews have been conducted in families where there are young children. These have been transcribed, ready for analysis. A new project has been created, ready for you to begin work. In this stage you will import and work with just one document to gain an introduction to the basic processes of starting a project and working with qualitative software.

# Exploring A New Project

You will notice that a new project opens with three windows showing.

The Document Explorer is your window into the documents contained in your project database. When you import your prepared documents into the program, they will be listed in the Document Explorer, giving easy access to their text. The Explorer also provides rapid access to a number of options for working with a selected document: buttons for these are provided across the bottom of the window, or you can use your right mouse button to choose any of these actions from the right mouse menu.

The Node Explorer is your window into what you know and think about the project and the documents that are in it. Nodes store categories for thinking about your data. There are several types of nodes: free nodes are unrelated single categories; tree nodes are organised hierarchically to represent categories and subcategories; results of searches are stored under text and node searches. Each node holds references to text coded at it, so that text can be retrieved, reviewed, recoded, or searched. The Node Explorer provides options, symmetrical to those in the Document Explorer, for working with selected nodes.

The symmetrical structure of documents and nodes is also evident in the third window on your screen, the Project Pad. It also gives access to tools to create, explore and search data stored either as documents or nodes. Click on the Nodes tab to see the parallel options.

These three windows are set within a main project window. Across the top of the main window, a toolbar has buttons for managing both documents and nodes. Hover over any of the buttons on it to obtain a screen tip telling you what it will do. Across the bottom of the main window you will find a Status Bar, which will provide a fuller explanation of some of the options or buttons you select.

Press F1 from any window, and a Help file will open to outline the features of that window. Rather than giving long explanations of the components of every dialogue and window you will meet as you work through the tutorial, the tutorial will provide you with the language that will assist you to access further descriptive data from the Help files.

# Exploring Documents

Documents for Stage 1 of this project have been stored in the RAWFILES folder, provided for your convenience with any new project. There are six interviews with parents about a recent immunization experience, and one with a grandparent about his thoughts regarding immunization. Each of these interviews has been transcribed into a regular word processor file, then saved as a text only file (choose **Text Only** from the **Save as type** format slot when ‘Saving as’). A brief description (or ‘header’) has been typed as reference information into the top of each document (marked by commencing the first line with \*, followed by a blank line) and speaker names have been inserted into the file as sub-headers (also marked with \*), to break the text into sections based on speaking turns.

## *View a prepared document (and N6’s folders)*

- From the **File** Menu, select **Open > Text File**. The navigation dialogue will (most likely) open in your projects area. Click the up arrow to navigate to Samples\Tutorials\Immunization\Stage 1.

The file structure for your Immunization project is now showing: all projects created using N6 have a similar file structure. The Database holds the work you do in the project, with copies of your documents translated into N6 format, and your nodes (files and folders in this area should never be opened directly). Other folders (COMMANDS, RAWFILES, REPORTS, Temp) are provided for your convenience in managing your files and data. How each is used will become apparent as you work through the tutorial.

- From the Immunization\Stage 1 folder, select RAWFILES. Choose one of the text files, and click **Open**. When the file opens in N6’s text editor, you will see the structure of the document, with its description marked with \* at the top of the document, a blank line following, and sub-headers also marked with \*. The file has been saved as text only.
- Close the file without saving any changes.

## *Import a document*

In order to work with a document in N6 you need to import it. When you do so, the document is copied into N6’s Database. Depending on how the document has been set up in your word processor, and the Preferences selected within N6, a range of properties will be acquired by the document.

- From the Project Pad or the toolbar, click on the **Import Text Files** button . You will be taken to the RAWFILES area of your project folder, where available files will be listed. (If files for importing are not located in the RAWFILES folder, you can navigate to find them.)
- Choose to import “Vivien”. The document will be listed and selected in the left pane of the Document Explorer, with its properties (its title, description and number of text units) showing in the right pane.
- Click on the document description. Add the additional information that Vivien also has a daughter, Helen, aged 5 years. Click on **Accept Changes** to confirm the change.

## **Browse a document**

- With Vivien still selected in the Document Explorer, click on the **Browse** button. The text will open in a document browser. Note the description is separated from the rest of the text, at the top of the document. The description cannot be searched or coded.
- Click on a line of text that begins with \* (a sub-header). From your right mouse menu choose **Spread Selection**. Click on **To Section** and **OK**. Now everything said by that speaker in that speaking turn will be highlighted.
- Click on a line of text. You will find that the whole line is selected. Move to another line, and the same thing will happen. When you click on text, N6 will always select a whole text unit—the smallest unit of text that can be coded or retrieved. Text units are created by the program when a document is imported. They are marked by hard returns (end of paragraph marks). You can set Preferences to make your choice of lines, sentences or paragraphs the text units for your documents (**Project > Preferences > Text Unit Type...**). The default option (which you have just used) is to use lines as text units.
- Drag your mouse cursor through several lines of text. You can select as many or as few lines as you want, e.g. for coding at a particular node. Note that the status bar at the bottom of the main window shows the sequential numbers for the text units you have selected.
- Read right through the text from Vivien's document, in preparation for the next step.

# Working With Text

N6 offers several tools for recording your interpretation of document text. A memo can be used to record your overall impressions (or field notes that throw additional light on the text). The memo could also be used to record reflections on specific units of text. Annotations are seen within the text itself, and are used to draw attention to particular circumstances or features of the text.

Coding "tags" the text at nodes, so that it will become possible to quickly retrieve and review all the text about each of the categories in the project. Later, coding at nodes allows questions to be asked of the data (e.g. Does knowing someone who has had a vaccine preventable disease impact on attitudes to immunization?).

## ***Record a memo***

- With Vivien selected in the Document Explorer, click on the **Memo** button at the bottom of the Explorer, or select **Memo** from the right mouse menu. Click **Yes** to confirm that you do wish to create a memo for Vivien.
- Before you start writing, from the **Edit** menu choose to **Insert date & time**. This will assist in keeping an "audit trail" of the development of your ideas in this project.
- Type your thoughts about the main issues raised in Vivien's story about getting her son immunized. You might choose to keep the memo open in the background while you continue working with the text in Vivien's document, so that you can add reflections about particular points in the text to your overview of the main issues.

In the Document Explorer, an exclamation mark (!) in front of the document title indicates there is now an existing memo for the document.

## ***Insert an annotation***

In text unit 29 of the document, Vivien uses "I" and then "we" with reference to her son having to have a needle—an interesting slip of the tongue that is worthy of attention.

- Locate text unit 29 by pressing **L** on your keyboard (or choosing **Locate** from the **Browser** Menu at the top of your screen), typing in 29 and pressing **Enter** (or clicking on **OK**).
- Select **Add Annotation** from the right mouse menu, or press **A** on the keyboard.
- Type a comment about the way Vivien identifies with her son and projects her own fear into the situation, then close the text box, saving the changes.

The annotation will be inserted into the text, marked by the use of << >>. The added text unit will also be coded at the document annotations node, in case, for example, you might wish to exclude your annotations from future searches (or indeed, search only your annotations).

## ***Add to your memo***

Having noticed how Vivien uses "I" rather than "he" in one place, you might now see that there are several places where she makes similar statements, so that you now consider it worth recording a memo reflecting on these specific texts.

- Return to the memo for Vivien by clicking on the **Memo** button while Vivien is selected.
- Type in a heading for your reflection. Use the text unit numbers to identify the texts relevant to your comments, and/or use **Ctrl+C** (or right click or **Browser** and **Copy Selection**) to copy text units from the browser for pasting into the memo. Add your thoughts about the meaning of this particular discourse and its significance for understanding parental attitudes and behaviour regarding immunization.

You can also copy browser text to the clipboard and paste it into any project document or to an external application. However, note that the coding is *not* copied along with the text!

### ***Create free nodes to capture ideas in the text***

Free nodes are an especially useful way of capturing ideas at the beginning of a project when you have not yet developed structure in your coding system, as they do not presume any relationships between categories.

- Browse Vivien and highlight a passage of text that is of interest. Think about why it is interesting, or why that might be of significance, and find a word or phrase that will convey that.
- Type the word or phrase into the slot in the coding bar at the base of the Document Explorer, and press **Enter**. Through this simple action you have created a new free node, and coded the highlighted text at it.
- The new node will be selected in the Node Explorer, allowing you to add a description if you choose.
- Create further free nodes as you work through the document, or add further coding to the nodes already available.

### ***Create an in vivo node***

- Drag your mouse cursor through a word or phrase used in the text that would make a good node title (e.g. needle, or disease). The whole text unit will become highlighted, but the selected word or phrase will be shown "in-vivo" in the status bar at the base of the main window.
- Click on **In Vivo** in the coding bar of the Document Explorer. A free node will be created with the selected word/s as its title. The highlighted text unit will be automatically coded with the new free node.
- Extend the range of text units to be coded, and click **Code** (the new node title will already be showing in the coding bar).

### ***Code at an existing node***

There are several alternative ways to add further coding to existing nodes as you work through this (and other) documents. Choose the one with which you are most comfortable, or which best suits your needs at any particular time.

- Select the node you wish to code at from the Node Explorer. It will show in the coding bar on your document. Click **Code**.
- If the node has been recently used, select it from the drop-down list in the coding bar for the document, and click **Code**.

### ***Create nodes without coding***

Nodes can be created "up front" without immediately coding text at them, for those categories you know you are going to need to answer the questions you are asking.

- In the Node Explorer select **Free Nodes**. From the right mouse menu select **Make Free Node**,  
or click on the **Make Free Node** icon in the toolbar.

- In the right pane of the Node Explorer, type in a word or phrase as a name for the new node (e.g. experience of disease). If necessary (or helpful!) type in a description as well, to explain what the category is about. Click on **Accept Changes**, or **Tab** and **Enter** to confirm the changes.
- If, during this process, you have a document (or node) browser open and text selected for coding, click in the Title Bar of the browser (or **Ctrl+Tab**) to return to the text without changing the highlighting. The new node will be showing in the coding bar at the base of the document. Click **Code**.

# Review And Report On Your Work

At various stages as you are working with a document you will want to review what you have done, and finally you may wish to make a report of your document.

## ***Check your coding***

As you continue to add coding, you may wish to check what coding you already have on a selected passage of text.

- From the right mouse menu, choose to **Examine selection's coding** (or press **X** on your keyboard).

Any nodes coding any of the selected text will be shown in the Examine Coding dialog. You can obtain more detail by clicking on a node in the left pane, and you can choose to delete coding for any selected node.

## ***Review text coded at a node***

- Select one of your free nodes in the Node Explorer, then click on the **Browse** button to retrieve the text coded at that node.
- Highlight one of the retrieved passages. Press **X** on your keyboard to see what other nodes this text is coded at.
- To see the passage in its original context, choose **Jump to this Document** from the right mouse menu, or press **J** on your keyboard. The document browser will open on the highlighted passage of text, so that you can immediately view the surrounding text. If you choose to code additional text at the node while you are in the document browser, the display in the node browser will immediately update to show the added text.
- Your review may prompt reflection about the category. With the node selected, click on **Memo** at the base of the Node Explorer. As with documents, use the memo's text box to record your thinking about this category or concept (and don't forget to insert the date and time!). An exclamation mark (!) is added to the node title in the Explorer to show that it now has a memo.

## ***Report on the document***

Finally, you may wish to report on what you have done, so you can share your work with a colleague, or sit and review it away from the computer.

- Select the document in the Document Explorer and click on the **Report** button at the base of the Explorer.
- Experiment with the various options offered to see what each provides in the report.
- You can change the display for text unit numbering with document text by going to **Project > Preferences > Text Unit Numbering...** and indicating your preference in the resulting dialogue.

A report is a fixed-in-time picture of what is in your N6 database. A report can be saved and/or edited without changing the database. You can print it directly from N6, or if you save it (in the Reports folder is the best place), it can be opened in your word processor (you will need to open your word processor first, then choose the file to open, otherwise the file will open in Notepad).

# Moving on

Now you are ready to import more documents, and build a more elaborate coding system. [Stage 2](#) will outline these processes.

## *In the next stage:*

[Adding Further Documents](#)

[Creating And Using Tree Nodes](#)

[Coding Information About The Participants](#)

[Moving On](#)

## Stage 2 - Building Up

### *In this stage:*

[Adding Further Documents](#)

[Creating And Using Tree Nodes](#)

[Coding Information About The Participants](#)


[Moving On](#)

In working with the first document you have discovered many of the principles and processes by which N6 operates. In this stage you will see how to progress the project, and also gain a sense of the flexibility of N6's tools.

# Adding Further Documents

Document text can be imported in more than one way, and can be formatted in N6 with different kinds of text units. The formatting of descriptions and sub-headers (using \*) is always the same, however.


When you import documents, N6 automatically sets the text units according to your wishes. These can be set as lines (the smallest unit), sentences, or paragraphs.

- Choose which kind of text unit you would prefer to work with by making a selection from the options under **Project > Preferences > Text Unit Type**. Any further documents imported will reflect your choice. Lines are often best for interview data, but you may want to try working with sentences. You will find paragraphs are usually too long in interviews.
- You can import one or more documents at a time (via **Documents > Import Text Files as Documents**). Select "Angela", "Chris-Snr", "Felicity", "Janice", "Sandra" and "Stephen". (Use **Shift+click** or **Ctrl+click** to multiple select these documents.)
- If you are unhappy with the text unit type you chose for a document, select the document and click on the **Delete Document** icon in the toolbar , or choose **Delete '[Name]'** from your right mouse menu.
- Change the text unit type preference, and re-import.

Quickly read through the text for one or more of these documents. You might want to practice using some of the tools for annotating, memoing and coding to make a note of the concerns that different interviewees express about immunization or of experiences that may have influenced their attitudes.

## Appending text to documents

Lynn was later able to interview Pam, Stephen's daughter and mother of his grandchildren. This later interview was typed up separately, but would be better appended to the first rather than kept as a separate file.

- Select "Stephen" in the Document Explorer (the document you wish to append to).
- Click on the **Append Text File** icon in the toolbar , or from the top menu select **Documents > Append Text File to 'Stephen'**. Choose the file to append (Pam.txt) from the RAWFILES folder.
- Rename the document (in the right pane of the Document Explorer) to "Stephen and Pam" to reflect the change in its content.

Text can also be imported or appended from files that are not in text only format by bringing it in via the clipboard. Using the clipboard is an easy way also of importing selected text from a document, or information obtained from other sources, such as a web site. Just select what you want, copy, and then import or append.

You can also join two project documents together. The interview with Chris senior (Janice's father-in-law) could be appended to the document "Janice" rather than kept as a separate file.

- Select "Janice" in the Document Explorer.
- Click on the right mouse or **Documents Menu option Append document to 'Janice'**.
- Select the document ("Chris-Snr") that you wish to append to Janice's text and click **OK**.

If you browse "Janice" you will notice that the text of document "Chris-Snr" has been appended. Note that "Chris-Snr" is no longer a separate document.

### ***Splitting documents***

Of course, you may have a coded document containing two people's responses (e.g. "Janice" from the above example) that you wish to split into two documents.

- Browse document "Janice".
- Select the first text unit of Chris-Snr's text.
- Click on the right mouse or **Browser** Menu option **Split before Selection**.

The text beginning at the selection, together with all its coding, is removed from document "Janice" and becomes the text of a new document called "Janice B". A new browser will open on this document and you can rename it appropriately, e.g. "Chris".

### ***Adding an external document***

For this project you might wish to store references to the national guidelines for immunization providers, to published literature on immunization behaviour, or perhaps to photos of parents at immunization clinics, but these are things you can't type up and import. You can quite simply record these as external documents in the project, and this allows you to then record further information and coding.

- Select **Documents > Record External Document**.
- A new document named "External Document" appears in the Document Explorer with its name selected in the right pane, ready for editing. Type in a name and description for the document. For example, for Bazeley and Kemp's literature review you might type "Bazeley & Kemp 1994" as the document name, and then "Childhood immunisation: the role of parents and service providers. A review of the literature. Canberra: AGPS" as the description.
- Record the number of text units in the document. For a book, the most obvious choice is the number of pages so in this case the number of text units would be 55. This will allow coding references to be stored for particular pages. You might add "Pages are text units" to the description for the document, as a reminder.
- Click **Accept Changes** when you are done.
- Record another external document by selecting **Documents > Record External Document**.
- This time, name your document "Clinic Photos" and record its description as "Photos of mothers before, during and after having their baby immunised."
- For now you have only 10 photos, so record the number of text units as 10. Later, when you take more, you can modify the number of text units to include the additional photographs. Click **Accept Changes** when you are done.

You can record memos about your external documents in the normal way.

### ***Coding an external document***

You cannot browse an external document, and therefore you cannot code it in the normal way. Instead, the **Quick Coder** is used to record coding references.

- With the external document selected in the Document Explorer, select **Project > Quick Code...**  
*or* click on the Quick Coder icon in the toolbar . The document name will be showing already in the Coder dialog.

- Click on the node you wish to code part or all of your external document at (or type it into the node address slot). Enter the text unit numbers for the section to be coded at that node, and click **Code**. Thus, for example, you might code Photo 3 at both 'reactions' and 'fear'; and Photos 7 and 8 at 'needles' and 'fear'.

Your last action will show in the bar at the base of the Quick Coder.

# Creating And Using Tree Nodes

We can see already that these parents raise a number of concerns about immunization and have a variety of beliefs. They also express a range of feelings and talk of actions occurring in response to their concerns. So far these categories have been created and stored in the free nodes area of the Node Explorer. It will make sense to begin to organize them into "sorts" of categories, and to place them in trees.

As a starting point, we could create some trees to hold issues to do with vaccines, the diseases, the vaccination process, as well as feelings and actors. These will become "parent nodes" to more specific categories such as pain and suffering, anxiety, gives Panadol, or doctor. Later, you can ask questions about the interrelationship of these categories.

## *Create nodes without coding*

- In the Node Explorer, click on tree nodes and then from the right mouse menu, choose **Make top-level Tree Node**.
- A new tree node appears in the Explorer with the title selected ready for editing. Record a title for the new node (e.g. 'Issues re vaccines') and also a description if that would be helpful, then click **Accept Changes** (or **Tab** and press **Enter**).
- Repeat for 'Issues re diseases', 'Issues re process', 'Feelings' and 'Actors'.
- If the nodes appear to be out of order, check **Project > Preferences > Node Order**, and change from **By Title** to **By Address**.

## *Moving nodes*

- Needles are clearly an issue about the process for these mothers. Highlight 'needles' in the free nodes area of the Node Explorer. From the right mouse menu, select **Cut** (the insecure might prefer to **Copy!**). The node will be placed on the clipboard.
- Highlight 'Issues re process' under Tree Nodes. From the right mouse menu, select **Attach Node from Clipboard**. The clipboard node, 'needles', will appear under 'Issues re process'. Note that it retains its specific node address (unless that is in conflict with an existing address). Note also that a memo has been created for the node, to record the move. If the move is a significant one, open the memo and add a comment to it about the significance of shifting the node.
- **Browse** the node, and you will see that text coded there has followed with it when it was moved.
- Continue to move free nodes into appropriate trees. You may need to create a further level of sub-categories for some concepts.

If you are unsure where to place a node, leave it as a free node. You can move nodes at any time. You will also find, at times, that you will want to move a node from one tree to another, as your understanding of that concept grows.

## *Merging nodes*

If you start by making too many small categories, or you find that you have two versions of the same node, you may want to combine the coding at these for further analysis.

- Highlight the free node ‘do them good’, and **Cut** (or **Copy**).
- Select ‘commitment’ and then from the right mouse menu select **Merge Node from Clipboard**. You will be asked whether you want to append the memo for the clipboard node to the memo for the target node (unless you have been recording notes there, this is likely simply to record the cut you have just made). Choose **Yes** or **No**, and you will then find that the text references from the two nodes have now been combined.

**Warning!** It is difficult to "undo" a Merge, even if you copied rather than cut the source node, as some text may have been coded at both.

## ***Coding with tree nodes***

This would be a good time to browse another document. You will be able to continue to use the nodes you already have (both free and tree), and develop new nodes as you work.

- Select a document in the Document Explorer, and click on **Browse**. It will open in a document browser. Quickly read through, to remind yourself of what this document was about. You may wish to open the memo for the document at the same time, so that you can add comments there, as you work through coding the text.
- Highlight a passage of interest, and decide how you are going to code it. If the node exists in your Node Explorer, press N (or browser menu option **Select Node for Coding...**) and select the node.
- Press Ctrl+Tab to return to the browser and note that the node has been copied to the coding bar. Simply click **Code** on the coding bar to code the selected text.
- Recently used nodes can be most easily accessed from the drop-down list in the coding bar for the document.

You will have noted by now that nodes have “addresses” as well as titles: these are shown in both left and right panes of the Node Explorer. They provide a shorthand way of telling which node you want, and can be used as an alternative to clicking on the node for coding. You will need to have a list of your nodes (with their addresses) beside the computer to make best use of this way of coding (see below for how to generate a list).

- Highlight a passage of text to be coded.
- Type the node address (with spaces between numbers—brackets are not necessary) in the coding bar slot. For example, type node address 3 1 to identify the node ‘needles’ under ‘Issues re process’.
- Press **Enter** or click **Code**.
- To create a new tree node, type an address for the new node in the coding bar slot, click **Code** (or press **Enter**). N6 will create it with the name slot highlighted in the right pane of the Explorer, ready for you to rename it.

## ***Displaying your trees***

Your nodes can be displayed as a list (all nodes), or as a picture (tree nodes only).

- To obtain a list of all tree nodes, select **Nodes > Nodes Report**. Choose category **Tree Nodes**, click **OK**, and a list of tree nodes will be provided. This list can be printed directly from N6.
- If you want a report on just one tree, select **Sub-Tree beginning at:** and select address. You could also make a report for **All Nodes**, **Free**, **Text** or **Node** searches.

There are a number of options available for additional information: you might choose to include descriptions for each node in your list, or to see the frequency (in terms of number of documents) with which each node has been used, for example.

- To see a “live” diagram of your nodes, select **Nodes > Node Tree Display**. Move through the layers of the display by clicking on the relevant arrow heads. A click on any node in the tree will bring up the context menu for that node, so that you can choose to work with that node from the tree display rather than the Node Explorer.
- Nodes can be exported to the graphical programs: Decision Explorer and Inspiration. These allow manipulation of the display for presentation purposes (both programs) or for further analysis (DE). Demonstration versions of these programs are available from <http://caqdas.soc.surrey.ac.uk/>.
- If nodes are exported to Inspiration without memos, the resulting file can be opened in a word processor to give an alternative layout for a tree node report (you do not need to have the Inspiration program for this to work). From the main menu, select **Nodes > Export Tree Nodes**, then in the resulting dialogue select Inspiration as the desired format and other options as required (specific node titles and no memos gives the best output).

# Coding Information About The Participants

In this project, like most others, we know something about the participants as well as what they say. For example, we know how many children they have, and which was the last immunisation that the youngest child had. Each was also asked about their level of education and whether they were working or not. It will be useful also to code these documents as interviews, to distinguish them from the surveys which are yet to be imported. Such information, typically referred to as base data in N6 projects, may be relevant to everything said by a participant, and needs to be included in the coding of their text. This can be achieved in three different ways.

1. Nodes can be created for the demographic categories, and used in coding the text. Whole documents can be selected for coding using **Ctrl+A** (Select All). Part documents can be selected in the usual manner, and the coding added.
2. Nodes can be created, and then the Quick Coder can be used to code either whole or part documents. This is generally more time-efficient than the first method (especially for whole documents).
3. The information can be entered into a spreadsheet or other table based software and imported into the project. Data is matched using document names. In this method, the necessary nodes are created from the variables and values in the table as it is being imported, and the documents are coded at the same time. Whole documents only can be coded in this way, but for these, this is the most efficient method of undertaking this kind of routine coding.

## ***Importing a base data table***

A table has been prepared for documents in this tutorial. It can be seen by using **File > Open** to display the file docdata.txt, which is located in the RAWFILES\Data folder. Note that this information was originally typed into a spreadsheet, then saved as tab-delimited text to prepare it for import to N6. The word "document" (no quotes) must appear at the head of the first column for N6 to recognize this as document information. Do *not* try to "tidy up" this file, as you may move values under the wrong variables by changing the tabs.

## ***To import the document base data table:***

- First, select **Tree Nodes** and right click to **Make top-level Tree Node**, then name it 'Base Data' (this will be the base data parent node). Change its node address to 10 (typically base data is placed at Node 1, but we already have other coding there.)
- From the main menus, select **Project > Import Table**. Select docdata.txt from the project's RAWFILES\Data folder.
- Type in the address for your base data node (i.e. 10). Confirm that you wish to go ahead with the import.
- N6 will create the nodes and code the documents. A coding report will be made at the end of the import, advising you of documents coded and any (listed in the table) for which coding was skipped.

Note that educational information for the document Stephen and Pam could not be imported in this way. Stephen's contribution will need to be coded separately from Pam's for education (Stephen had some secondary education only and Pam has tertiary education). This can be done using either of the first two methods of coding mentioned above.

A parallel table import process allows for base data coding of part documents or collections of documents gathered in nodes to represent cases—refer to the manual if this may be relevant to your project.

# Moving On

In [Stage 3](#) of this tutorial you will meet data about parental responses to immunization derived from a self-report survey.

If you are unlikely to use survey or structured data in your project you might skip Stage 3 and go straight to [Stage 4](#) where you will learn about further tools for searching, reflection and analysis.

## ***In the next stage:***

[Preparing And Importing Survey Documents](#)

[Importing Quantitative Data](#)

[Searching Text To Code Responses To Questions](#)

[Searching For Patterns In The Data](#)

[Further Coding](#)

## Stage 3 - Working With Survey Data

### *In this stage:*

[Preparing And Importing Survey Documents](#)

[Importing Quantitative Data](#)

[Searching Text To Code Responses To Questions](#)

[Searching For Patterns In The Data](#)

[Further Coding](#)

In this stage of the project, you will import data gathered using structured self-report surveys. Typical of most surveys, the data come in a variety of forms including text responses to open-ended questions, scale and other numeric data, and categorised responses to fixed choice questions.

Categorical information can be imported as for base data in the Stage 2 project, i.e. by setting it out in a spreadsheet or statistical package, and using the table import process in N6 to create the nodes and code the documents. And because the text responses are structured, with judicious use of subheadings, text search can be used to locate and code all the answers to each of the questions. This autocoding process can be speeded up even further through the use of a command file which uses a simple language to tell N6 to carry out repetitive functions.

With responses coded for the question that was asked (and thus, usually, for the main issue they deal with), and documents coded for demographic and other categorical information, it is possible to immediately (and very efficiently) obtain a breakdown of text responses by different categories of respondents, so for example, finding if experiences vary depending on the level of education of the respondent. This may be all that is required for some simple types of analyses. Further coding of the detail of responses could then allow for more sophisticated analyses of the data.

N6 allows a choice of ways of working with detail in the data. You can work through the text responses for each document, if it is important to keep a focus on the person who is responding. But if, as is often the case, your focus is much more on the issues being discussed, you may prefer to go through all the text for a particular question. That way, you can keep your focus on that issue, become familiar with the nodes the data are being coded-on into, reflect on the issue and then move straight into analysis of data around that issue, while it is fresh in your mind, before coding another. This is possible because the node browser in N6 provides live access to data and tools for working with it, rather than simply a report of what is stored at the node.

It is important when working with survey data to keep each respondent's data in separate documents, even if it makes for a lot of short documents. This is to allow both for table import of quantitative information and for document counts in results as a way of representing numbers of respondents. The latter is important if the data is also to be analyzed statistically in any way (and most users of surveys want frequencies and some cross tabulations, at least). Note that N6 can provide both text and statistical output for use in analysis and reporting.

# Preparing And Importing Survey Documents

Survey documents are highly structured documents. Use that structure to your advantage when documents are imported into N6. (The survey form used for this tutorial is located in the RAWFILES folder for Stage 3.)

## ***Preparation of documents***

A document template (located in the same folder as the survey form) was created for typing the responses to the survey, and a shortcut to the template placed in a convenient location. The template contained question identifiers, correctly formatted to be read as subheadings in N6. When you use a template, each time you double-click on the template shortcut, it opens as a new document showing text (e.g. sub-headers) already entered. Apart from being faster for the typist, using a template means that sub-headers will be absolutely standardised so they can be used with text search to autocode responses. If there is no response to a question, the typist can simply delete the sub-header for that question. As each document is typed, it is saved in text only format, ready for importing.

## ***Preparation for document import***

Before importing documents, you must tell N6 how you want the text units in those documents to be formatted (i.e. as lines, sentences or paragraphs). Usually for survey data, sentences or paragraphs make the best text units (paragraphs if the answers are all short, sentences if some are longer).

- From **Project > Preferences > Text Unit Type**, choose **Paragraphs**.

## ***Importing using a command file***

Earlier you learned how to import a number of documents together, simply by selecting all the ones you wanted (using **Shift+click** or **Ctrl+click**) at the one time. Alternatively, a command file is a way of telling N6 to do a task without your needing to click and point to select options through the menus or dialog. Command files are particularly useful for repetitive tasks. A simple, one-line command file (introduce-documents all) will import all the documents in your RAWFILES folder in one go, regardless of how many there are. Optional lines can be added to the command to tell N6 to also code the documents at a particular node as they are being imported.

- Select **File > New > Text File** (there is another option there **New > Command File** – we’ll encounter this later) and into the empty text document type the lines:

```
(introduce-documents all
node (10 5 2)
node-title “surveys”)
```

Make sure you include the hyphens, the quotation marks and all the brackets.

- Save the file as Introsurveys, in the COMMANDS folder for the Stage 3 project.
- To run the command file, go to **Project > Run Command File**, and choose to run the file Introsurveys. Then sit back and watch as the documents are each imported and named in the project!
- Browse one or two of the survey documents, to see how they look in N6.

Note that the interview documents for this project had been removed from the RAWFILES folder to another location prior to running this command file, so that they were not reimported along with the survey documents.

# Importing Quantitative Data

N6's table import facility can be used to include not just demographic data about the survey respondents, but also pre-categorised responses and scaled scores. Any or all of these variables may be relevant to an understanding of parents' experience of and attitudes to childhood immunisation.

## *Preparation of data*

Demographic, scaled and fixed response data, which included sub-scale scores and continuously scored variables, were originally set up in a statistical database for computation of totals and to allow for statistical analysis. After totals were computed and categorised, the relevant columns of data (with value labels showing) were copied to an Excel spreadsheet to be saved for use in N6. (An alternative approach would have been to enter the data directly into Excel or other table-based software.) The first column was renamed "document" (rather than ID), and formatted to text (**Format > Cells > text**) so that Excel did not drop the leading zeros for the document names. The table was then saved as a tab-delimited text file from the Excel spreadsheet, to prepare for import into the N6 project. (The spreadsheet and its text version are located in Immunisation\Stage 3\RAWFILES\Data.)

The "quantitative" variables for this study are:

Last imm: age for last immunisation

Stress: how stressful is it for the parent (scale)

Assist: did someone else go too, to help?

Do check: do mothers "just do it", or check first?

Extfam: Do they have extended family living nearby?

Chn<5

Education

Employment

Hinternal: control own health

Hothers: health controlled by powerful others

Hchance: health a matter of chance

## *Importing the data*

- From the main menus, select **Project > Import Table**. Select surveydata.txt from the DATA folder located inside the RAWFILES folder.
- Alter the address for the base data node to (10) and confirm that you wish to go ahead with the import.

- N6 will create the new nodes as needed and code the documents. A coding report will be made at the end of the import, advising you of documents coded and of those (listed in the table) for which coding was skipped because a match could not be found in the documents list.

Now check the frequencies for each value of each variable for the survey data only:

- Select **Nodes > Nodes Report** and enter the base data node address 10 in the **Sub-Tree beginning at:** field and check **Number of Documents Coded**.
- Click on the **Restrict** button and specify **Only** those **Documents** that are **Coded At** the node with address 10 5 2, the node 'surveys'.

# Searching Text To Code Responses To Questions

A text search can find any string of characters in the text of imported documents. Shortly we will use text search to explore the use of particular words through the text, and to code those uses. But the ability to search text can also be used for mechanical, clerical purposes such as to gather all the responses to a question into a node for that question. Autocoding in this way can facilitate further analysis, and the use of command files to do this kind of coding frees the researcher to spend more time reflecting on the content of the data.

When a search is run, the finds from the search are automatically saved at a node. You can then spread the coding to include surrounding text units, such as all those in the same section, for all finds at that node.

## *Searching text interactively*

- In the Node Explorer, select the node (10 5 2) 'surveys'. Click the **Text Search** button at the base of the Explorer. The resulting search dialogue immediately indicates you will be searching only text coded by (10 5 2) ~/surveys.
- In text search dialog, type \*Q1 as the **Text search specification**. Ensure that **Search for whole word or phrase only** is checked (for example, to avoid confusion with \*Q10). In each document, select to **Keep first find only** (for surveys there is no need to continue searching once the relevant question has been found.)
- Now, **Do the Search**. A report will be provided, indicating that there was one find in each document (a summary of finds would be sufficient for this type of search).

Results of the search have been placed at a node in the text searches area of the Node Explorer. This node can be renamed in the usual way (e.g. to 'Q1 Baby's health'), and is already provided with a description indicating how it was created.

- Browse the new node. You will find that just the sub-headers for Q1 have been retrieved.
- Select the node in the Node Explorer, and from your right mouse menu, choose to **Spread all Coding** then click **Section** and **OK**. Note that the contents of the node browser are immediately updated, and now show all the responses to Q1, as well as the sub-headers.
- Read quickly through the responses at 'Q1 Baby's health'. Having all responses to a question together allows the researcher to gain a different view of the data—in this case to gain a rapid overview of the range of responses given to the question.
- Close the node browser and return to the Node Explorer. Select **Tree Nodes** and right click to **Make top level Tree Node**, naming it 'Questions' and giving it address (11).
- Finally, select and right click to **Cut** the 'Q1 Baby's health' node from under **Text Searches**, then select the new 'Questions' node and **Attach Node from Clipboard**.

You could repeat these steps for each question, or use a command file...

## *Using command files to search and code*

Although text search efficiently gathers all the answers to a question together, doing it interactively (searching, spreading, shifting the results node into the tree) for a number of questions becomes a slow and tedious task. Better to write a command file, which will do it in one step, and which can be re-run if needed (e.g. if more surveys are added later).

The format for a command file of this type follows the components of the text search dialogue. To tell N6 to search only the survey documents, a **Restriction** is placed around the commands that search and code each of the questions.

- Open the Command Assistant Window by clicking the on the toolbar button *or* by selecting **File > New > Command File**. The Command Assistant Window opens showing Command Assistants on the left side of the window.
- From the **Main** tab, select the **Set Search Restriction** command.
- Type **10 5 2** in the **Specify a node address** field to restrict only documents with coding at (10 5 2) 'surveys'.
- Click **Add to Command Assistant** and the restriction command is added to the command edit window.

Note that the cursor (insertion point) is in between the set-restriction and end-restriction command lines. Any search commands inserted at this point will be subject to the restriction. (Tip: never place a restriction inside a restriction – it will not work!)

Now to add the text search commands.

- From the **Reports & Tasks** tab, select the **Code Cases** task.

The Code Cases task dialog asks you to provide the unique search strings (question identifiers) that identify each case, the address of the parent node of the nodes that will code each question and a name for that node.

- In the first field, type each of the unique search strings in turn (without quotes) and **Add to List** or paste (using **Ctrl+V**) a list if you have one – the strings in the list must be identified using quotes like this although it doesn't matter if they are on separate lines or all run together on one line:

"Q1"

"Q2"

"Q3" "Q4" "Q5" "Q6" "Q7" "Q8" "Q9" "Q10" "Q11" "Q12" "Q13" "Q14"

- Each of the survey questions are a section so next to the list of search strings, ensure that the context to be coded is **Section** so this is also coded as context.
- In the two fields at the bottom of the window, type the address of the parent node e.g. **(11)** or **11** for the nodes coding each case.
- Click the **Add to Command Assistant** button and the command to autocode cases is added to the Command Assistant Window's text:

The first line of the inserted text is a comment describing the command. Sub-comments like the second line begin with a single semi-colon and describe the parts of the Code Cases Task. Note that the list of search strings and other information from the Code Cases task dialog has been added to this series of commands in the right format so the command can be run correctly by N6.

This command will tell N6 to search all documents looking for Q1 (and whole-word ensures it will ignore Q10) Q2, Q3 and so on. Once it has found the specified text in a document, it can move to the next document. Results are to be spread to include the responses (Sections) and saved at a new or existing node under (11) and the name 'Q1', 'Q2' etc., i.e all the steps you carried out sequentially in the previous exercise above are done in one step.

Before running the command, expand the (11) 'Questions' sub-tree in the Node Explorer and note that at present there is only one case node (11 1) 'Q1 Baby's health' that was coded in the previous exercise above.

- Click **Run Command File** to run all commands in the window.

In the Node Explorer, you can see that new nodes have been created under (11) 'Questions'. One advantage of using a command file to code for questions is that you can run it again later should you import further survey documents.

- Save this command file for future use by choosing **File > Save As "Code Cases.txt"**


And with that, you can move straight to preliminary analysis of the survey responses!

# Searching For Patterns In The Data

N6 provides a set of search tools, the node searches, which will find the text you need for whatever question you want to ask of the data—providing you have nodes for each of the concepts in the question, or can generate them. For now you will ask quite straightforward questions, using just the nodes from the survey that have been created through autocoding.

## ***Compare groups with a Vector search***

Already you can explore to see if responses to the open ended questions in the survey differ for different sub-groups of the survey sample, or for those who score differently on scaled measures. Does level of education make a difference to the theme of their main thoughts about immunization, for example? Asking this sort of question requires that you do a Crosstab (Vector) Node search.

- Click on the **Search & Compare Nodes** icon  in the toolbar *or* under the **Nodes** tab on the Project Pad.
- Click on the **Others** tab and click **Vector** in the **Crosstabs** area *or* select **Nodes > Search/Compare Nodes > Vector**). The Vector dialog will open.
- Type **11 3** (the node address for ‘Q4 Main thoughts re imm’) into the first slot of the Vector dialog. Select **Intersect** from the drop-down list in the centre slot, and then type in the *parent node* address for ‘Education’ **10 3** in the third slot. (Intersect finds text which is coded at both this node *and* that node.)
- Click **Restrict** to limit the scope of the search to **Only** those **Documents** that are **Coded At** the ‘survey’ node **10 5 2**.
- N6 will make a new node of the results, placing it in the node searches area of the Node Explorer (rename it if you wish) and opening it in the Matrix Viewer.

As a vector, this table has only one row. The top of the Viewer Window records the title and description of the vector node as well as the parent nodes of the row and columns in the table.

The row and column headers in the table record the row and column nodes themselves – hover over them to see their full title and address in the status bar.

- Click on the row header for ‘Q4 Main thoughts re imm’ to browse that node.
- Click on the column header for ‘some secondary’ to browse that node.

The cells in the body of the table show which intersections contain coding.

- Click on each of the cells to see what text is common to ‘Q4 Main thoughts re imm’ and each of the education nodes. The text is reported in a text window that you can save if you wish.
- To generate a report of all the cells in the table, click on **Make Report...** and then press **OK**.

The report from the Vector search will show you the text responses from Q4 sorted by the different categories of education. From this we can see a pattern: those who did not complete high school all made reference to the baby being upset, crying or experiencing pain as their main thought about immunization (Cell 1 2), while those with more education (completed high school or university training) referred to health issues as their main thought about immunization (Cells 1 3 and 1 1 respectively). (You might want to reorder some of the sub-nodes in your base data tree, by changing their specific addresses in the right pane of the Explorer, before doing these kinds of searches!)

## Further Coding

Often survey responses can be quite complex, and warrant more than simple sorting according to what question was asked. Coding the detail of responses can be done in the same ways as for interview data, i.e. nodes are created to capture the ideas present in the data. As with interview data, these may be created as free nodes, or located in node trees under **Tree Nodes**. Often researchers are tempted to create a set of nodes under a particular question to tap the responses to just that question, and then create another set under the next question, and so on. This should be avoided, as it will lead to a coding system that becomes unmanageable in two ways:

- a. repetitions of the same categories under different questions will make for an unnecessarily large and unwieldy number of nodes, and
- b. it becomes difficult to access all your data about a particular theme or issue or response because the text will be dispersed under several different parent nodes.

If you code everything on a particular topic to the one node (sorted, as for the interviews, according to what kind of thing it is about), it is easy to ask which question prompted those responses using a Vector Intersect search (theme by questions). More importantly, it allows you to examine that particular concept as a whole, its relationships with other nodes, and what it means for the project as a whole.

### ***Coding from the node browser***

As suggested earlier, interactive coding can be done using the node browser rather than the document browser, and for surveys this is often a useful procedure.

- From the Node Explorer, select (11 2) 'Q3 Recent experience', and choose to **Browse** the node.
- Work through (some of) the text in the browser, using and creating free nodes and tree nodes as needed. You will find that many of the issues raised here came up in the interviews as well, so you can use the nodes created earlier. If, later, you want to separate the interview data from the survey data to compare them or to view just one source, this can be easily done.

# Moving On

In [Stage 4](#) you will find that much of the data from both the interviews and the surveys has been coded, and some memos written. You will learn more about text search, and you will be able to explore further tools provided by N6 for reflecting on and refining and testing your ideas about the data.

## ***In the next stage:***

[Reviewing Nodes](#)

[Searching](#)

[Accessing Data With Text Search](#)

[Asking Questions With Node Searches](#)

[Matrix Searches](#)

# Stage 4 - Accessing Data To Answer Questions

*In this stage:*

[Reviewing Nodes](#)

[Searching](#)

[Accessing Data With Text Search](#)

[Asking Questions With Node Searches](#)

[Matrix Searches](#)

Having worked through the data in your project, coding and writing memos, you will already have a number of ideas about the data which warrant exploration and testing. As an initial step you might simply review some key nodes, but then you will need to access data in special ways using the search tools provided by the program.

## Reviewing Nodes

The way in which parents weigh up the risks associated with childhood infectious diseases and immunization is of particular interest to public health professionals (and parents!). Four of the six mothers who were interviewed gave some indication of their thinking about these issues. Reviewing the text at core nodes such as this is a good way to progress your analysis beyond coding.

- In the Node Explorer, locate and **Browse** the node ‘weighing up’ (1 3 3). As you read through, make a note of the primary issues raised in a memo for the node.
- Select each segment and right click to **Examine Selection’s Coding** (X on your keyboard) to see what else it is coded at (including the demographics of the speakers). You might note, for example, that each of those who talk about a weighing-up process has either full secondary or tertiary education (none with some secondary only), and each (other than Janice, the non-immuniser) has a very young child—when perhaps the issues are more pressing.
- You will find that the segment from some of the documents, e.g. Sandra, is rather too short to be really helpful. In any case, it will be a useful step to review the context of each of the passages. Make a selection, then right click to **Jump to this Document** for each passage (J on your keyboard will take you there) to see what else these mothers talked about.

Each of them talks about experience of the diseases in one form or another (3 for self or close relatives, 1 via seeing it on TV), with this experience being negative for 3 and neutral for 1. Note that the conclusions they reach about immunization fit with their experiences of the diseases (3 pro-immunization, 1 anti). Your review then, could suggest (ignoring the issue of sample size for the moment!) that there may be a link between exposure to the (varying) results of childhood infectious disease and the development of attitudes to immunization.

# Searching

As well as facilitating review of nodes, N6 provides tools to access and explore text and the relationships between categories developed from the text—tools which allow the researcher to ask questions in ways that would be difficult, if not impossible, without software. These tools are of two types, text searches and node searches.

Text searches can be used to find all instances when a word or phrase is used. They might be used as an adjunct to coding; in an exploratory sense, to see if and how a word or phrase is used in the data; or to undertake a word study, e.g. of a metaphor, or perhaps the use of a particular pronoun within a specified context.

Node searches (in 17 different varieties) are used to examine patterns in occurrences of nodes through the documents in order to reveal relationships, patterns, links, nuances and new dimensions in the text.

What you will find, as you begin to explore data, is that you will start to use these tools interchangeably, taking the results of one search as input to further searches (referred to as "system closure"), in the quest for understanding what is happening.

# Accessing Data With Text Search

It was noted in Stage 3 that text search can be used to locate any string of characters in the text, and to store finds from the search at a node. This ability to rapidly access and retrieve data on particular topics is no substitute for regular coding (it depends, for example, on there being an appropriate string to search for), but it can be useful as an exploratory tool, or perhaps for checking the completeness of coding.

## ***Searching and coding***

Text search can find where mothers talk of giving Panadol to their infants in preparation for vaccination, for example. The node referencing these passages could then be attached below (3 1) 'preparation', to indicate a particular kind of preparation.

- To search all documents, click on the **Text Search** icon in the toolbar *or* under the **Documents** tab on the Project Pad.
- In the **Find what** slot of the Text Search dialog, type the word 'Panadol' then click the **Do the Search** button. The text search results node appears in the Node Explorer and a text report is generated showing the finds in uppercase.

Review the report to see what was found by the search. You might note that one of the surveyed mothers said she forgot to give Panadol. Another indicated that she gave Panadol after the injection. Some of the finds in the interviews are unclear because just the line (text unit) in which the word appeared has been located.

- Browse the text search results node. Note that the text search node is already selected in the coding bar slot of the node browser. Where a find is not about using Panadol as preparation for vaccination, highlight the text unit, and click **Uncode** in the coding bar.
- For those finds that are unclear, highlight the text unit, and choose from the right mouse menu to **Spread Selection's Coding...** selecting **Paragraph** (for example). You may then choose to leave the additional text at the text search node, or to remove it by then selecting **Undo last spread** from the right mouse menu.
- Alternatively, highlight the text unit that is unclear, and press **J** to jump to the source document. Again, the text search node will be selected in the coding bar, so the original text can be uncoded, or additional text selected and coded. Close the document and return to the text search node.
- Once you are satisfied with the text you have at the node, close the browser and in the Node Explorer rename the node to 'gives Panadol'. Cut 'gives Panadol' and attach it below (3 1) 'preparation'.

It is always necessary to check the finds from a text search. Use the node browser to refine coding done using text search, even where the search term is as clear as the word Panadol!

## ***Checking coding***

A number of respondents to the survey talked about vaccines in terms of giving protection from disease. You might want to check if any of the interviewees also spoke of it in this way, but that you had missed noting it. The node for text already coded as being about giving protection is (1 2 5). You can refine your text search to find only those references to protection you don't already know about, by restricting what is searched.

- Select the node (1 2 5) 'give protection'.
- From the right mouse menu, choose **Text Search > Passages not Coded by Selected Node**.

- In the **Find what** field, type "protect" (this will find "it protects her from" as well as "gives protection", for example). The restriction on the search is already noted at the top of the dialogue. **Do the Search.**
- There are no finds from this search, so it can be concluded that there are no further instances of people talking about vaccines giving protection. Close the report.

If there were finds that you wanted to keep, these could then be checked and coded at the existing node, or the text search node merged with the existing node.

### ***Exploring with text search***

Often the idea you want to explore in the data cannot be summed up in one word (e.g. worried, anxious), or there are alternative spellings for the word (e.g. immunise and immunize). Use Special Characters to allow for variations in the word/s that are searched. These may be typed in, or selected from the popup menu below (see the User Guide or access the Online Help for details). Make sure you carefully check through the results from these searches before using them in further searches.

# Asking Questions With Node Searches

Searches of various kinds used singly or in combination will give you complete access to the data you need to answer questions you might ask. Some examples follow, for you to work through, but what is possible is limited only by your level of inquisitiveness (and time!).

## **Comparing responses: Intersect, Less and Coding Stripes**

You might wish to compare the responses of mothers who express anxiety about reactions to vaccination with those who don't, and to see what might help explain those different responses. Your first search, then, will look for text where mothers express anxiety about reactions.

- Click on the **Search and Compare Nodes** icon in the toolbar. N6 provides pictorial assistance to help choose which of the 17 different search operators is appropriate.
- From the **Boolean** tab click on **Intersection**, a search that will find text that is coded at both 'reactions' (1 1) and 'fear-anxiety-worry' (4 5), to answer the first part of our question.
- Type the two relevant node addresses into the Intersection dialog. For Intersection, the order of the nodes does not matter e.g. (1 1)(4 5) or (4 5)(1 1), or type them with a comma in between, thus: 1 1, 4 5
- N6 will search the documents to find any text coded at both these nodes, and will place the found text at a node search node (N 2)—rename the node if you wish. Browse and assess the text that has been found.

Now, to find what is said by those who don't express anxiety in relation to reactions, you can use a Difference (Less) search.

- Click back to the **Search & Compare Nodes** dialog's **Boolean** tab and click on **Less**. Notice that the nodes are entered as for the previous search.
- In this case it will be important to have the nodes typed in the correct order. The search will find all the text at the first node, less any that is also coded at subsequent node(s). If you typed (4 5)(1 1) for the intersection, delete the entry and type (1 1)(4 5) to find 'reactions' less 'fear-anxiety-worry'.
- Again, the found text will be placed at a node search node (N 3). Browse and compare with what was found earlier. Can you detect differences in the way these mothers talk, compared with those who were anxious (apart, of course, from the anxiety itself!)?

The results of these searches can then be further examined to see if there are differences between these groups which might explain the different patterns in the way they talk about reactions. You might have a "hunch" that the age of their children might have something to do with the mothers' level of anxiety about reactions. One way to check this would be to look at the text with coding stripes for age categories showing.


- Select the node search node coding the intersection of 'fear-anxiety-worry' (4 5) and 'reactions' (1 1)—this should be node (N 2) but check the node's description to make sure.
- Click on **Report...** at the base of the Node Explorer.
- In the **Make Node Report** dialog, check against **with Coding Stripes...** (leaving **Coded text** and **with Sub-headers** already checked). Click **OK**.
- To select age stripes for display, expand the **Tree Nodes > Base Data > Last imm** nodes so that the ages when children were last immunised are showing. Click on '2 mths', and then **Add**, then '4 mths', **Add**, and so on (in age order rather than node order) through to 'pre-school' and **Add**, so that the six age groups are showing in order in the right pane of the dialog (ignore None). Click **OK**.

- Expand the report window as needed to see the text from the first node search with letter codes in the right margin, indicating the age groups for the children of the mothers who expressed anxiety about reactions. Note that their children are primarily in the younger age groups.
- Now make the same report for the second node search node (N 3), using the same coding stripes (these will already be selected). This time you will notice that more of these mothers (though not all) have children who are from the older age groups.

As with all searches, you need to assess the text, consider exceptions to observed patterns and determine if there are conclusions which can be safely drawn (and argued) from the evidence before you.

### ***Testing relationships: Proximity (Near)***

One might wonder whether those who gave Panadol in preparation for vaccination were particularly anxious about the process and/or their child's reactions to it. Coding for each of these concepts ('gives Panadol' and 'fear-anxiety-worry') is unlikely to be on exactly the same text units, so it will not be found by Intersection. This time, you will need to use a Proximity search.

- Click back to the **Search & Compare Nodes dialog** (opened from the toolbar button ) and from the **Proximity** tab choose **Near** as the search you want to do.
- In the first slot of the Node Search dialog, type the address for 'gives Panadol' (this should be 3 1 1). In the second slot, type the node for 'fear-anxiety-worry' (4 5).
- You now need to specify how near you expect or want the text for the two different nodes to be (the **Context**). In this case, in the same **Document** would be appropriate. Retrieve the text for **Both** (selected by default). Click **OK**.
- The results node (N 4) of this search can be browsed in a new node appearing in the Node Search area of your Node Explorer (name it if you wish). Note that the definition for the node describes how it was created.

Your conclusion? Three of the four mothers who gave Panadol as preparation for vaccination also expressed anxiety about either the needles, or reactions, or both. For these mothers, the giving of Panadol was clearly linked with (an outcome of) their anxiety.

To find what mothers who were anxious about needles or reactions did (if they did not use Panadol) requires two preliminary steps:

- Click back to the **Search & Compare Nodes dialog's Boolean** tab and make a **Union** of the nodes for 'reactions' (1 1) and 'needles, pain' (3 4). Name the new node if you wish and note its address e.g. (N 5)
- Do an **Intersection** of the results of that search e.g. (N 5) with anxiety (4 5). Again, name the new node if you wish but also note its address e.g. (N 6)

This node now codes all mothers' anxiety about reactions and/or needles. Finally, to identify how the mothers *who did not give Panadol as preparation* express anxiety about reactions and/or needles:

- Click to the **Others** tab of the **Search & Compare Nodes dialog** and choose **Excluding-docs-from**. Type the node address for the results of the last search e.g. (N 6) into the first slot of the dialog, and the node for 'gives Panadol' (this should be 3 1 1) into the second slot.
- Review the text at the resulting node and compare with that found for those mothers who did give Panadol as preparation.

These mothers also expressed anxiety about needles and reactions, so the giving of Panadol is not specifically linked to that kind of anxiety, after all. Do any other factors appear to differentiate between these groups, or is it quite random (as far as can be determined from the information available) as to who gives it and who doesn't?

Now that you know how to search for nodes which are near each other, you could use the same technique to check earlier tentative conclusions from your review of context around weighing up risks (that this is linked to, or influenced by, experience of the diseases).

## Matrix Searches

Matrix searches allow for exploration of patterns of relationships across nodes by creating a qualitative "cross tabulation" of data where nodes belonging to one subtree are each paired with nodes from another subtree. Earlier (in Stage 3) we used a similar search (Vector) to "split" the text coded at a single node by the subcategories of another.

Our node review above suggested that mothers who questioned immunization (or at least, who weighed up the alternatives) may have a higher level of education than those who don't. We could cross-tabulate the various levels of education with issues raised about immunization to see if there are patterns suggesting an association between a mother's education and her approach to thinking about immunization, particularly as to whether she takes a "trusting" approach, or a "questioning" approach.

### *Using Collect in preparation for further searching*

For this exploration, it would be useful to first gather together the subnodes under each of the 'trusting' and 'questioning' nodes, so that the comparison can be made at the more general level first. Collect will copy all the text at a specified parent node and all nodes below it into a new node, without destroying the original detailed nodes.

- Click back to the Search & Compare Nodes dialog (opened from the toolbar button ) and from the **Others** tab select **Collect**. Type (1 2), the parent node for 'trusting' into the dialogue, and click **OK**. Your results will be under **Node Searches**. Rename the node search node to 'trust imm'.
- Repeat this procedure to Collect (1 3), and rename the resulting node to 'question imm'.
- In the Node Explorer, select **Tree Nodes** and right click to **Make top-level Tree Node** (this will probably be Node 7) as a working node. You might name this "Thinking about imm".
- **Cut** and attach each of your collect nodes ('trust imm' and 'question imm') to the new tree node.

### *Create the matrix*

- From the Search and Compare Nodes dialog, select **Others > Matrix**.
- In the Matrix dialog, type the address for the parent node for thinking about immunization in the first slot e.g. 7, select **Intersect** as the **Operator** from the drop-down list in the second slot, and type 10 3 as the parent node 'Education' in the third slot.

Intersection is appropriate because education is coded across whole documents, so any other node must intersect with it.)

- Click **OK** to run the matrix search.

The results will be saved as a node search result e.g. <<Node Search 6>> and will also appear in a Matrix Viewer. The description will tell you how the node was created, as well as how many cells the matrix has, and how many of these have text in them.

The children of the first of your specified nodes ('Thinking about imm') will form the rows of the matrix, and the children of the second ('Education') will define the columns. From this report you may be able to determine if there are any overall patterns in thinking, in relation to the education of the respondents. (Perhaps, indeed, those with less education are more likely than others to take a trusting approach.)

You can also export the table as a tab-separated text file for import into external applications (such as Excel or SPSS) or as a presentation-style plain text format.

- Click on the **Export Matrix...** button.
- From the options presented, choose **CELLS: Number of documents coded**, and **TABLE TYPE: Plain text file** displayed on screen.

A report will open with the summary table from your search. You can copy this into an external application or save it if you wish.

As above, the results of a matrix or vector search can be viewed in two ways: as a numeric summary, or as a text report. The numeric summary is a good starting point and it is already presented to you in the Viewer.

- Click on the **Make Report...** button and ensure **Show coding data** for each cell is selected with the options **Coded text, with Sub-Headers** (if you have not run a report since the last one above, you will need to uncheck **with Coding Stripes**.)

The text for each cell of the matrix will be provided sequentially (working across the rows), allowing you to review the intersection of each pair of nodes.

It is interesting, in the context of being trusting, that only those with tertiary education mentioned checking things out with others first. Similarly, those with tertiary education, in the context of questioning, talk in terms of weighing up the risks of vaccines versus diseases, while those with lower education (some secondary) talk of stories of possible long term effects of vaccines.

If the amount of data warranted it, you could have run matrix intersections for trusting and questioning separately, to get a cross-tabulation of each of the subnodes of these with each category of education.

## ***Exporting coding***

It is also possible to export coding tables that report the presence or absence of coding at nodes (or the amount of text coded at those nodes) for individual documents. It is another way of reviewing the patterns of coding, and if data is sufficient, to examine patterns of association between codes.

- From the **Project** Menu, select **Export Table > Coding**.
- In the **ROWS** pane, select **All Documents** option.
- In the **COLUMNS** pane, specify the node (1 2) 'trusting'.
- From **TREE NODE OPTIONS**, select **The node's children** to include these 3 nodes.

Note that the dialog tells you how many documents and nodes will appear in the table. You can also choose whether to omit items for which there is no coding.

- From the CELLS pane choose **Number of text units coded**
- From the TABLE TYPE pane, select the default option **Plain text file displayed on screen** then click **OK** to export the table.

A table appears showing for each node in that subtree (each child of the parent node specified) whether or not each of the documents is coded at it or not.

- Repeat the above for the parent node (1 3), to obtain a similar table for 'questioning'.

Like matrix tables, coding tables can also be exported as a file rather than being presented on screen. The resulting file (which is in tab-delimited text format) can then be opened in any table-based software, such as a spreadsheet or statistics program. Some further manipulation in that software may be necessary to deal with missing data in coding tables.

Take a moment to note the difference between exporting a matrix node and exporting coding. The matrix export generates a table display of the children of one node paired with each of the children of another. The coding export generates a table display of selected documents paired with each of the children of a specified node.

See the manuals and Online Help for more about exporting coding tables. Note that base data and case data tables can also be exported.

## Moving On

You have now learned a great deal about the tools for both qualitative analysis and mixed methods analysis offered by N6, and it is time to work on your own project. As you do, remember that additional help is available in the User Guide, and through Online Help. You might also join the qsr-forum, an e-mail list where users ask and answer questions arising as they work with the software. Further information about these resources is available through the QSR web site: [www.qsrinternational.com](http://www.qsrinternational.com)